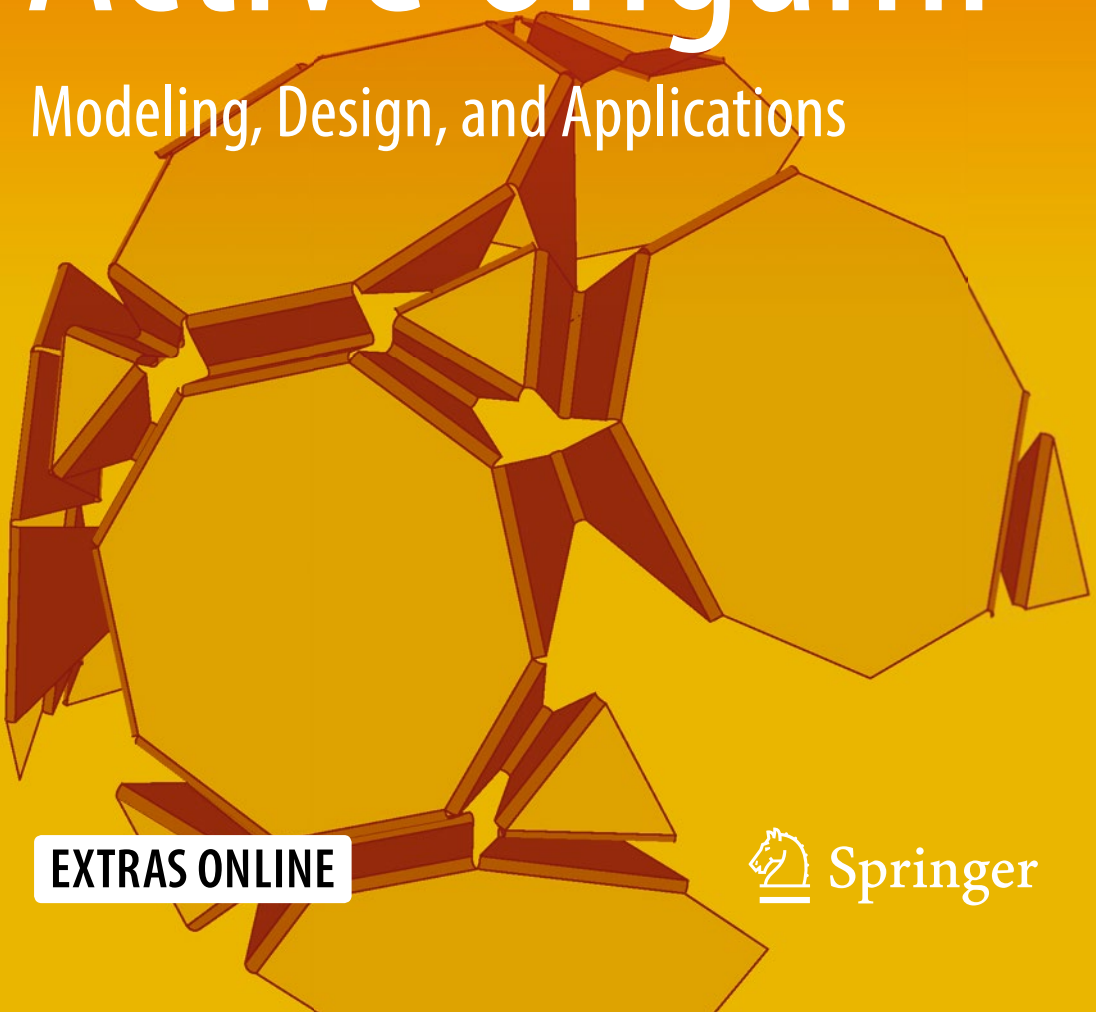
Edwin A. Peraza Hernandez
Darren J. Hartl
Dimitris C. Lagoudas

Active Origami

Modeling, Design, and Applications



EXTRAS ONLINE

Springer

Active Origami

Edwin A. Peraza Hernandez • Darren J. Hartl
Dimitris C. Lagoudas

Active Origami

Modeling, Design, and Applications

Springer

Edwin A. Peraza Hernandez
Department of Aerospace Engineering
Texas A&M University
College Station, TX, USA

Darren J. Hartl
Department of Aerospace Engineering
Texas A&M University
College Station, TX, USA

Dimitris C. Lagoudas
Department of Aerospace Engineering
Texas A&M University
College Station, TX, USA

ISBN 978-3-319-91865-5 ISBN 978-3-319-91866-2 (eBook)
https://doi.org/10.1007/978-3-319-91866-2

Library of Congress Control Number: 2018944147

To all of our loved ones.

Preface

Over the past decade, there has been an obvious expansion in the exploration and application of mathematical and computational origami. "Ted Talks," documentaries such as "Between the Folds," and pioneers in computational origami such as Robert Lang, Erik Demaine, Koryo Miura, and Tomohiro Tachi have shown us that the principles of folding two-dimensional sheets into three-dimensional forms can lead to both beautiful art and practical engineering solutions, often simultaneously. The ability of origami-based designs to be relatively easily fabricated in two dimensions, to enable deployment of large structures from small initial volumes, and to be inherently reconfigurable has captured the attention and imagination of engineers from across the world. As researchers in the nonlinear mechanics of active materials and smart structures, we have ourselves been intrigued by the theoretical challenges and implementation potential of folding structures since the topic first came to the forefront of our national research conversation, where it is expected to remain for some time.

This book represents our contribution to this growing field. Here we apply the three pillars of solid mechanics (conservation laws, constitutive modeling, and most especially kinematics) and combine them with our interest in structural design to present a full theory for origami structures. As the fundamentals of traditional origami theory (creased folds, flat foldability, etc.) have been well addressed in prior works, we focus in particular on the folding of engineering materials, including those having strain limitations and formed into sheets of finite thickness. We consider elastic and shape memory materials and develop the mathematical relations needed to design flat sheets for folding into arbitrary shapes and to model their full mechanical response as they do so.

The book begins with an overview of recent applications and theoretical developments of origami. Since a primary contribution of this work is the modeling of active origami structures having self-folding capabilities, a comprehensive survey of materials that enable this behavior and of applications that take advantage of it is presented. Toward the simulation of such applications, Chap. 2 begins the technical

treatment by reviewing the kinematic modeling of origami structures having creased folds. This allows for the establishment of a framework and especially notation that will carry the reader through the remainder of the book.

The design of three-dimensional forms from two-dimensional reference sheets is first addressed in Chap. 3, where the simplifications of unfolding polyhedra are employed. Design methods are expanded in applicability in Chap. 4, where the more general tuck-folding method of Tomohiro Tachi is developed for origami structures with creased folds. The technical novelty of this body of work is truly initiated in Chap. 5, where prior constraints associated with creased folds are eliminated, and a kinematic theory for the strain sensitive ("smooth") folding of engineering materials is presented. The unfolding polyhedra method for creating three-dimensional forms from two-dimensional sheets having smooth folds is presented in Chap. 6, and the tuck-folding method is likewise expanded to consider smooth folds in Chap. 7. Afterwards, we present the theory for the structural mechanics of origami sheets with smooth folds in Chap. 8, which is applicable to structures comprised of arbitrary materials (e.g., elastic materials, active materials). At the end of Chap. 8, we integrate the design methods of Chaps. 6 and 7 with the mechanics of active origami to develop a framework for the design of self-folding structures that morph toward goal shapes under the application of nonmechanical stimuli (e.g., thermal, chemical, electromagnetic).

Throughout the book, the reader is provided with clear development examples as well as problems that assess understanding and challenge further thought. Perhaps most importantly, the supplemental material includes the full MATLAB® scripts that represent the implementation of mathematical theories derived in Chaps. 2–7. The actual scripts used to generate the numeric and graphical results associated with several examples in these chapters are provided. It is hoped that the curious and motivated reader will use these tools to further advance the field of origami engineering and the applications of active origami.

We expect the readership of this book to range from advanced undergraduate students to experienced engineers and researchers with interest in origami engineering. Chapter 1 is recommended for a general reader exclusively interested in learning new developments and applications of origami engineering. An instructor interested in developing a short course on modeling and applications of origami may start with Chaps. 1 and 2, and perhaps extend to the design methods of Chaps. 3 and 4 if time allows. In particular, the problems at the end of Chap. 1 are suggested for a course where students will get familiarized with concepts of origami and freely available design and simulation tools for origami. Readers interested in simulating their own fold patterns or generating origami designs are encouraged to directly experiment with the codes provided in the Supplemental Materials and read their associated descriptions in Appendix B, although we suggest the code users to read the corresponding chapters to obtain higher understanding of the codes. We recommend reading Chaps. 2–4 to students and engineers exclusively interested in conventional origami with creased folds and its theory, numerical simulation approaches, and design methods. Chapters 5 and 8 are recommended for advanced students and engineers with interest in analyzing origami structures comprised of

engineering materials. For students, engineers, and researchers with the goal of advancing the field of computational origami, specially for applications that include active materials, we recommend them to absorb the material of Chaps. 5–8.

The authors would like to gratefully acknowledge a number of individuals and entities that have made this unified body of work possible. Our collaborators in our research on active origami, and most especially Ergun Akleman and Richard Malak, have contributed much to our thinking and progress. We appreciate the positive comments of individuals such as Robert Lang and Glaucio Paulino as they followed our developments over the past years. We would like to express our sincere gratitude to Robert Lang, Jun Mitani, Ying Liu, Michael Dickey, Samuel Felton, Tomohiro Tachi, Elliot Hawkes, David Gracias, Larry Howell, Milton Garza, Beatriz Borges, and others for providing figures used in this book. The proofreading help from Sameer Jape, Jacob Mingear, Pedro Leal, Patrick Walgren, Hannah Stroud, William Scholten, Brent Bielefeldt, Gregory Wilson, and Jobin Joy is also greatly appreciated. This monograph would not have been possible without the help of undergraduate researchers Cullen Nauck and Trent White, who assisted in the development of the Supplemental Materials. The joint support of the National Science Foundation (NSF) and the Air Force Office of Scientific Research (AFOSR) over the last 5 years with the Origami Design for Integration of Self-assembling Systems for Engineering Innovation (ODISSEI) program enabled all of the authors' efforts described herein, in addition to many of the other outside developments reviewed in Chap. 1.

Origami engineering is an important topic with potentially many new applications yet to be developed. We hope that our efforts combined with the support of our families and colleagues have allowed us to produce an important resource for those looking to make advances in the theory and practice of this exciting field.

College Station, TX, USA
March 2018

Edwin A. Peraza Hernandez
Darren J. Hartl
Dimitris C. Lagoudas

Contents

List of Symbols

a_i	$i = 1, 2, 3$. Shape memory alloy model parameters associated with the transformation hardening function
$\hat{a}_i$	Asymmetry variable of the ith smooth fold
A_f	Austenite finish temperature at zero stress
A_s	Austenite start temperature at zero stress
$\mathbf{b}$	Body force vector
$\mathbf{b}^{jk}$	Position vector of the corner of the kth edge module associated with the jth interior node of the goal mesh
$\mathbf{b}_L^{jk}$	Position vector of the point where the path $\boldsymbol{\gamma}^j(\eta)$ enters the kth fold adjacent to the jth interior fold intersection
$\breve{\mathbf{b}}_L^{jk}$	Position vector of the point where the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ enters the kth fold that it crosses
$\mathbf{b}_R^{jk}$	Position vector of the points where the path $\boldsymbol{\gamma}^j(\eta)$ exits the kth fold adjacent to the jth interior fold intersection
$\breve{\mathbf{b}}_R^{jk}$	Position vector of the point where the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ exits the kth fold that it crosses
$\mathbf{c}^i(\zeta_1)$	Fold cross-section parametric curve
$\hat{\mathbf{c}}^i(\zeta_1)$	Fold cross-section parametric curve expressed in the basis $\{\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i\}$
$\bar{\mathbf{c}}^i(\zeta_1)$	Non-dimensional form of $\hat{\mathbf{c}}^i(\zeta_1)$
$\hat{\mathbf{c}}_{L_j}^i$	End condition at $\hat{\mathbf{c}}^i(-1)$ required for jth-order geometric continuity
$\bar{\mathbf{c}}_{L_j}^i$	Non-dimensional form of $\hat{\mathbf{c}}_{L_j}^i$
$\hat{\mathbf{c}}_{R_j}^i$	End condition at $\hat{\mathbf{c}}^i(1)$ required for jth-order geometric continuity
$\bar{\mathbf{c}}_{R_j}^i$	Non-dimensional form of $\hat{\mathbf{c}}_{R_j}^i$
$\boldsymbol{C}$	Fourth-order stiffness tensor
C^A	Stress influence coefficient of austenite
$\mathbf{C}^{\mathcal{E}_I}$	Edge connectivity matrix
$\mathbf{C}^{\mathcal{F}}$	Fold connectivity matrix
$\mathbf{C}^{\mathcal{I}}$	Fold intersection connectivity matrix
C^M	Stress influence coefficient of martensite

$\mathbf{C}^{\mathcal{M}}$	Mesh connectivity matrix
$\mathbf{C}^{\mathcal{N}_I}$	Node connectivity matrix
$\mathbf{C}^{\mathcal{P}}$	Face connectivity matrix
$\boldsymbol{C}^{P}$	Stiffness matrix for plane stress
$\mathbf{C}^{\sharp}$	Trim connectivity matrix
$\mathbf{d}$	Vector having components corresponding to the design variables
$\mathbf{d}^j$	Translation constraint vector associated with the jth interior fold intersection
$\tilde{d}_i^{jk}$	$i = 1, 2$. Edge trim lengths associated with the kth edge of the jth face in the goal mesh
$\hat{d}_{mn}^{i}$	$m = 1, 2;\ n = 1, 2$. Edge trim lengths associated with the ith edge module
D	Shape memory alloy model parameter associated with the stress dependence of the critical thermodynamic force for transformation
$\mathbf{e}_i$	$i = 1, 2, 3$. Orthonormal vectors that define the fixed global coordinate system
$\hat{\mathbf{e}}_j^i$	$j = 1, 2, 3$. Fold-attached orthonormal vectors associated with the ith smooth fold
E	Young's modulus
E	Surface area efficiency
$\boldsymbol{E}$	Electric field vector
E^A	Young's modulus of austenite
E^M	Young's modulus of martensite
$\mathbf{f}^i$	Force vector of the ith point load
$\mathfrak{F}^i$	Volumetric region occupied by the ith smooth fold domain
$\boldsymbol{\mathcal{F}}^i(\zeta_1, \zeta_2)$	Surface parameterization of $\mathcal{F}_t^i$
$\mathcal{F}_0^i$	Reference configuration of the ith smooth fold
$\mathcal{F}_t^i$	Current configuration of the ith smooth fold
$\boldsymbol{g}$	Vector associated with the inequality constraints
$\mathcal{G}$	Smooth surface representing the goal shape
G^n	Order of geometric continuity
$\boldsymbol{h}$	Vector associated with the equality constraints
$\mathbf{h}^i$	Vector along the length direction of the ith smooth fold
$\mathfrak{h}_i$	Thickness of the ith fold domain
$h_{jk}(\zeta_1)$	$k = 0, \ldots, j$. Hermite interpolation polynomials of jth-order
H^{cur}	Transformation strain magnitude for full transformation
$\mathbf{H}^{jk}$	Transformation matrix associated with the kth fold adjacent to the jth interior fold intersection
$\breve{\mathbf{H}}^{jk}$	Folding transformation matrix
$H_{\min}$	Minimum transformation strain magnitude for full transformation
H_{sat}	Maximum attainable transformation strain magnitude for full transformation
$\mathbf{I}_n$	$\mathbb{R}^{n\times n}$ identity matrix
k	Transformation strain exponential parameter

K_j	Discrete Gaussian curvature associated with the jth interior vertex
$\mathbf{l}^{jk}$	Vector with start point and end point respectively corresponding to the points where the path $\boldsymbol{\gamma}^j(\eta)$ enters and exits the face between the kth and $(k+1)$th folds adjacent to the jth interior fold intersection
$\hat{l}_i$	Trim length associated with the ith interior edge of the goal mesh
$\tilde{l}_{jk}$	Trim length associated with the kth edge of the jth face of the goal mesh
$\mathfrak{L}$	Penalty function associated with kinematic constraints
$\mathbf{m}^{jk}$	Vector along the length of the kth fold centerline incident to the jth interior vertex that emanates from such a vertex
$\breve{\mathbf{m}}^{jk}$	Vector along the fold centerline of the kth fold crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$
M	Moment resultant
$\mathcal{M}$	Goal mesh
$\mathcal{M}^j$	jth face of the goal mesh
M_f	Martensite finish temperature at zero stress
M_s	Martensite start temperature at zero stress
$\mathcal{M}_\sharp$	Trimmed mesh
$\mathcal{M}_\sharp^j$	jth face of the trimmed mesh
$\hat{\mathbf{n}}^{ij}$	$j = 1, 2$. Unit normal vectors of the faces connected by the ith interior edge
$\mathbf{n}^{jk}$	Unit normal vector of the kth face adjacent to the jth interior node in the goal mesh
$\bar{\mathbf{n}}^{jk}$	Unit normal vector associated with the kth edge incident to the jth interior node in the goal mesh
$\bar{n}^{\mathcal{M}}$	Average number of edges incident to the interior nodes of the goal mesh
n_i	$i = 1, 2, 3, 4$. Transformation hardening exponents
n_j	Number of fold adjacent to the jth interior fold intersection
$\breve{n}_j$	Number of folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$
$n_j^{\mathcal{M}}$	Number of nodes of the jth face in the goal mesh
$n_j^{\mathcal{N}}$	Number of faces connected to jth interior node in the goal mesh
$\mathbf{n}_{\mathcal{M}}$	Unit normal vector of the goal mesh
N	Axial force resultant
N_{adj}	Number of increments in the step associated with adjustment of fold angles
$N_{\mathcal{B}}$	Number of vertices located at the boundary or outside the sheet
N_{dof}	Number of degrees of freedom
$N_{\mathcal{E}}$	Number of edges in the goal mesh
$N_{\mathcal{E}}^I$	Number of interior edges in the goal mesh
$N_{\mathcal{E}}^{Ij}$	$j = 0, 1, 2$. Number of interior edges incident to j interior nodes
$N_{\mathcal{F}}$	Number of folds in the sheet
N_{fol}	Number of increments in the step associated with constant guess increments for the fold angles

$N_{\mathcal{I}}$	Number of fold intersections in the sheet
N_{inc}	Total number of increments during numerical simulation
$N_{\mathcal{M}}$	Number of faces in the goal mesh
$N_{\mathcal{N}}$	Number of nodes in the goal mesh
$N_{\mathcal{N}}^{I}$	Number of interior nodes in the goal mesh
$N_{\mathcal{P}}$	Number of faces in the sheet
p	Height of the focus point of a parabolic surface
$\hat{\mathbf{p}}^{ij}$	$j = 1, 2, 3, 4$. Position vectors of the corner points of the ith smooth fold in the reference configuration
$\mathcal{P}_0^i$	Reference configuration of the ith face
$\mathcal{P}_t^i$	Current configuration of the ith face
$\mathbf{q}$	Position vector of a point in a parabolic surface
$\mathbf{Q}_i(\phi)$	Transformation matrix in homogeneous coordinates associated with a rotation by ϕ about an axis aligned to $\mathbf{e}_i$
$\hat{r}^{i1}, \hat{r}^{i2}$	Length parameters associated with the ith smooth fold
R	Radius of a parabolic surface
$\mathfrak{R}$	Residual vector
$\mathbf{R}^j$	Rotation constraint matrix associated with the jth interior fold intersection
$\mathfrak{R}^s$	Vector of residuals associated with fold arc-lengths
$\mathfrak{R}^\theta$	Vector of residuals associated with fold angles
$\mathbf{R}_i(\phi)$	Transformation matrix associated with a rotation by ϕ about an axis aligned to $\mathbf{e}_i$
$\hat{\mathbf{s}}$	Vector having components corresponding to the fold arc-lengths of the sheet
$\hat{s}_i$	Arc-length of the ith fold cross-section
$\bar{s}_i$	Non-dimensional arc-length of the ith fold cross-section
$\mathcal{S}_0$	Sheet reference configuration
$\mathcal{S}_t$	Sheet current configuration
$\mathcal{S}_\star$	Sheet goal configuration
t	Time parameter
$\mathbf{t}$	Traction vector
$\bar{\mathbf{t}}$	Applied traction vector
$\mathbf{t}^i(\zeta_1)$	Unit tangent vector of the curve $\hat{\mathbf{c}}^i(\zeta_1)$
T	Absolute temperature
$\mathbf{T}(\mathbf{b})$	Transformation matrix in homogeneous coordinates associated with a translation by vector $\mathbf{b}$
T_0	Reference absolute temperature
`tol`	Numerical tolerance
$\mathbf{u}$	Displacement vector
$\bar{\mathbf{u}}$	Applied displacement vector
$\mathbf{u}^i$	Displacement vector of the point of application of the ith point load
$\mathbf{v}$	Velocity vector
$\hat{\mathbf{v}}^{i1}$	Position vector of the vertex from which the ith fold centerline emanates

$\hat{\mathbf{v}}^{i2}$	Position vector of the vertex at which the ith fold centerline ends
$\mathbf{v}^j$	Position vector of the jth vertex
v_f	Volume fraction of fibers embedded in a composite
$\hat{\boldsymbol{w}}^0$	Vector whose components correspond to the fold widths in the sheet
$\mathbf{w}^{jk}$	Vector with start point and end point corresponding to the points where the path $\boldsymbol{\gamma}^j(\eta)$ crosses the boundaries of the kth fold adjacent to the jth interior fold intersection
$\hat{w}_i$	Current distance between the end points of the cross-section curve of the ith fold
$\hat{w}_i^0$	Fold width of the ith fold
$\hat{w}_i^{\mathsf{E}0}$	Fold width of the exterior folds in the ith edge module
$\hat{w}_i^{\mathsf{E}\star}$	Distance between the end points of the cross-section curve of the exterior folds in the ith edge module in the goal configuration
$\hat{w}_i^{\mathsf{I}0}$	Fold width of the interior fold in the ith edge module
$\hat{w}_i^{\mathsf{I}\star}$	Distance between the end points of the cross-section curve of the interior fold in the ith edge module in the goal configuration
w_{jk}	Distance between the end points of the cross-section curve of the kth fold adjacent to the jth interior fold intersection
$\breve{w}_{jk}$	Distance between the end points of the cross-section curve of the kth fold crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$
w_{jk}^0	Fold width of the kth fold adjacent to the jth interior fold intersection
$\breve{w}_{jk}^0$	Fold width of the kth fold crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$
$\hat{\boldsymbol{w}}_L^0$	Vector whose components correspond to the lower bounds of the fold widths in the sheet
$\hat{\boldsymbol{w}}_U^0$	Vector whose components correspond to the upper bounds of the fold widths in the sheet
$\hat{\boldsymbol{W}}$	Vector having components corresponding to the length variables $\hat{W}_1, \ldots, \hat{W}_{N_{\mathcal{E}}^I}$
W_E	External work exerted by body forces and boundary tractions
$\hat{W}_i$	Length design variable of the ith edge module
W_I	Internal work
W_{jk}	Length design variable of the kth edge module associated with the jth interior vertex of the goal mesh
$\mathbf{x}$	Position vector of a point of the sheet in the current configuration
$\mathbf{X}$	Position vector of a point of the sheet in the reference configuration
$\hat{\mathbf{y}}^i$	Position vector of the ith node in the goal mesh
$\tilde{\mathbf{y}}^{jk}$	Position vector of the kth node of the jth face in the goal mesh
$\tilde{\mathbf{y}}_\sharp^{jk}$	Position vector of the kth node of the jth face in the trimmed mesh
Y^t	Critical thermodynamic driving force for transformation
$\hat{\mathbf{z}}^i$	Vector along the length of the ith interior edge of the goal mesh
$\mathbf{z}^{jk}$	Vector along the edge connecting the jth interior node to its kth adjacent node
α	Thermoelastic expansion coefficient

$\boldsymbol{\alpha}$	Second-order tensor of thermoelastic expansion coefficients
α_{jk}	Angle between adjacent fold centerlines incident to an interior vertex
β_{ij}	Shape parameter of $\hat{\mathbf{c}}^i(\zeta_1)$ arising from the condition for jth-order geometric continuity
$\boldsymbol{\gamma}^j(\eta)$	Closed path enclosing the jth interior fold intersection
$\breve{\boldsymbol{\gamma}}^j(\eta)$	Path connecting the fixed face to the jth face
$\Delta\hat{\boldsymbol{\theta}}^l$	lth guess increment for the fold angles
$\boldsymbol{\varepsilon}$	Second-order linearized strain tensor
$\boldsymbol{\varepsilon}^A$	Second-order tensor of actuation strains
$\boldsymbol{\varepsilon}^{MS}$	Second-order tensor of strains caused by changes in the material microstructure
ε^t	Transformation strain
ε^{t0}	Pre-strain
$\boldsymbol{\varepsilon}^{th}$	Second-order tensor of thermoelastic strains
$\boldsymbol{\varepsilon}^{t-r}$	Second-order transformation strain tensor at the cessation of forward transformation
ε_0	Mid-surface strain
ε_{mn}	Components of the linearized strain tensor in the basis $\{\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i\}$
$\hat{\boldsymbol{\theta}}$	Vector having components corresponding to the fold angles
$\hat{\boldsymbol{\theta}}^l$	Vector having components corresponding to the fold angles at the lth increment during numerical simulation
$\hat{\boldsymbol{\theta}}^\star$	Vector having components corresponding to the fold angles in the goal configuration
$\hat{\theta}_i$	Fold angle of the ith fold
$\hat{\theta}_i^L$	Lower bound for the fold angle of the ith fold
$\hat{\theta}_i^U$	Upper bound for the fold angle of the ith fold
θ_{jk}	Fold angle of the kth fold adjacent to the jth interior fold intersection
$\breve{\theta}_{jk}$	Fold angle of the kth fold crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$
$\hat{\Theta}_i$	Dihedral angle of the ith interior edge in the goal mesh
$\kappa_i(\zeta_1)$	Signed curvature of $\hat{\mathbf{c}}^i(\zeta_1)$
$\hat{\kappa}_i(\zeta_1)$	Curvature of $\hat{\mathbf{c}}^i(\zeta_1)$
$\tilde{\kappa}_i(\zeta_1)$	Non-dimensional curvature of $\hat{\mathbf{c}}^i(\zeta_1)$
$\bar{\kappa}_i(\zeta_1)$	Non-dimensional signed curvature of $\hat{\mathbf{c}}^i(\zeta_1)$
$\kappa_i^G(\zeta_1)$	Goal signed curvature of $\hat{\mathbf{c}}^i(\zeta_1)$
$\bar{\kappa}_i^G(\zeta_1)$	Goal non-dimensional signed curvature of $\hat{\mathbf{c}}^i(\zeta_1)$
λ_B	Penalty weight for residuals from fold angle bounds
λ_d	Penalty weight for residuals from translation constraints
λ_R	Penalty weight for residuals from rotation constraints
ν	Poisson's ratio
ξ	Martensite volume fraction
ξ^r	Martensite volume fraction at the cessation of forward transformation
Ξ	Field variable
$\bar{\Xi}$	Applied field variable

ρ	Mass density
$\boldsymbol{\sigma}$	Second-order Cauchy stress tensor
σ_{mn}	Components of the Cauchy stress tensor in the basis $\{\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i\}$
$\hat{\tau}_j^i$	$j = 1, 2$. Trim angles of the ith edge module
τ_{jk}	Trim angle of the kth edge module associated with the jth interior node of the goal mesh
ϕ_{jk}	Corner angle of the kth face adjacent to the jth interior node of the goal mesh
$\tilde{\phi}_{jk}$	kth interior corner angle of the jth face in the goal mesh
$\boldsymbol{\Phi}^t$	Transformation function
$\boldsymbol{\Phi}^t_{\text{fwd}}$	Transformation function for forward transformation
$\boldsymbol{\Phi}^t_{\text{rev}}$	Transformation function for reverse transformation
$\varphi(\mathbf{y})$	Counterclockwise angle from $\mathbf{e}_1$ to $\mathbf{y} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$
$\hat{\boldsymbol{\psi}}$	Vector having components corresponding to the angular variables $\hat{\psi}_1, \ldots, \hat{\psi}_{N_{\mathcal{E}}^I}$
$\hat{\psi}_i$	Angular design variable of the ith edge module
ψ_{jk}	Angular design variable of the kth edge module associated with the jth interior vertex of the goal mesh
Ω	Volumetric region occupied by the material body
$\boldsymbol{\omega}^{jkl}$	Vector along the intersection axis of the kth and the lth edge modules associated with jth interior node
$\partial\mathcal{M}$	Boundary of the goal mesh
$\partial\mathcal{M}^C$	Introduced boundary edges
$\partial\mathcal{M}^O$	Boundary edges of the given goal mesh
$\partial\mathcal{S}_0$	Boundary of the sheet in the reference configuration
$\partial\Omega$	Boundary of the volumetric region occupied by the material body
$\partial\Omega_u$	Boundary of the volumetric region occupied by the material body where displacement boundary conditions are applied
$\partial\Omega_\sigma$	Boundary of the volumetric region occupied by the material body where traction boundary conditions are applied
$\mathbf{0}_n$	Zero vector in $\mathbb{R}^n$
$\varnothing$	Empty set

Chapter 1
Introduction to Active Origami Structures

Abstract Origami, the ancient art of paper folding, has inspired the design and functionality of engineering structures for decades. The underlying principles of origami are very general, it takes two-dimensional components that are easy to manufacture (sheets, plates, etc.) into three-dimensional structures. More recently, researchers have become interested in the use of *active materials* that convert various forms of energy into mechanical work to produce the desired folding behavior in origami structures. Such structures are termed *active origami structures* and are capable of folding and/or unfolding without the application of external mechanical loads but rather by the stimulus provided by a non-mechanical field (thermal, chemical, electromagnetic). This is advantageous for many areas including aerospace systems, underwater robotics, and small scale devices. In this chapter, we introduce the basic concepts and applications of origami structures in general and then focus on the description and classification of active origami structures. We finalize this chapter by reviewing existing design and simulation efforts applicable to origami structures for engineering applications.

1.1 Origami Structures

Traditionally, the term *origami* has been primarily associated with the ancient art of folding paper. *Origami* has the Japanese roots *oru* meaning *folded*, and *kami* meaning *paper* [1, 2]. Its original purpose was not particularly utilitarian, but rather recreational and artistic [3]. Origami was and remains the art of folding sheets of paper into decorative and often intriguing shapes, either abstract in form or representative of realistic objects. Figure 1.1 shows examples of paper origami. These examples range from those that are relatively simple such as the Miura-Ori and the three twisted boxes (Fig. 1.1a and b, respectively) to those of high complexity such as the scorpion and mask (Fig. 1.1c and d, respectively). Currently, the base materials for origami structures extend far beyond paper and include metals, polymers, and other complex materials.

In origami, a goal shape is obtained from an initially planar sheet exclusively through *folding*. In paper origami, folding is often obtained by creasing the paper

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_1

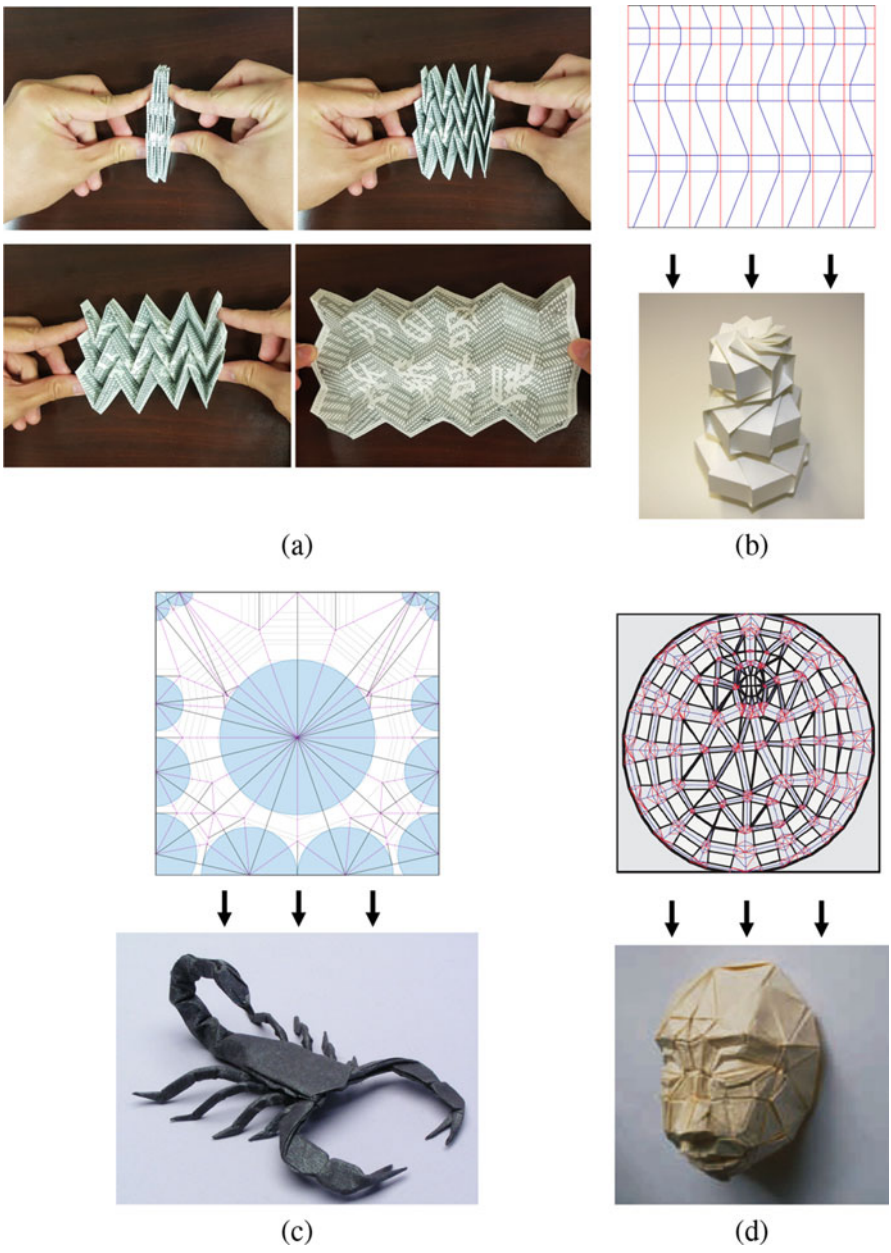


Fig. 1.1 Examples of paper origami: (**a**) Miura-Ori; (**b**) Three twisted boxes. *Credit: Jun Mitani* (http://mitani.cs.tsukuba.ac.jp/ja/cp_download.html); (**c**) "Scorpion varileg." *Credit: Robert Lang* (http://www.langorigami.com/composition/scorpion-varileg-opus-379); (**d**) Mask. © 2010 IEEE. Reprinted with permission from [4]

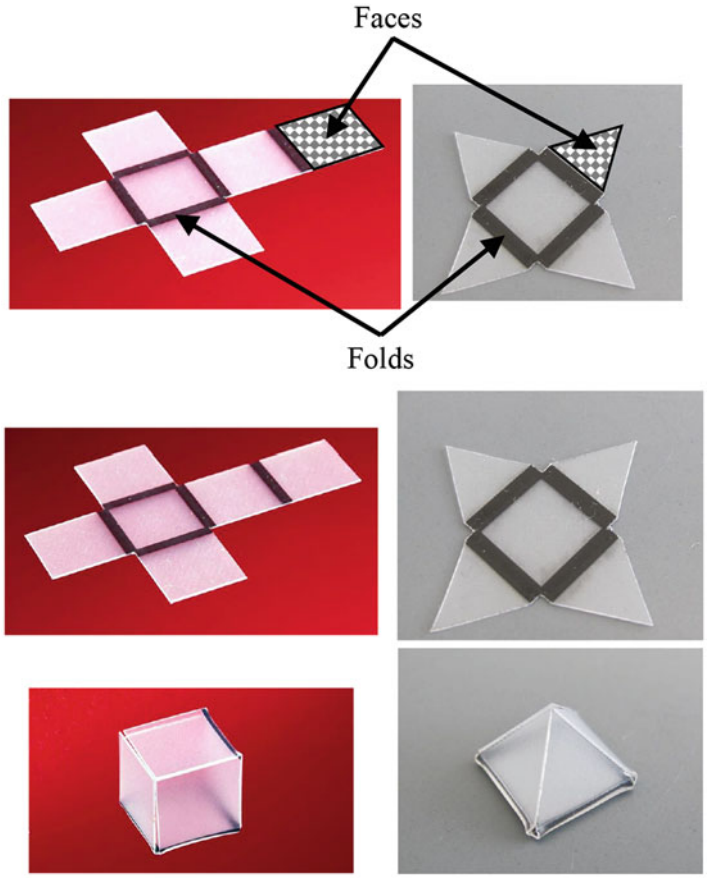


Fig. 1.2 Identification of folds (localized bent regions) and faces of a foldable cube and a foldable pyramid. The folding motion of the cube and the pyramid is driven by shape memory polymer (SMP) actuation triggered by heating through light absorption. Adapted from [5] with permission of The Royal Society of Chemistry. Credit: Ying Liu and Michael Dickey

along straight line segments as shown in the examples of Fig. 1.1. For origami structures having larger thickness than a sheet of paper or comprised of engineering materials for which direct creasing is not possible or desirable, destructive scoring and creasing might not be performed, and folding is instead obtained by bending localized regions as shown in Fig. 1.2.

In any origami structure, we can identify two region types: the *folds* and the *faces*. The *faces* correspond to the regions bounded by the folds and by the boundary of the sheet. In Figs. 1.2 and 1.3, we indicate regions corresponding to folds and faces for different origami structures. The layout of the folds in the planar configuration of the sheet is termed the *fold pattern*. The fold patterns for the three twisted boxes, the scorpion, and the mask are shown in Fig. 1.1b, c, and d, respectively.

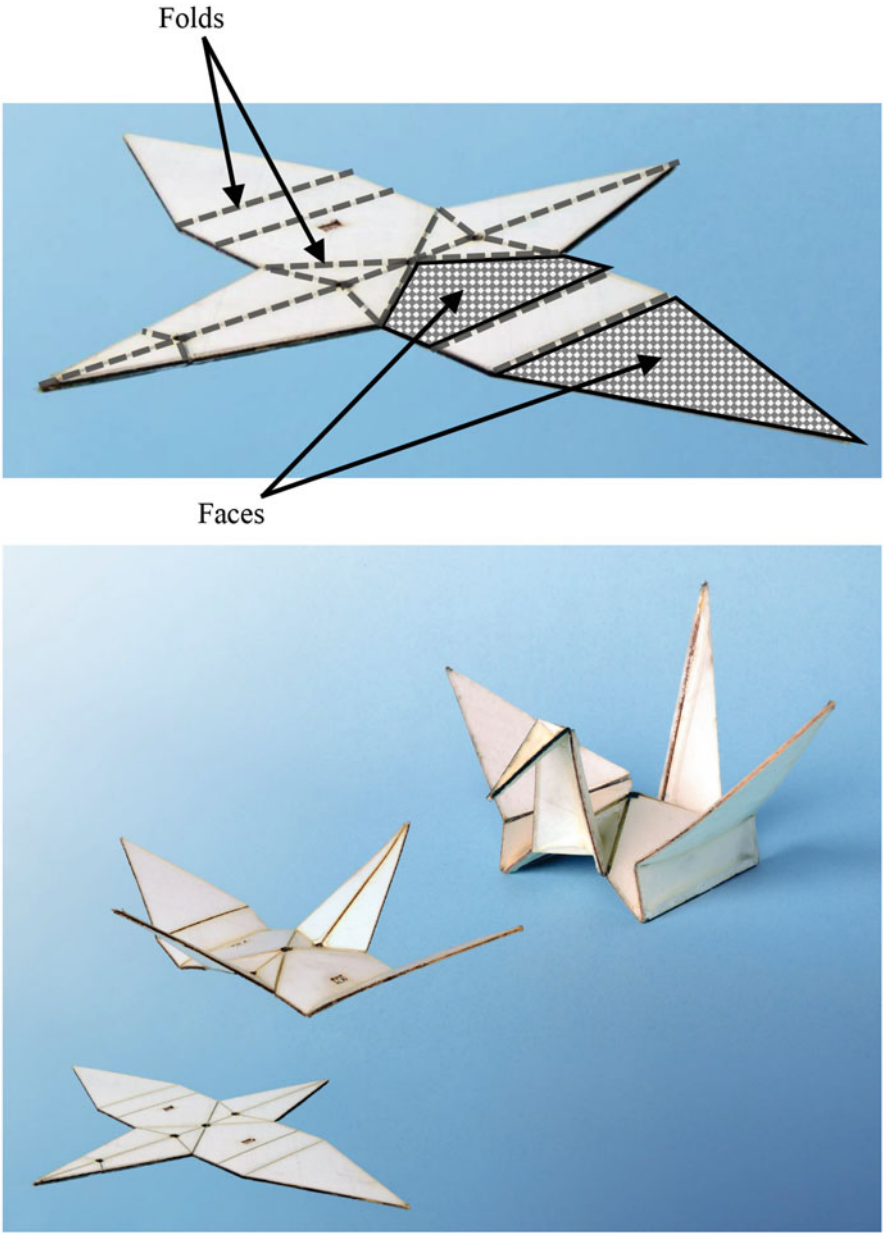


Fig. 1.3 Identification of folds (creases) and faces of a crane. The folding motion of this crane is driven by SMP actuation. Adapted from [6] with permission of The Royal Society of Chemistry. Credit: Samuel Felton

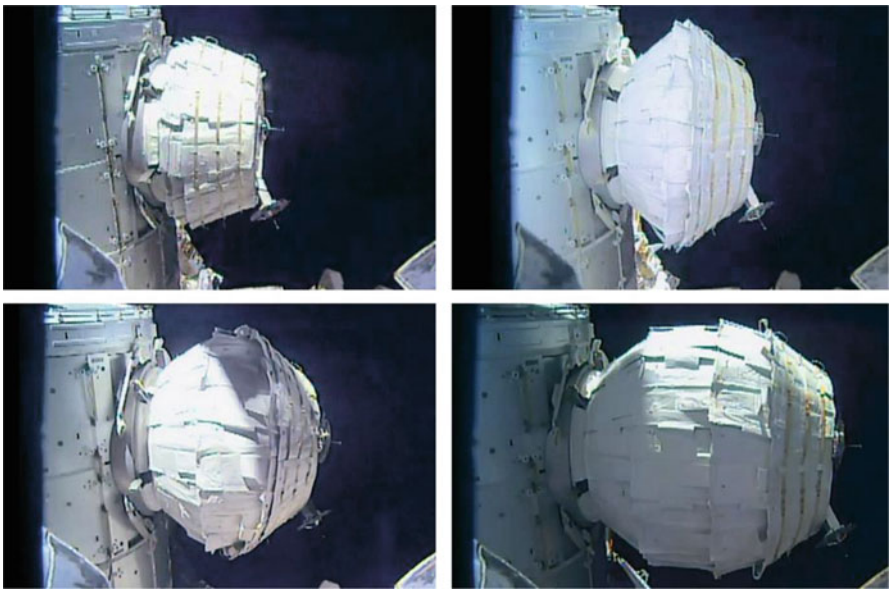


Fig. 1.4 Demonstration of unfolding-based expansion of the Bigelow Expandable Activity Module (BEAM). Credit: NASA TV (https://blogs.nasa.gov/spacestation/2016/05/28/beam-expanded-to-full-size/)

Researchers discovered in the 1970s that an endless number of shapes could in theory be mathematically attained using conventional origami (initially planar shape with only *folding* allowed) [2]. These discoveries enabled new approaches for manufacturing, assembling, and morphing of devices and structures based on origami, which is evident from the increasing attention mathematicians, scientists, and engineers have given to research on origami during the past decades [7–12].

Potential engineering advantages of origami structures include compact storage/deployment capabilities (e.g., airbags [13, 14]), potential for reconfigurability [15–22], and reduction of manufacturing complexity [23–28] (reduced part counts and improved assembly using collapsible/deployable components). Moreover, origami has been utilized across scales through its applications ranging from the nano- and micro-scales [29–34] to deployable aerospace structures at the macro-scale [35–38]. For example, Fig. 1.4 shows the origami-inspired Bigelow Expandable Activity Module (BEAM). As the BEAM is pressurized, the various creases in the structure unfold until the structure reaches its full volume expanded configuration. Another space application of origami is the design of deployable solar arrays for spacecraft. Figure 1.5 shows a computational simulation of an origami-based deployable solar array [39]. The solar array can be stowed in a compact folded configuration for launch and then unfolded in space towards its maximum solar panel area configuration.

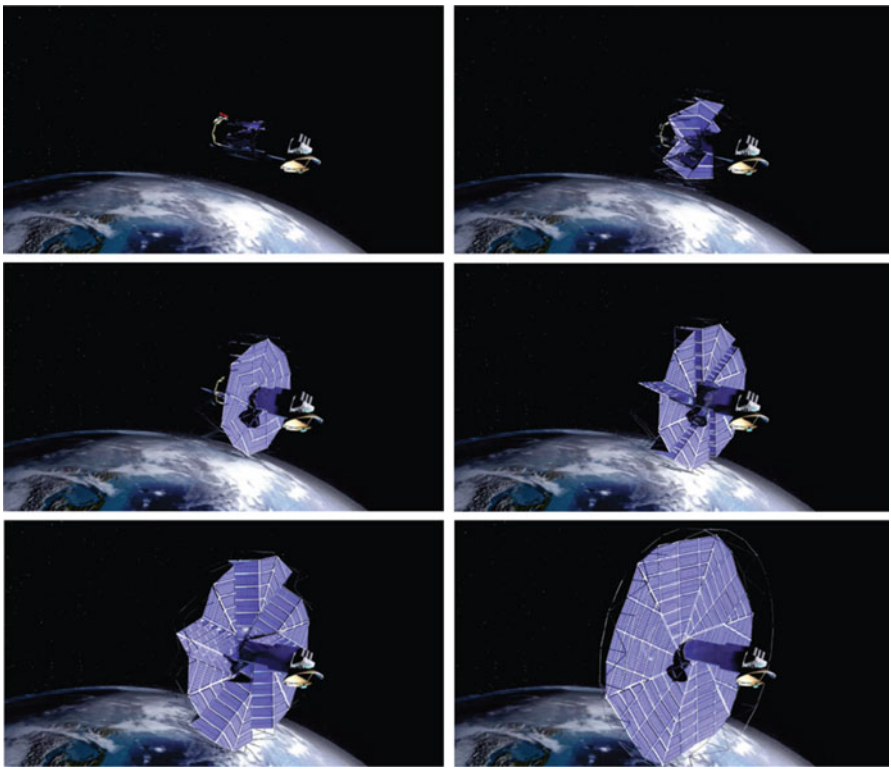


Fig. 1.5 Simulation of an origami-based deployable solar array for spacecraft applications. Reproduced from [39] with permission of the American Society of Mechanical Engineers (ASME)

Origami has been used to develop foldable aerostructures [40–47]. Figure 1.6 shows an example of a self-deployable origami gliding wing that can be folded in a compact form for storage [47]. Upon the release of the constraints maintaining the structure in its compact form, the stored elastic energy in the structure allows for the unfolding towards its gliding wing form without the need for actuation.

Origami has also inspired the manufacturing and design of robots [48–58]. Two examples of applications of origami in robotics are shown in Figs. 1.7 and 1.8. Figure 1.7 shows a robot fabricated using an origami-inspired technique whereby the structure starts as a planar sheet with embedded electronics and shape memory polymer (SMP) actuators, and then transforms under the application of heat into a functional robot via folding driven by SMP actuation [59]. Figure 1.8 shows a robot having origami wheels with variable size that are manipulated through internal actuators that drive their foldable outer shape to collapse or expand [49, 60]. By changing the size of the wheels, the robot can trade speed for the ability to move in smaller spaces.

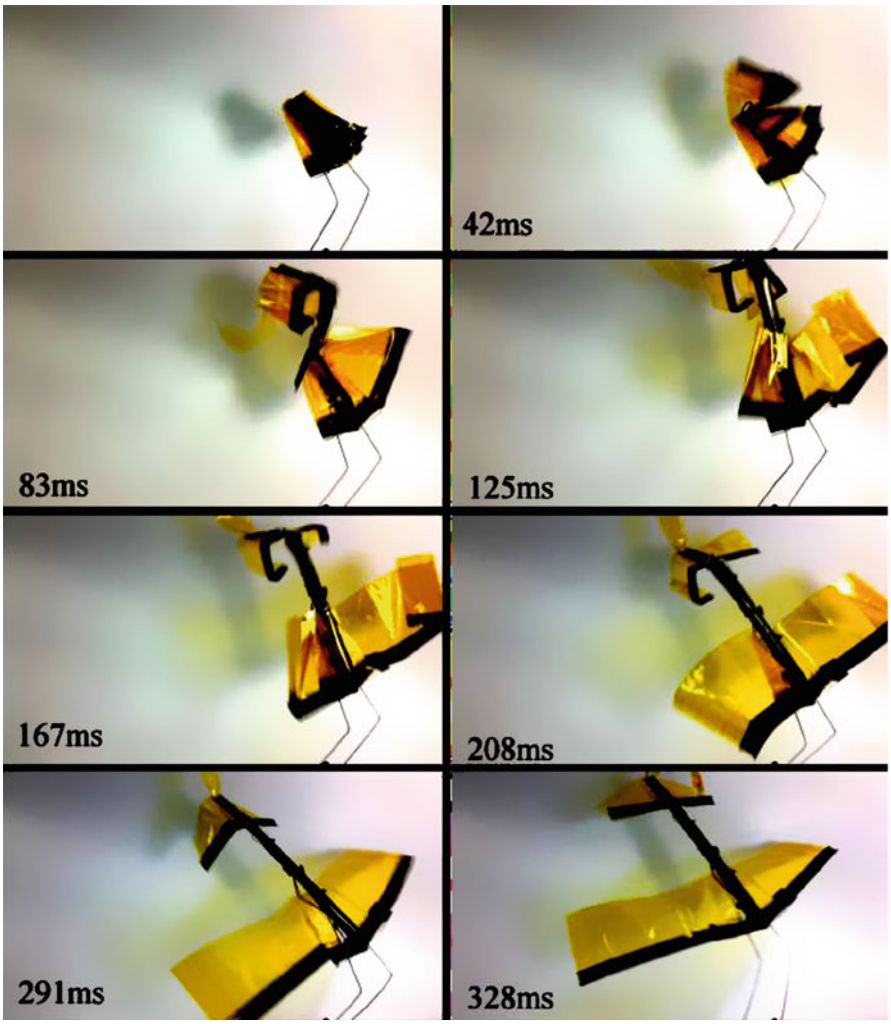


Fig. 1.6 Elastic deployment of an origami gliding wing. Reproduced from [47] with permission of the American Society of Mechanical Engineers (ASME)

Several biomedical devices are based on origami [61–67]. For example, Fig. 1.9 shows an origami stent fabricated from a shape memory alloy (SMA) foil having several creases [61]. The deployment of any such stent can be achieved via the shape memory effect activated by body temperature or by making use of the pseudoelastic effect exhibited by SMAs [68].

Various shelters and similar architectural deployable structures have been inspired by origami [69–75]. Figure 1.10 shows a prototype for a deployable origami structure connecting the entrances of two buildings [75]. Howell and

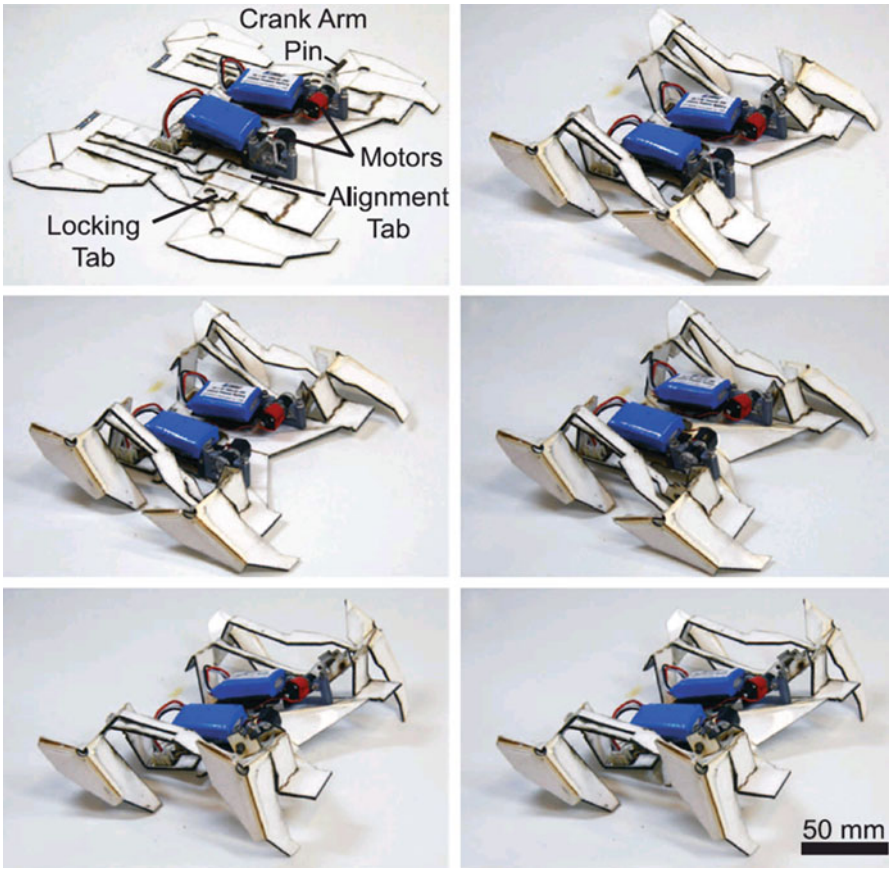


Fig. 1.7 Self-assembly of a crawling robot through folding via thermally induced SMP actuation. From [59]. Reprinted with permission of the American Association for the Advancement of Science (AAAS). Credit: Samuel Felton

coworkers proposed an origami-inspired design for lightweight ballistic barriers to protect law enforcement [76]. Figure 1.11 illustrates their design. The ballistic barrier is comprised of twelve layers of Kevlar. The barrier can be compactly stowed for transportation in a vehicle and rapidly deployed when required. The origami-inspired bulletproof shield weights only 55 pounds, where most steel-based barriers in current use weight approximately 100 pounds.

Engineering developments towards the practical fabrication of origami structures at various scales have been proposed by Huang, Rogers, and coworkers [77–79]. Their approach to fabricate three-dimensional origami structures consists of attaching selected regions of a planar origami sheet to a pre-stretched elastomeric substrate. The substrate undergoes in-plane shrinkage upon release of sheet/substrate assembly, causing controlled compressible buckling of certain regions of

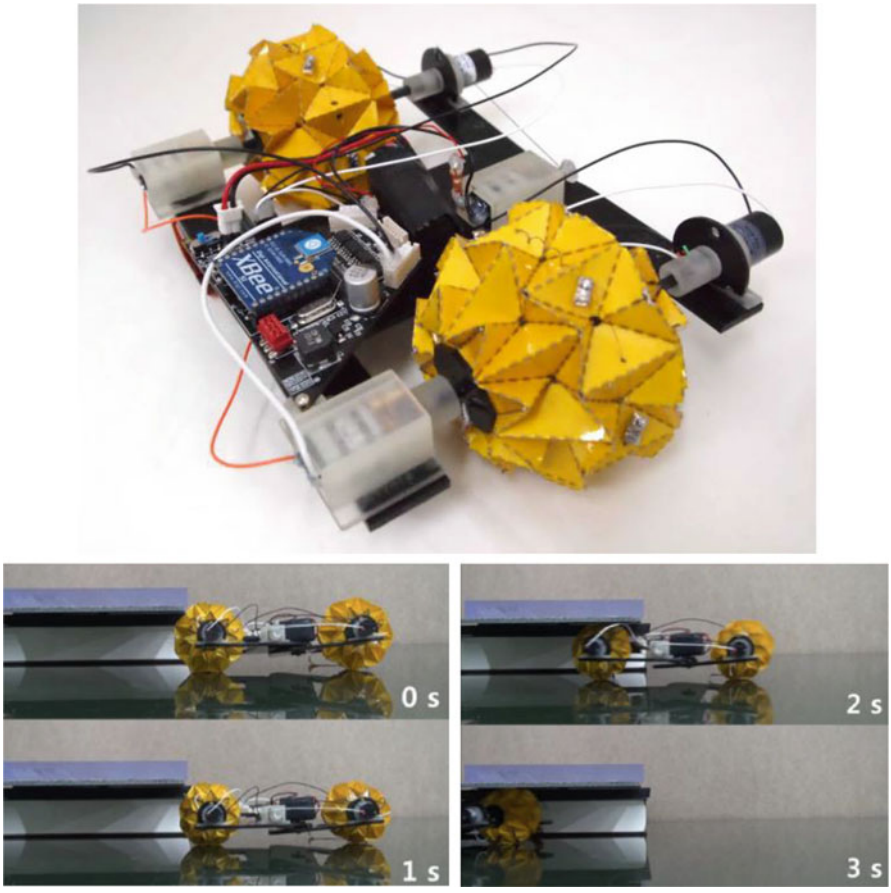


Fig. 1.8 Robot with variable size origami wheels. The size of the origami wheels can be manipulated through actuated folding and unfolding, allowing the robot to move in smaller spaces if needed. Reproduced from [49] with permission of the American Society of Mechanical Engineers (ASME)

the sheets (the folds) and forming the targeted goal configuration. Figure 1.12 shows various examples of origami structures fabricated via such an approach. The structures in Fig. 1.12 are made of copper films having thickness of 30 μm.

DNA nanotechnology is one of the newest areas where origami principles are being applied [29]. In *DNA origami*, a long single thread of DNA, known as the *scaffold*, is folded towards an arbitrary shape by short synthetic molecules (oligonucleotides [30]), known as the *staple strands*. Each staple strand is designed to adhere to a distinct location along the scaffold. The staple strands determine the size and shape of the resulting compact structure. DNA origami has been employed to study single-molecule chemical reactions and to organize several molecules such

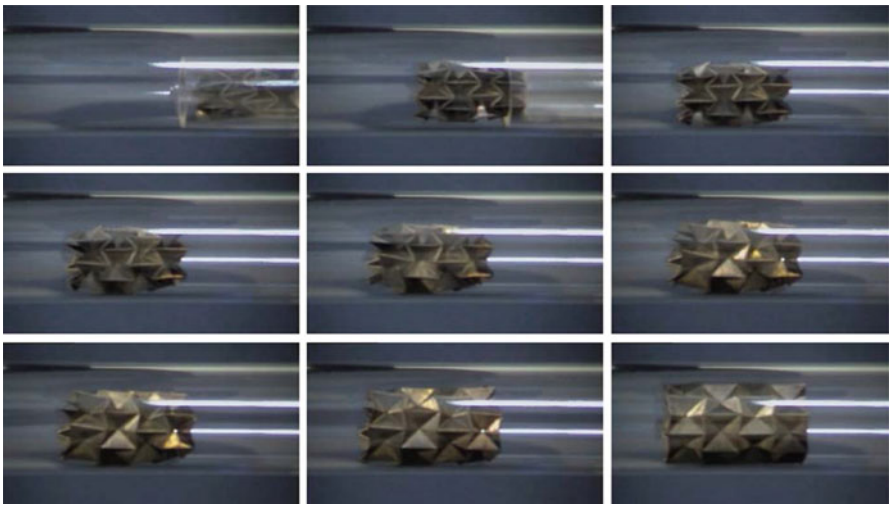


Fig. 1.9 Expansion of an SMA origami stent. Reprinted from [61], Copyright (2006), with permission from Elsevier

as proteins, metal nanoparticles, and carbon nanotubes [31]. Detailed reviews on DNA origami can be found in [30–32, 80].

Other applications of origami structures include: various space structures (e.g., solar panels, telescope lenses [81]), micro-mirrors [82], electronic components [83–88], foldcore-based structures for enhanced mechanical properties and impact resistance [89–99], crash boxes and other energy absorption systems [100–105], metamaterials [106–116], microelectromechanical systems (MEMS) [18, 117–125], and many others [7, 126, 127]. To keep up to date with the newest developments on active origami structures, we refer the reader to the journals included in the bibliography of this chapter. For more general publications of advancements in origami theory and applications, the reader is referred to the proceedings of the International Meetings of Origami Science, Mathematics, and Education (OSME) [128–131] and the Journal of Mechanisms and Robotics.

As briefly mentioned at the beginning of the chapter, a key observable feature of practical origami structures is the shape of their folds, which can be of *hinge-type* or *bending-type* as illustrated in Fig. 1.13. Many origami structures such as those folded from paper utilize hinge-type folds that are idealized as line segments on the sheet that behave as rotational hinges (i.e., axes of panel rotation) during folding. In this book, we denote these folds as *creased folds*. An example of an origami structure with creased folds is shown in Figs. 1.1 and 1.3. In other cases, origami structures may have significantly large thickness or be comprised of materials for which creased folds are not desirable (since creasing may damage the materials and reduce engineering performance), and folding is obtained by bending localized regions. Such folds cannot be idealized as line segments in the sheet but rather as bent regions of non-zero surface area. We denote these folds as *smooth folds*.

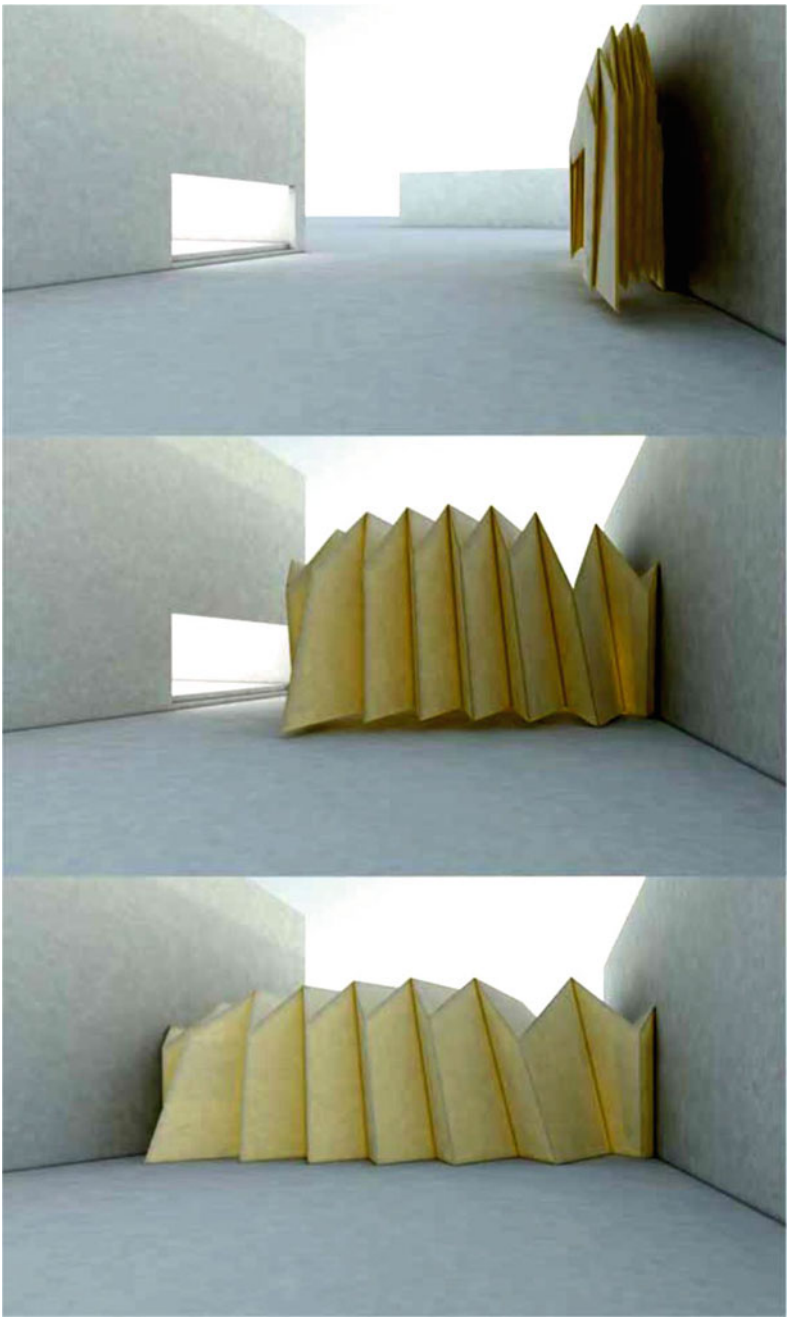


Fig. 1.10 Prototype of a deployable origami structure connecting the entrances of two buildings [75]. Credit: Tomohiro Tachi

Fig. 1.11 A deployable bulletproof origami shield [76]. Credit: Brigham Young University (BYU) Photo and Larry Howell

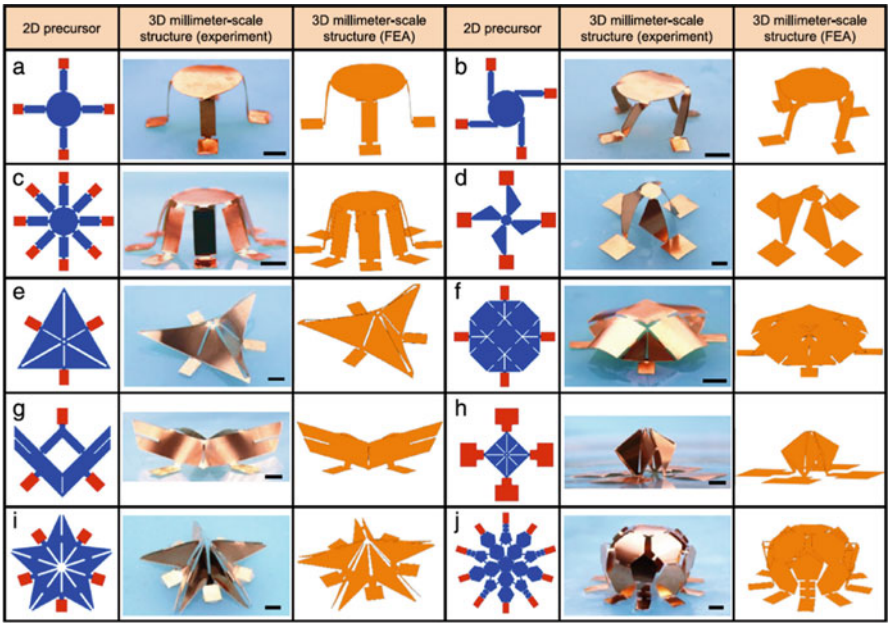


Fig. 1.12 Results of computational and experimental studies of metallic origami structures at the millimeter scale. The structures are folded via compressive buckling resulting from in-plane shrinkage of a pre-stretched elastomeric substrate attached at selected regions of the sheets. The planar precursors, optical images of the goal configurations, and finite element analysis (FEA) simulations for ten samples are shown. The scale bars denote 4 mm. (**a**)–(**c**) Tables. (**d**) Windmill. (**e**) Tent. (**f**) Altar. (**g**) Plane. (**h**) Asparagus pea. (**i**) Five-pointed star. (**j**) Soccer ball. Reprinted from [77], Copyright (2016), with permission from Elsevier

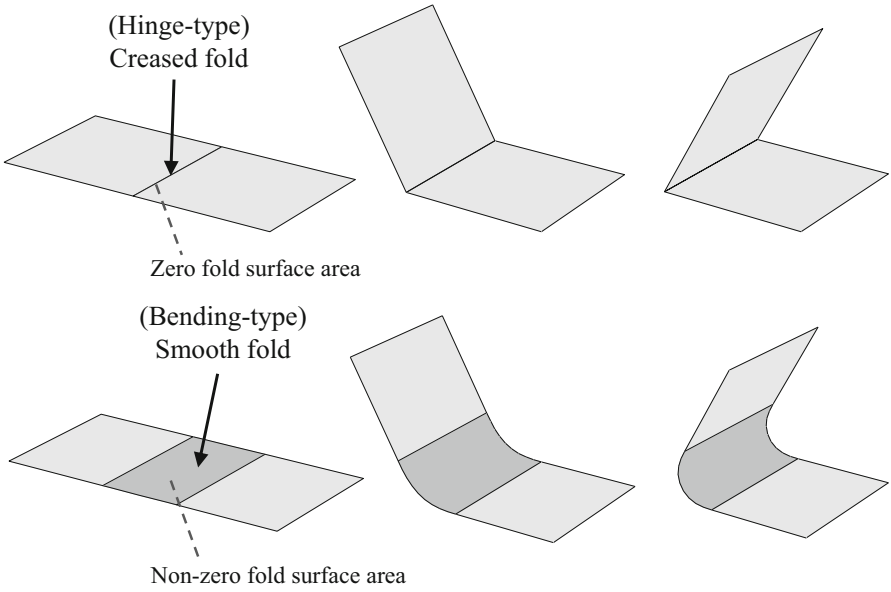


Fig. 1.13 Schematics of a sheet with a single creased fold and a sheet with a single smooth fold

An example of an origami structure with smooth folds is shown in Fig. 1.2. The distinction between creased folds and smooth folds is critical in the development of mathematical models and design methods for origami structures addressed in the subsequent chapters.

1.2 Active Origami Structures

In many applications of origami, the folding motion of the origami structure is driven by mechanical loads (forces and moments) applied externally. For example, in recreational or artistic origami where goal shapes are obtained by folding paper, the mechanical loads required to fold the paper are exerted by the fingers of the artist. For certain engineering applications of origami, however, it is impractical to externally apply the mechanical loads necessary to fold a given structure. This is the case for remote applications (e.g., space structures, underwater robotics, invasive biomedical devices) or structures at the nano- and micro-scales. In such circumstances, self-folding capabilities are essential [132]. A *self-folding* structure is one that has the capability of folding and/or unfolding without the external application of targeted mechanical loads [132, 133].

One approach to the development of self-folding structures is to leverage the use of *active materials*, which convert various forms of energy into mechanical work [68], to generate the desired folding behavior [132, 134, 135]. Such structures

are termed as *active origami structures* and are capable of folding and/or unfolding without the application of external mechanical loads but rather by the stimuli of a non-mechanical field (thermal, chemical, electromagnetic). We have already shown active origami structures in Figs. 1.2, 1.3, 1.7, and 1.9. In Sect. 1.2.1, we provide a brief description of the main active materials used in active origami structures. We then review existing active origami structures classified by their activation field (thermal, chemical, or electromagnetic) in Sect. 1.2.2.

1.2.1 Active Materials

As previously mentioned, *active materials* are those that convert various forms of energy (thermal, chemical, electromagnetic) into mechanical work [68]. This energy conversion is caused by the generation of inelastic strains and/or changes in stiffness or other material properties triggered by a non-mechanical stimulus [136]. The mechanical response exhibited by active materials under the stimulus of a non-mechanical field is typically one or more orders of magnitude larger than the response resulting from conventional material behavior such as thermal expansion.

Here, we are specifically interested in active materials capable of providing the mechanical response (i.e., forces/moments, displacements/rotations) necessary to fold a given origami structure under the application of a non-mechanical field. Extensive descriptions and mathematical models for various active materials are beyond the scope of this chapter and we refer the curious reader to other sources for such content [68, 137–140].

Active materials that provide mechanical response under *thermal* stimulus include *shape memory alloys* (SMAs) and *shape memory polymers* (SMPs). We also consider other thermomechanically coupled materials such as polymers that exhibit relatively large thermal expansion or changes in their stiffness due to changes in temperature. SMAs are metallic alloys that undergo solid-to-solid phase transformations induced by temperature and/or stress changes and during which they can recover seemingly permanent strains ($\approx$5%). SMAs possess the highest actuation energy density, which is the product of the actuation stress and the actuation strain, among all active materials [68]. Thus, SMAs can provide significant thermally induced deformations even if subject to large loads, which is a characteristic desired for a robust active origami structure (such as the origami stent shown in Fig. 1.9). SMPs are materials that also have the ability to thermally recover seemingly permanent large deformations under moderate mechanical loads. Although both SMPs and SMAs have the ability to recover apparently permanent deformations, SMPs possess the ability to recover extensions up to $\approx$100% [141–144]. Additionally, SMPs are inexpensive and lightweight as compared to SMAs; however, SMPs typically have stiffness and strength values two to three orders of magnitude lower than their metal counterparts. Considering these characteristics, SMPs present a viable solution for applications in origami structures that demand large deformations at reduced load levels (such as the foldable polyhedra and the

crane shown in Figs. 1.2 and 1.3 that fold only under their own weight). It is also noted that unlike SMAs, SMPs can only actuate once unless mechanical loads are applied to reset their initial configuration.

Active materials that provide mechanical response under *chemical* stimuli include polymers that exhibit swelling behavior when exposed to particular liquids (e.g., *hydrogel-based actuators* [145, 146]). These materials can change their shape (e.g., increase or decrease their volume) through absorption or desorption of liquids, which mimics the hydromorphic movement of plants, for which deformations such as bending are triggered by changes in the concentration of water in the cells and tissues [145]. Active materials that exhibit swelling behavior when exposed to particular liquids are ideal for origami applications in the biomedical field when the activation solution corresponds to a body fluid. However, force/moment generation capability is highly limited in these materials.

Examples of active materials that exhibit mechanical response under the application of *electromagnetic* stimulus include dielectric elastomers and magnetoactive elastomers. *Dielectric elastomers* are actually composites that are typically classified as electroactive polymers. They have the configuration of a capacitor, where an elastomer lamina is placed between two electrodes [147]. When a voltage is applied, the electrodes are drawn toward each other, the distance between the electrodes decreases, and the incompressible elastomer expands in the plane of the electrodes. *Magnetoactive elastomers* [148] are elastomeric materials containing magnetizable components embedded during their fabrication process. Active origami structures that make use of dielectric elastomers and magnetoactive elastomers can be folded or unfolded at larger frequencies than those that are thermally or chemically activated [149]. This is because temperature and species concentration typically cannot be manipulated at higher frequencies, which is possible with electric and magnetic fields.

1.2.2 Review of Active Origami Structures

Existing examples of active origami structures as classified according to the physical field inducing their folding motion are described in this section. The reader is provided with a systematic view of the state of the art in this important and evolving area, with thermal, chemical, and electromagnetic field-activated origami structures being presented. Tables 1.1, 1.2, 1.3 summarize the characteristics of the active origami structures considered. Examples of each type are classified in terms of their fold concept (creased or smooth; refer to Fig. 1.13), active material, characteristic sheet thickness, and current or potential applications.

Table 1.1 Examples of active origami structures folded via thermal stimuli

Fold type	Active material	Characteristic thickness (mm)	Application	References
Creased	SMA	5.0E –1	Multi-purpose morphing structures	[150, 151]
Creased	SMA	5.0E –2	Stent	[61, 152, 153]
Creased	SMA	1.0E –1	Robotics	[154, 155]
Creased	SMA	1.0E0	Robotics	[49, 60, 156]
Creased	SMP	1.0E0	Multi-purpose morphing structures	[6, 157–159]
Creased	SMA	1.0E0	Lifting structure	[160]
Creased/Smooth	SMA	2.0E0	Flexible mobile devices	[161]
Smooth	SMP	5.0E –1	Novel fabrication methods	[162]
Smooth	SMA	1.0E0	Multi-purpose morphing structures	[163–166]
Smooth	SMA	5.0E –1	Flexible mobile devices	[167]
Smooth	SMA	5.0E –1	Decorative paper structures	[168–170]
Smooth	SMA	2.5E –2	Microgripper	[171, 172]
Smooth	SMP	2.5E –1	Multi-purpose morphing structures	[5, 173, 174]
Smooth	SMA/SMP	1.0E0	Multi-purpose morphing structures	[175]
Smooth	Polymer[a]	1.0E –2	Multi-purpose morphing structures	[176]
Smooth	Polymer[b]	5.0E –3	Capture/release small scale devices	[177, 178]
Smooth	Polymer[b]	3.0E –1	Novel fabrication methods	[179]
Smooth	Polymer[b]	5.0E –2	Folding actuators	[180–182]

[a] Thermally induced changes in stiffness
[b] Thermal expansion

1.2.2.1 Thermally Activated Origami Structures

Table 1.1 provides a classification of various existing active origami structures that are thermally activated. The design space for thermally activated origami structures is large due to the various methods available for localized supply of heat that include convection, Joule heating, induction heating [210, 211], radiation [212, 213], etc. It is noted that various heating methods such as Joule heating and induction heating rely on the transformation of electromagnetic to thermal energy at the system level prior to the transformation of thermal to mechanical energy provided by the active material. Although there are several ways to alter the temperature in a structure, the

Table 1.2 Examples of active origami structures folded via chemical stimuli

Fold concept	Active material	Characteristic thickness (mm)	Application	References
Smooth	Swelling polymer	2.0E –1	Multi-purpose morphing structures	[183]
Smooth	Swelling polymer	5.0E –4	Capture/release small scale devices	[184–186]
Smooth	Swelling polymer	5.0E –4	Microtubes	[187–190]
Smooth	Swelling polymer	1.0E –1	Multi-purpose morphing structures	[191]
Smooth	Swelling polymer	2.0E –2	Drug release device	[192, 193]
Smooth	Swelling polymer	5.0E –2	Micro-containers	[194]
Smooth	Swelling polymer	5.0E –3	Multi-purpose morphing structures	[195]
Smooth	Swelling polymer	5.0E –3	Capture/release small scale devices	[196]
Smooth	Polymer degradation	1.0E –2	Small scale grippers	[197, 198]

Table 1.3 Examples of active origami structures folded via electromagnetic stimuli

Fold concept	Active material	Characteristic thickness (mm)	Application	References
Creased	Dielectric elastomer	3.0E –1	Robotics	[199]
Creased	Electroactive polymer	2.0E –2	Robotics	[200, 201]
Creased	Magnetoactive elastomer	3.0E0	Multi-purpose morphing structures	[202–204]
Creased	Magnetoactive elastomer	1.5E0	Bistable mechanisms	[149, 205, 206]
Creased	Magnetoactive elastomer	5.0E –1	Multi-purpose morphing structures	[207]
Smooth	Dielectric elastomer	2.0E0	Flexible mobile devices	[161]
Smooth	Dielectric elastomer	1.0E0	Robotics	[50]
Smooth	Dielectric elastomer	5.0E –1	Multi-purpose morphing structures	[204, 208, 209]

diffusive nature of heat represents a design challenge requiring the consideration of methods for controlling the spatial distribution of temperature over time, e.g., by adding thermal insulators to maintain high temperatures concentrated only in localized regions of the self-folding structure.

At the macro-scale, thermally activated self-folding structures have been fabricated mostly using SMAs and SMPs. Regarding SMAs, Rus and coworkers [150,

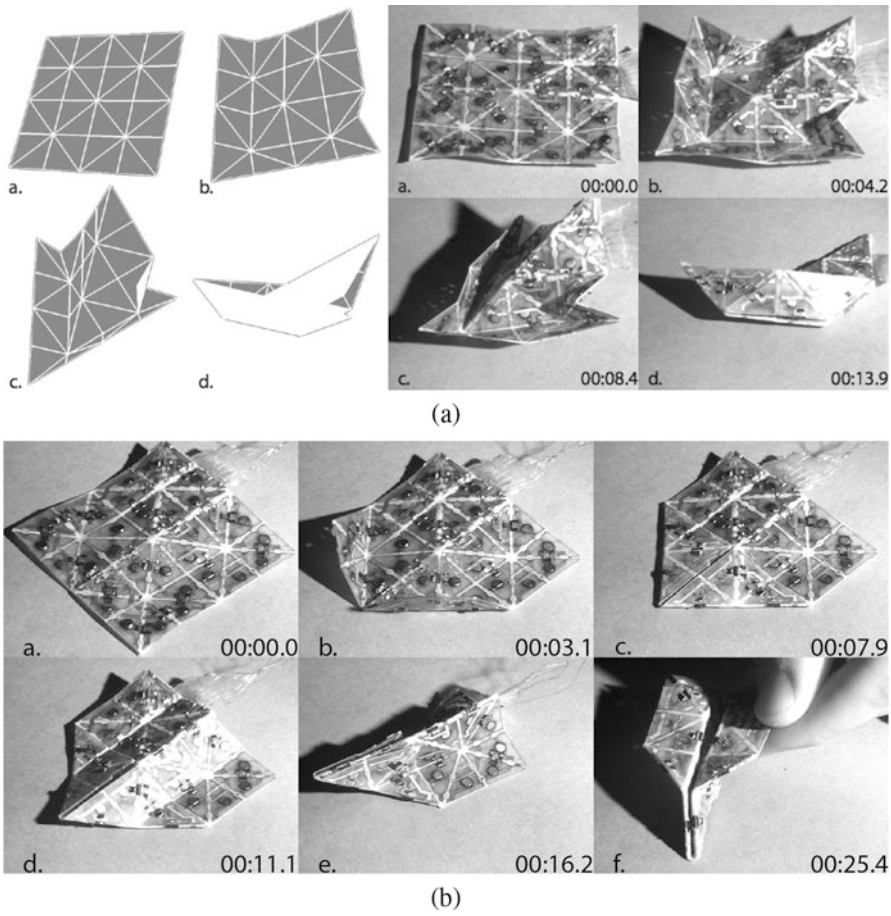


Fig. 1.14 Examples of active origami structures having SMA-actuated hinges [150]. (**a**) Simulation and experiment of a self-folding boat. (**b**) Experimental demonstration of a self-folding airplane. The numbers in the lower right corner of the pictures indicate the elapsed time (minutes:seconds). Reprinted from [150] with permission of the Proceedings of the National Academy of Sciences (PNAS). Credit: Elliot Hawkes

151, 214] developed an active origami structure consisting of a single sheet having repeated triangular tiles connected by hinges. The pattern of repeated triangular tiles provides flexibility regarding the various shapes towards which the sheet can fold. Thin SMA foils shape-set to a folded configuration were used to actuate the hinges. Such a structure was successful in demonstrations of folding towards different conventional origami shapes such as a boat and an airplane, shown in Fig. 1.14. An and Rus provided a design and programming guide for active origami structures of this kind [215].

Active origami structures having creased folds that make use of the actuation of torsional SMA wires have also been developed. An example is the origami lifting structure shown in Fig. 1.15a [160]. Another kind of SMA-based active origami structures makes use of the concept of massively foldable laminated sheets [163, 165, 216–219]. The concept consists of a laminated composite with two outer layers of SMA separated by a compliant and thermally insulating layer such as an elastomer. The outer layers of the SMA may consist of thin pre-strained SMA films [165, 220, 221] or meshes of pre-strained SMA wires [163, 222, 223]. It was shown through finite element simulations that structures comprised of this laminate are able to self-fold and form arbitrary three-dimensional structures [164]. An experimental demonstration of such self-folding structures is shown in Fig. 1.15b. Laminated composites having both SMA and SMP layers based on this concept have also been investigated. Such a shape memory composite is capable of locking in its folded shape upon a heating/cooling cycle [175].

Kuribayashi and coworkers [61, 152] addressed the design, manufacturing, and characterization of a self-deployable origami stent. The stent is comprised of a single SMA foil with pre-engineered folds. The deployment of this stent can be achieved by using the shape memory effect activated at body temperature or by making use of the pseudoelastic effect [68, 224]. It was demonstrated that the stent design successfully deploys as expected as shown in Fig. 1.9. Such a prototype is a valuable addition to the extensive list of existing and potential applications of SMAs in the biomedical field [225–227].

Active origami structures using SMA actuation have also been used for paper animation. Qi and coworkers [168, 169] used SMA wire actuators attached to paper in order to self-fold towards different artistic paper shapes. This study on SMA-actuated paper animation represents another contribution on the topic of active materials-based artwork [228, 229]. Active origami structures can also be potentially applied to other areas such as flexible display devices [230–236]. For example, Roudaut and coworkers [161] and Gomes and coworkers [167] proposed new flexible display designs for mobile devices that can alter their shape via fold-like deformations driven by SMA actuation.

Of course, SMAs are not the only option for thermally activated origami structures. Shape memory polymers [237] provide higher actuation strains at the cost of lower stresses. One example of thermally activated self-folding with SMPs is the work of Demaine and coworkers [6, 157] (see Fig. 1.3), who developed self-folding hinges comprised of SMP, paper, and resistive circuits.

Light sources have been used to provide heat to thermally activated origami structures based on the conversion of light to heat [238], where the thermal energy is subsequently converted to mechanical work by the active material. Liu and coworkers fabricated active origami structures that employ localized absorption of light cast over a compositionally homogeneous sheet of SMP. The uniform externally applied stimulus (i.e., diffuse light) generates a focused folding response via localized absorptivity [5]. Their approach uses mass-produced materials without the need for multiple fabrication steps, where the folded regions are defined by the presence of black ink patterned via a printing process. Examples of such active

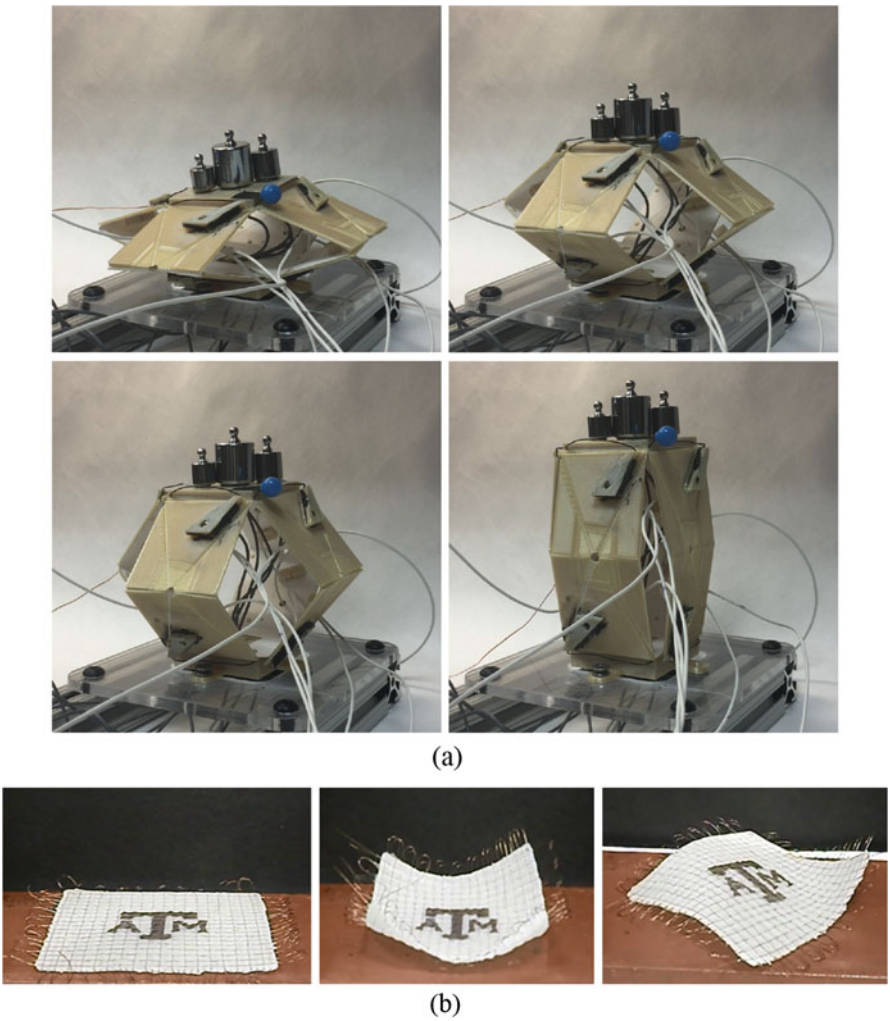


Fig. 1.15 Examples of active origami structures having SMA-actuated folds. (a) Origami lifting structure having hinge folds actuated via torsional SMA wires. *Credit: Leo Wood and Jaime Rendon Cavazos*. (b) Self-folding laminated sheet consisting of two outer layers comprised of SMA wire meshes separated by an insulating elastomer layer in which very smooth "folding" is exhibited. Credit: Aaron Powledge, Katherine Frei, and Anchal Goyal

origami structures are shown in Fig. 1.2. The polymer regions located beneath black inked areas heat faster than the areas elsewhere and eventually their temperature increases beyond the activation temperature of the SMP. After such a temperature is exceeded, these localized SMP regions actuate and the structure is folded. The reference planar configuration can be recovered by increasing the temperature of the entire sheet above the SMP activation temperature.

There are various examples of thermally activated origami structures at the small scales. For instance, Lee and coworkers designed a microgripper that uses SMA actuation to open and close [171, 172]. The outer layers of the microgrippers are comprised of deposited SMA films, which act as actuators. Applications for the microgripper include assembling small parts for manufacturing, minimally invasive tissue sampling, and remote handling of small particles in extreme environments [171].

As another example of thermally activated origami structures at the small scales, Gracias and coworkers fabricated self-folding structures at the micro-scale able to perform sequential folding through heating of pre-stressed hinges using lasers [176]. Their hinges were comprised of Cr/Au-polymer bilayers. The polymer layer softens under laser radiation and the bilayer bends due to existing pre-stress generated during the bilayer fabrication process. Kalaitzidou and coworkers [177, 178] also developed self-folding polymer-metal bilayer structures. Their self-folding laminated composites were comprised of a polydimethysiloxane (PDMS [239, 240]) layer and a gold layer. The PDMS layer had a thickness of several micrometers while the thickness of the gold layer was in the order of nanometers. Upon changes in temperature, the bilayer folds or unfolds due to dissimilar thermal expansion of the materials in the two layers. They also fabricated PDMS-silicon carbide (SiC) bilayers with similar behavior to demonstrate that their approach can be applied using any two materials with dissimilar thermal expansions. The ability of the bilayers to capture, transport, and release different solids was demonstrated indicating their potential application as delivery tubes [178].

1.2.2.2 Chemically Activated Origami Structures

Active origami structures driven by chemical stimulus have also been explored by multiple researchers. Most of these structures are based on folds comprised of laminated composites and utilize the degradation or swelling behavior of certain polymers under the presence of specific substances, sometimes based on their pH levels [241, 242] (see Fig. 1.16 for an example). Active origami structures driven by chemical stimuli are classified in Table 1.2. Active origami structures that make use of polymers exhibiting swelling behavior are able to act solely in liquid media, which limits their applications. Nevertheless, such materials represent a viable option in applications where other active materials such as thermally induced or electromagnetically induced varieties may not be feasible or desirable.

The microtubes fabricated by Kumar and coworkers [187] are examples of active origami structures folded through chemical stimulus. Kumar and coworkers considered a three-layer laminated composite comprised of PDMS, polystyrene (PS), and poly(4-vinylpyridine) (P4 VP). The folding mechanism of the microtubes was based on the different amount of swelling in their comprising polymer layers. Polystyrene demonstrates minimal water uptake while P4 VP is less hydrophobic and swells in acidic aqueous solutions [187]. A P4 VP layer increases its volume upon swelling and, if fixed to a PS layer, will cause the polymer laminate to fold. Micro- and

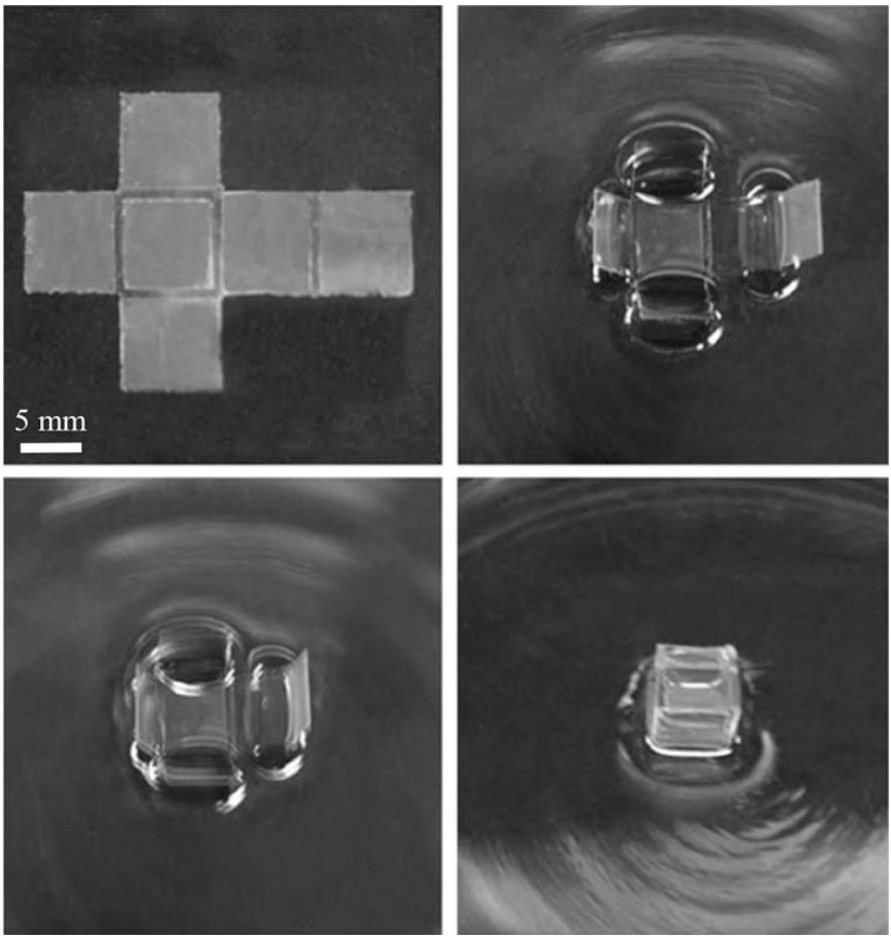


Fig. 1.16 Optical micrographs of an origami cube having hydrogel-actuated folds. Adapted from [183] with permission of The Royal Society of Chemistry

nano-tubes fabricated through this method are promising for applications including nano-syringes for intra-cellular surgery and nano-jet printing [243]. In another work, Shim and coworkers [194] fabricated robust microcarriers using hydrogel bilayers that exhibit reversible folding behavior. The bilayer composite consisted of a layer of poly(2-hydroxyethyl methacrylate-co-acrylic acid) (p(HEMA-co-AA)), and a layer of poly(2-hydroxyethyl methacrylate) (p(HEMA)). Planar structures comprised of this laminated composite were able to fold towards micro-containers by swelling of the p(HEMA-co-AA) layer triggered by changes in pH. A similar approach was adopted by He and coworkers in the fabrication of an oral delivery device [192]. The primary component of the device consisted of a finger-like bilayer comprised of pH-sensitive hydrogel based on crosslinked poly(methacrylic

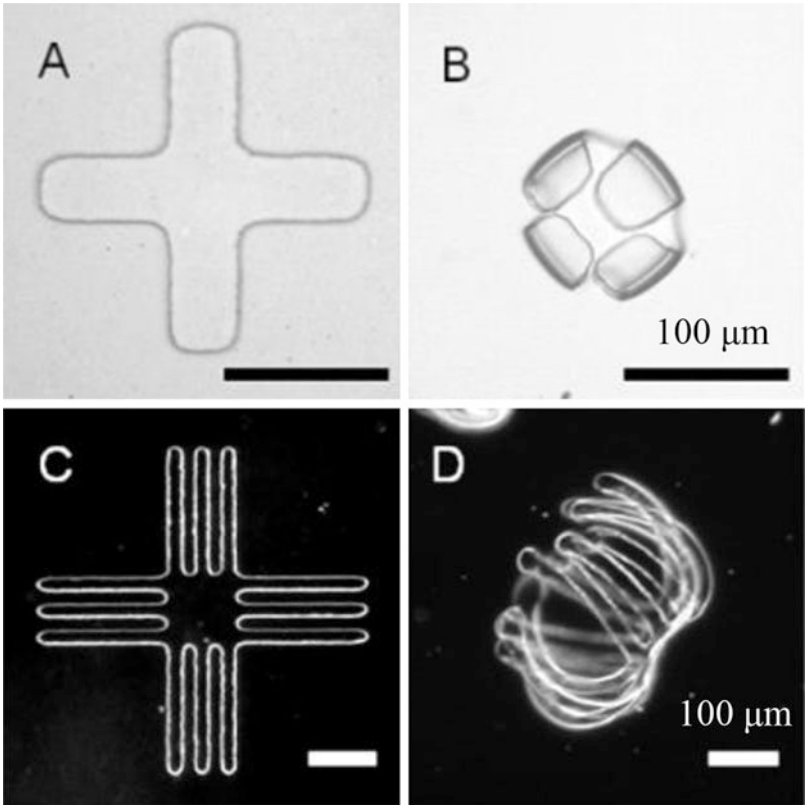


Fig. 1.17 Optical micrographs of active origami structures having folds actuated through polymer swelling. Adapted with permission from [195]. Copyright (2005) American Chemical Society

acid) (PMAA), which swells significantly when exposed to body fluids, and a second non-swelling layer. Guan and coworkers [195] studied the magnitude of folds achieved by a self-folding bilayer composite as a function of the active layer composition [195]. Two different micro-scale structures comprised of such a composite are shown in Fig. 1.17. In such studies, the active swelling layer was prepared with a mixture of poly(ethylene glycol methacrylate) (PEGMA) and poly-(ethylene glycol dimethacrylate) (PEGDMA). By controlling the ratio between the two components of such an active layer, different folding magnitudes were achieved.

A self-folding gripper that opens and closes by the actuation of polymer hinges was fabricated by Gracias and coworkers [197, 198]. The actuation of the polymer hinges was triggered by their sensitivity to the presence of enzymes, where they utilize two different polymers with two mutually exclusive enzyme sensitivities. The two polymers were placed at hinges in such a way that bending in opposite directions is activated given the appropriate stimulus. When one polymer is selectively degraded by its associated enzyme, the gripper closes. When the other

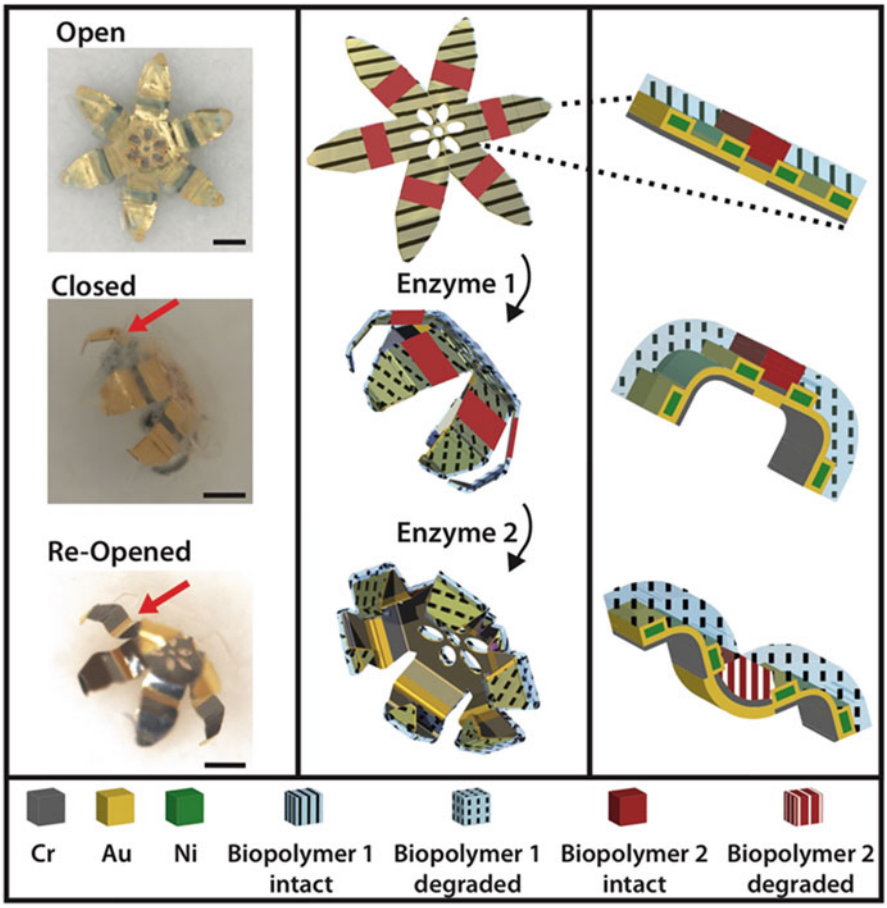


Fig. 1.18 Schematic of a self-folding microgripper that closes and reopens on exposure to enzymes. The scale bars represent 200 μm. Reproduced with permission from [197]. Copyright (2010) American Chemical Society. Credit: David Gracias

polymer is degraded through the action of its own distinct enzyme, its respective hinge bends and the gripper opens. This process is illustrated in Fig. 1.18.

1.2.2.3 Electromagnetically Activated Origami Structures

Examples of active origami structures driven by electromagnetic stimulus are provided in Table 1.3. Frecker and coworkers [204, 208, 209] developed a bending actuator that consisted of three layers: an active dielectric elastomer layer, a compliant passive layer, and compliant electrodes. The same group has also developed self-folding structures based on laminated composites having layers comprised of

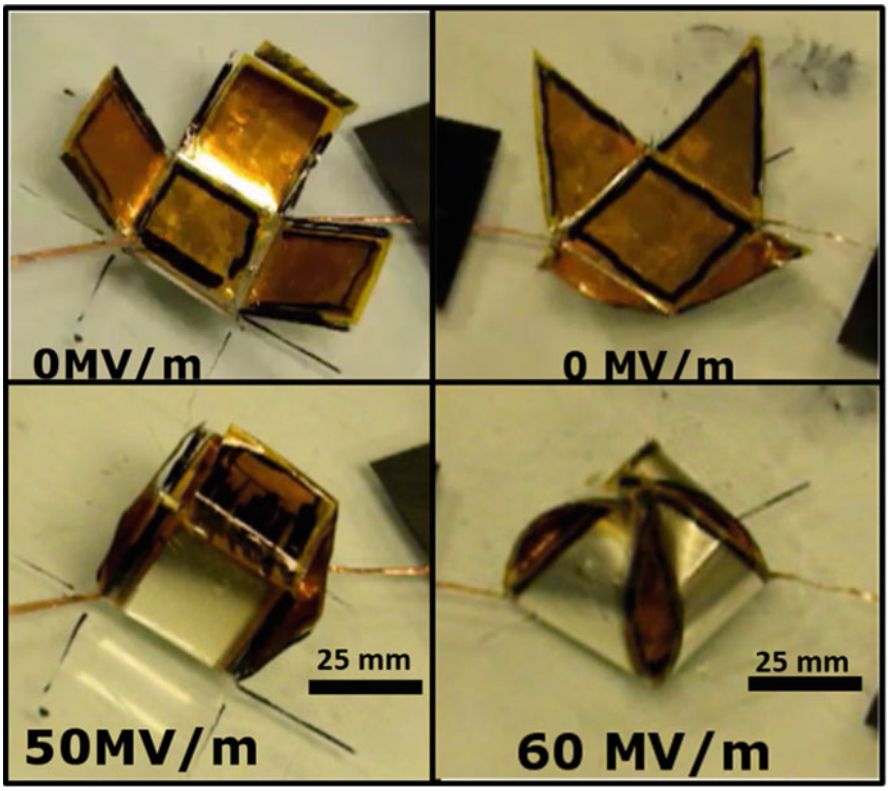


Fig. 1.19 Realization of a box and a square pyramid via self-folding using electroactive polymers. Reprinted from [244], Copyright (2017), with permission from Elsevier

electroactive polymers as shown in Fig. 1.19 [244]. Self-folding structures with dielectric elastomers were developed by Roudaut and coworkers for flexible mobile device displays. Their concept considered bending of a flexible mobile device through contraction and expansion of a dielectric elastomer actuator connected to two sides of the device [161].

More complex electrically driven folding motion was demonstrated by Okuzaki and coworkers, who created a biomorphic robot fabricated by folding a conducting polymer film [200, 201]. The folding actuation was generated by electrically induced changes in the stiffness of the polymer film. Okuzaki and coworkers demonstrated the feasibility of the concept by fabricating different prototypes such as an origami robot that moves rectilinearly with caterpillar-like motion. Such a motion was achieved by repeated expansion and contraction of its accordion-like body via folding/unfolding of electrically sensitive Polypyrrole (PPy [245, 246]).

Magnetoactive elastomers (MAEs [247]) have been investigated as actuators for self-folding structures by von Lockette and coworkers [202–204, 248]. The MAEs they considered were fabricated by mixing barium ferrite particles into a

silicone rubber matrix. Composites consisting of PDMS sheets with MAE patches were able to achieve locomotion under the application of a time-periodic magnetic field. Additionally, this team fabricated a cross-shaped PDMS sheet with four MAE patches on its sides that was shown to fold into a box, where translations and rotations were induced by a magnetic field. However, it should be noted that in this particular example, the MAE patches were used only for their ability to be magnetized and then drawn toward directions dictated by the applied magnetic field rather than for any intrinsic magneto-mechanical energy conversion. Bowen and coworkers utilized such a MAE-based self-folding approach to develop bistable origami mechanisms [149, 205, 206].

This review of active origami structures has demonstrated that the diversity in geometry, mechanisms, materials, and complexity of designs for such structures is large. Consequently, design and simulation methods for such structures are needed for engineers to make new developments in this area. The following sections present a review of efforts to enable design and simulation of origami structures that can be applied to both active origami structures and those without self-folding capabilities.

1.3 Origami Design

To realize a useful origami structure, designers must consider several issues beyond the choice of fold concept and material. The purpose of a design problem is to determine an engineered solution to fulfill one or more desired functions. For example, designers might achieve a storage/deployment functionality for satellite solar panels using an origami structure. Important considerations include determination of the final folded shape (or shapes, in the case of a reconfigurable structure), identification of a fold pattern that can achieve the desired shape(s), and scheduling of a folding sequence that results in the desired shape(s). Designers must achieve all of these subject to material failure constraints, applied energy and power limits, system-level failure criteria, and requirements for interfacing the structure with other parts of the system (e.g., mating points and surfaces). To this end, existing computer-aided design (CAD) and finite element analysis (FEA) tools can be used directly or be extended for use in the analysis and design of single folds. However, the challenges of identifying fold patterns and folding sequences require unique tools and methods. A review of the prior work on origami design methods in support of these design activities is provided in this section.

Creating an origami structure having desired characteristics, particularly a desired shape, is known as *origami design* [4]. Origami design is a challenge encountered not only by origami artists but also by designers and engineers who apply origami toward the solution of various engineering problems. Before the extensive interest exhibited by the mathematics, science, and engineering communities, most origami design was performed through trial and error or other heuristic approaches based on the intuition of an artist or designer [249]. With the increase in complexity of origami shapes and rising importance of design

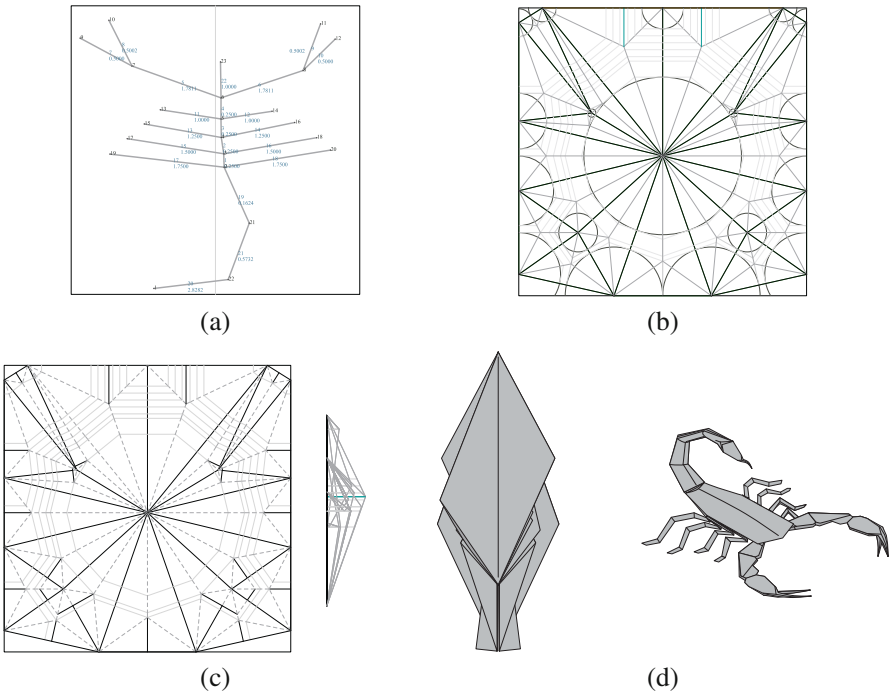


Fig. 1.20 Application of the tree method using TreeMaker (http://www.langorigami.com/article/treemaker). (**a**) Tree graph for a scorpion. Each line segment corresponds to a flap; desired flap lengths are assigned to each. (**b**) TreeMaker solves the equations corresponding to a packing of "circles" (that represent leaves of the tree graph) and "rivers" (that represent branches of the graph). Creases that overlay the circle/river packing are constructed. (**c**) The layer-ordering problem is solved, which allows one to create an assignment of mountain (solid black), valley (dashed gray), and unfolded (light gray) creases. TreeMaker also computes the silhouette of the desired base. (**d**) The folded base, which includes all of the desired flaps at their specified lengths, and the finished "Scorpion varileg," after additional shaping folds. Credit: Robert Lang

constraints that characterize application development, theoretical and computational approaches for origami design have become essential for developments in this area of study [1, 132, 249]. Current methods for origami design generally consider rigid faces and creased folds [10, 250], and although limited, they provide useful preliminary options for addressing the design of more general origami structures.

One of the most well-known approaches for origami design is Lang's *tree method* [249, 251–253], which is illustrated in Fig. 1.20. This method has been implemented in a software package named *TreeMaker* [254] and generates a pattern of creased folds on a squared sheet that allows for the folding of the sheet into a *base*, a folded shape whose projection to a plane is the tree line graph of the goal shape. This planar tree line graph may have arbitrary edge lengths and topology. After the base is folded, it is left to the designer to execute additional folds in order to closely approximate the goal shape.

Demaine and coworkers proposed an origami design method for goal shapes represented as polygons or three-dimensional polyhedral surfaces [1, 255]. The method is based on folding a sheet into a thin strip and then *wrapping* the strip around the goal shape using creased folds. Various algorithms for wrapping the goal shape were proposed [256], including one that uses any sheet area arbitrarily close to the goal shape area and another that maximizes the width of the strip subject to certain constraints.

Fuchi and coworkers [257] proposed an origami design method based on topology optimization [258]. In their approach, the location of a predefined set of fold lines is assumed; this given configuration is termed the "ground structure." The fold angles are the design variables. A topology optimization method is then used in an iterative effort to find the optimal set of fold angles that, combined with the ground structure, provide a targeted folded geometry.

A method for determining the geometry and pattern of creased folds associated with a planar sheet that can fold towards a goal polyhedral surface is termed *unfolding polyhedra* [1, 259–261]. An *unfolding* is defined as the flattening of a goal polyhedral surface to a plane such that the surface becomes a planar polygon having boundary segments that correspond to cuts made on the goal polyhedral surface [1]. Generally, the unfolding must be a single simply connected polygon having no overlaps and the cuts must be made on the edges of the goal polyhedral surface. This method is addressed in detail in Chap. 3 for origami with creased folds and in Chap. 6 for origami with smooth folds.

At the present time, the computational method for origami design applicable to the widest range of goal shapes was introduced by Tachi in [4, 262]. Tachi presented a method for obtaining a pattern of creased folds in a convex planar sheet that folds into an arbitrary three-dimensional goal shape if represented as a polygonal mesh [4, 262]. This method has been implemented in a software package named *Origamizer*, which is illustrated in Fig. 1.21, and is based on the introduction of regions having two rigid faces and three creased folds placed between any two faces of the polygonal mesh connected by an interior edge. The creased folds are used to tuck-fold such introduced regions to form the three-dimensional polygonal mesh starting from a planar configuration [262]. This method has been proven successful on goal polygonal meshes (convex and non-convex) of various complexities in terms of number of faces and non-regular connectivity. Such a method is denoted in this book as the *tuck-folding method* and is addressed in detail in Chap. 4 for origami with creased folds and in Chap. 7 for origami with smooth folds.

There are a number of works addressing the design of three-dimensional origami structures considering faces of non-zero thickness but assuming creased folds [75, 264–266]. Zirbel and coworkers [39, 267, 268] addressed the accommodation of thickness in origami deployable arrays, motivated by the need to fold thick panels that cannot bend during stowing or deployment (e.g., solar panels comprised of brittle materials). In their work, they propose a method for the modification of fold patterns originally intended for zero-thickness origami structures to accommodate thick faces in the context of a deployable space structure, and they demonstrate the applicability of their approach using a $1/20^{\text{th}}$ scale prototype of a deployable

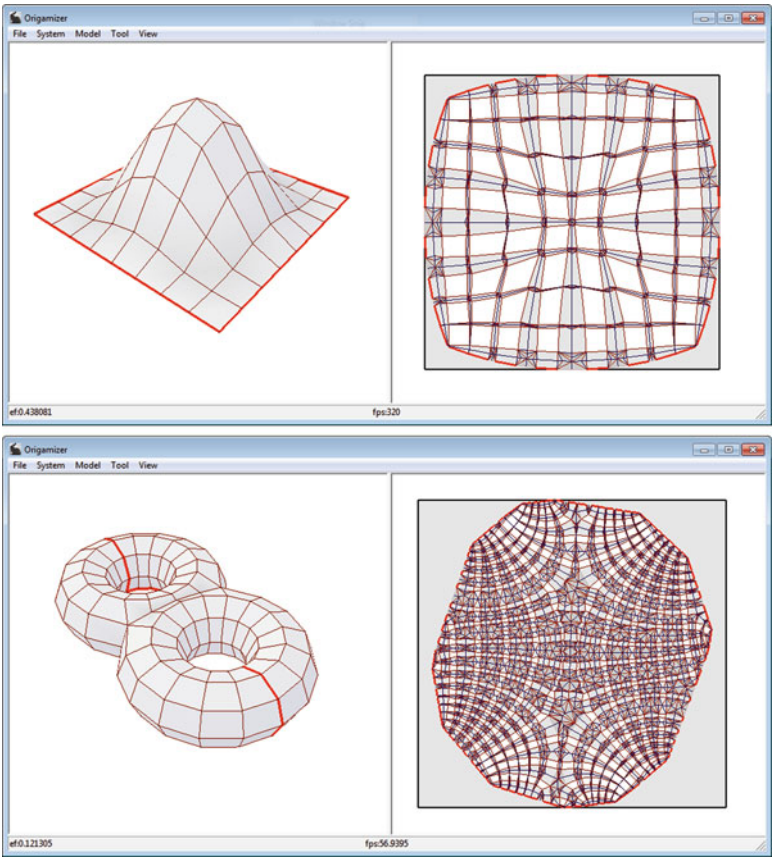


Fig. 1.21 Graphical user interface of Origamizer showing patterns of creased folds generated for two different goal shapes (left: goal shape, right: planar sheet with fold pattern). The Origamizer software [4, 263] by Tomohiro Tachi was obtained from http://www.tsg.ne.jp/TT/software/

solar array for space applications. In another work, Ku and Demaine presented an algorithm that transforms certain flat-foldable fold patterns assuming zero-thickness faces into fold patterns with similar folded configurations but having non-zero thickness faces [269, 270].

Although the preceding design methods (and more recent alternatives [10, 271–277]) allow for the design of fold patterns for goal shapes of various complexities, all assume only creased folds. Such a simplification may not be appropriate for structures having non-negligible thickness or those constructed from materials incapable of the strain magnitudes required for a creased idealization (e.g., metals, glassy polymers, active materials).

Akleman and coworkers [164] proposed an extension of the previously described unfolding polyhedra method for the consideration of smooth folds. Their main contribution was the process for in-surface thickening of the folds, which are

creased line folds in conventional unfolding polyhedra, to smooth folds having non-zero surface area. An algorithm to subdivide each obtained unfolding into smaller quadrilaterals and triangles to generate a finite element mesh that allows for high-fidelity analysis of the folding (or self-folding) process was also presented. The feasibility of such an origami design method was tested through simulation of the SMA-driven self-folding behavior of the generated unfoldings towards their associated polyhedral goal shapes [164]. In Chap. 8, we extend the ideas of [164] to address the design of active origami structures having self-folding capabilities. There, we propose extensions for the unfolding polyhedra method (Chap. 6) and the tuck-folding method (Chap. 7) to consider folds having active material actuators.

1.4 Simulation and Visualization of Origami Structures

Understanding and modeling the kinematics of origami structures is an active research topic investigated by various researchers exploring the simulation and visualization of origami structures [250, 278–284]. Kinematic modeling and computational simulation of origami structures [278, 285] permits understanding of their deformation and the development of additional computational tools for their design [286–288]. A kinematic model for origami structures must consider [1, 289, 290]: (i) The geometric definition of the sheet reference configuration and the associated fold pattern, (ii) the description of allowable deformations for the sheet, and (iii) the mapping between the reference and the current configurations of the sheet. These aspects of kinematic modeling for origami structures are rigorously addressed in Chaps. 2 and 5, while a brief literature review is presented in this section.

Belcastro and Hull [289, 290] presented a model for origami that mathematically represented creased folding deformations using affine transformations. In their model, Belcastro and Hull proposed constraints for the fold angles to allow only valid folded configurations. Tachi developed the *Rigid Origami Simulator* [278, 291] for the simulation of origami that also considers a set of constraints on the fold angles analogous to those presented in [289, 290]. Figure 1.22 shows the folding motion of an origami sheet simulated using the Rigid Origami Simulator. Applying a similar approach, Tachi also developed *Freeform Origami* [292] for the simulation and design of freeform origami structures represented as triangulated meshes [293]. Figure 1.23 shows the folding motion of an origami sheet simulated using Freeform Origami. Also using a kinematic model based on affine transformations, Ida and coworkers developed a software for interactive visualization and manipulation of origami called *Eos* [279, 294–296].

Alternatively, structural truss representations [297–300] have been used for kinematic modeling of origami, wherein the faces of the sheet are triangulated, each fold and boundary edge end-point is represented by a truss joint, and each fold and boundary edge is represented by a truss member. The configurations for which the

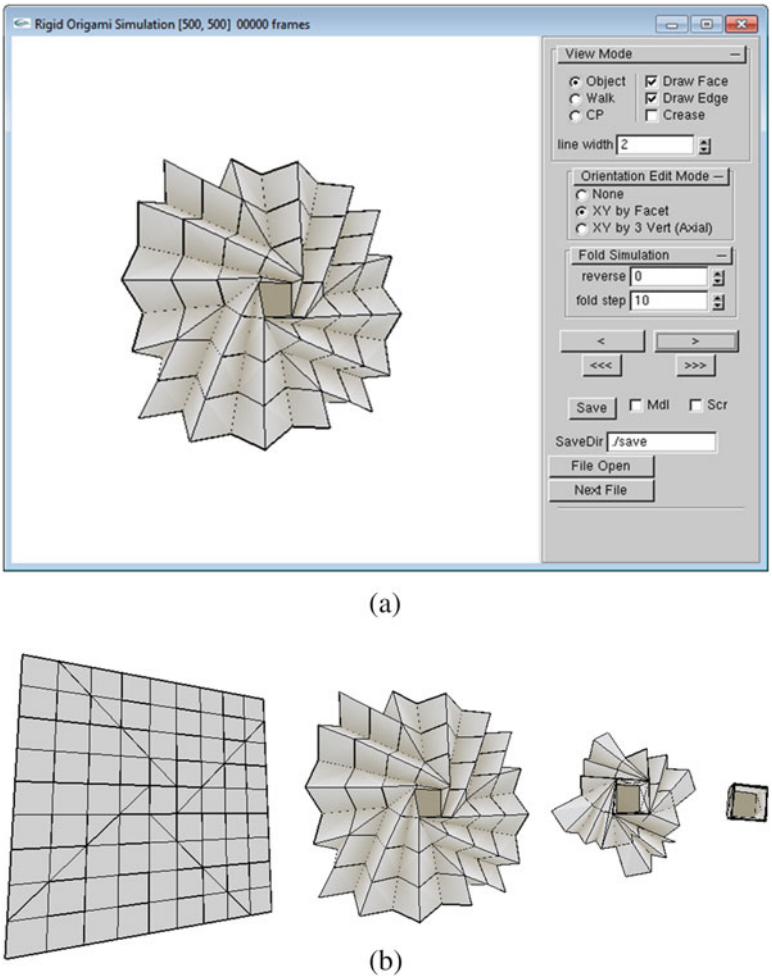


Fig. 1.22 Simulation of an origami flasher using the Rigid Origami Simulator [278]: (**a**) Sheet visualized in the Rigid Origami Simulator graphical user interface; (**b**) Various configurations attained by the sheet. The Rigid Origami Simulator software [278] by Tomohiro Tachi was obtained from http://www.tsg.ne.jp/TT/software/

displacements of the truss joints do not cause elongations of the truss members represent valid configurations.

Modeling the mechanics of origami structures is important in the engineering community and is especially applicable to the discussion of active origami in that it explicitly provides information regarding design and operational constraints. It accounts for the constitutive behavior of the materials that comprise the folding structures and the influence of the external environment on the resulting folding response. Modeling the structural mechanics of origami is needed for the develop-

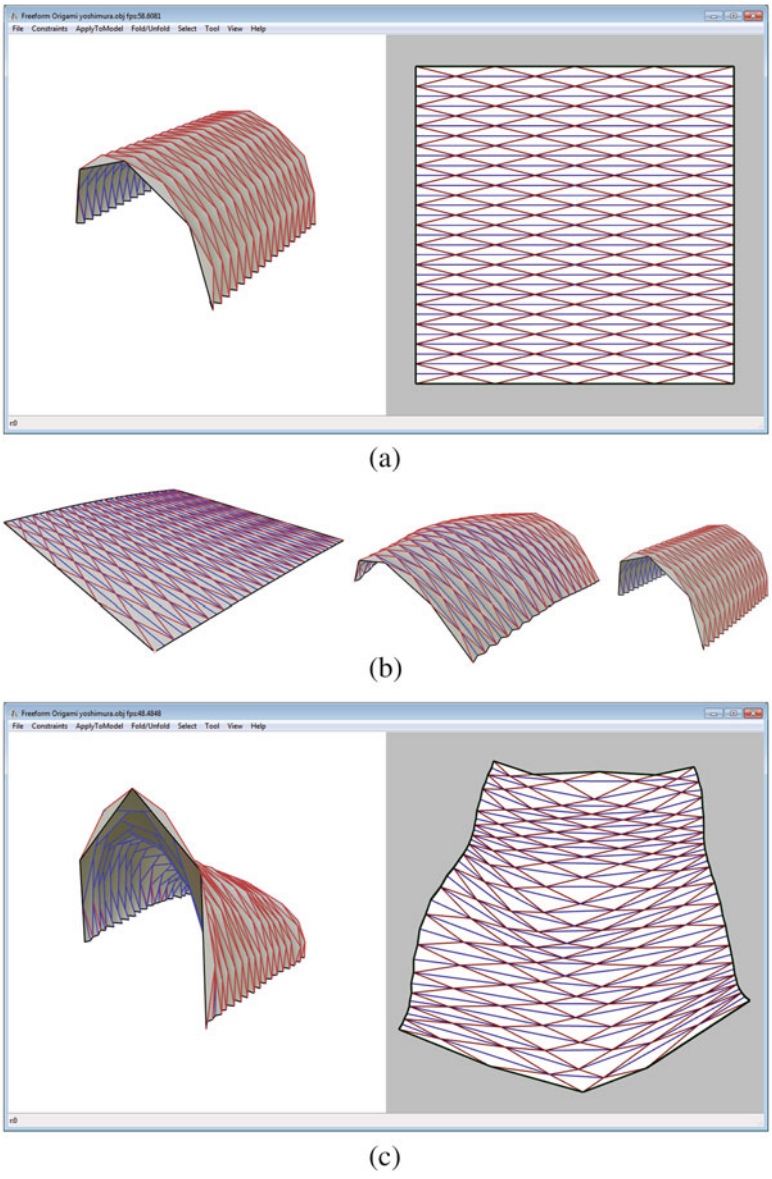


Fig. 1.23 Simulation of an origami sheet with the *Yoshimura* fold pattern using Freeform Origami [293]: (**a**) Sheet visualized in the Freeform Origami graphical user interface; (**b**) Various configurations attained by the sheet; (**c**) A user-modified Yoshimura pattern and its associated folded form. The Freeform Origami software [293] by Tomohiro Tachi was obtained from http://www.tsg.ne.jp/TT/software/

ment of origami-based applications ranging from those of structures comprised of passive materials [103, 110, 301–306] to the more complex structures having active material actuation, which are reviewed in Sect. 1.2.

Schenk and Guest [297] proposed a model for origami structures with elastic creased folds based on structural truss representations, where each fold or boundary edge end-point is represented by a truss joint, and each fold and boundary edge is represented by a truss member. Their model introduces torsional spring behavior to represent the creased folds and their resistance to folding deformation. Qiu and coworkers adopted such a modeling approach for the numerical analysis of origami carton-type packages [307, 308] and their model was validated against experimental data.

Existing FEA modeling approaches [309, 310] can be used directly or be extended for high-fidelity simulation of origami structures. There are various advantages provided by FEA approaches such as the capability of modeling structures having arbitrary geometry, materials, and boundary conditions. Structural finite elements developed on the basis of plate and shell theories [311–313] provide an option for modeling origami structures, which in general have geometries that are well represented as surfaces. Furthermore, available FEA software packages readily allow for multi-physical simulation, which is needed when considering active self-folding structures enabled by couplings between various physical fields. Origami structures composed of passive materials exhibiting elastic and elasto-plastic behavior have been previously studied using FEA. Examples include boxes [100, 101, 314] and beams [304, 306, 315, 316] with pre-engineered fold patterns for energy absorption in transportation systems, and lightweight foldcore composites for applications in the aerospace field [303, 317].

Finite element analysis has more recently been applied to self-folding origami structures comprised of various active materials [318]. For example, the folding response of SMA/elastomer composite sheets has been explored using both continuum finite elements [163, 319] and shell structural elements [320]. Ahmed and coworkers performed multi-physics simulation of origami structures actuated using dielectric elastomers and MAEs [204]. The aforementioned examples demonstrate the capabilities of FEA approaches to provide high-fidelity simulation of active origami structures considering couplings among various physical fields (e.g., mechanical-thermal [163, 319] and mechanical-electrical-magnetic [204]).

Although FEA approaches provide significant advantages in terms of fidelity and generality, they also have drawbacks when applied to the modeling of origami structures. First, the mathematical insights provided in the classical approaches of origami (e.g., geometric constructions [282]) are lost in the complexity and generality of FEA. The kinematic variables associated with FEA models (e.g., displacements and/or virtual rotations at nodes in displacement-driven FEA) and the very high number of degrees of freedom are not generally compatible with those of conventional origami (e.g., fold angles defined, one per fold). Moreover, FEA is not as computationally efficient as the alternatives, making it infeasible for modeling origami structures having high complexity in terms of the number of folds and/or the length of the folding sequence. In Chap. 8, we present a model for the

structural mechanics of origami bodies that takes into account kinematic variables compatible with those of conventional origami such as fold angles and that is also computationally efficient.

Chapter Summary

This chapter introduced the main concepts and applications of origami structures with focus on those having active material actuation, denoted here as *active origami structures*. In Sect. 1.1, we presented the basic concepts and applications of origami structures in general. It was observed that the folds in origami structures can be either creased (hinge-type) or smooth (bending-type) as illustrated in Fig. 1.13. In Sect. 1.2, we reviewed existing examples of active origami structures classified by their activation field. Finally, we briefly reviewed current design and simulation methods for origami structures in Sects. 1.3 and 1.4, respectively. For the remainder of this book, we present in great detail models and design methods for origami structures having both creased folds and smooth folds.

Problems

1.1 Refer to the origami flasher shown in Fig. 21 of [277]. Highlight the folds and the faces in such a figure and determine the number of folds and faces in the origami flasher.

1.2 Refer to the four active origami structures shown in Fig. 1 of [157]. Highlight the folds and the faces of each structure shown in such a figure and determine the number of folds and faces for each structure.

1.3 Refer to the active origami robots shown in Fig. 1 of [48]. Highlight the folds and the faces of each structure shown in such a figure and determine the number of folds and faces for each structure.

1.4 Summarize the physical mechanism(s) that allow the following active materials to convert non-mechanical energy into mechanical work and list five of their applications: (a) SMAs, (b) SMPs, (c) hydrogel-based actuators, (d) dielectric elastomers, and (e) magnetoactive elastomers.

1.5 Using the Origamizer software by Tomohiro Tachi (http://www.tsg.ne.jp/TT/software/), determine the fold pattern that can be used to fold a planar sheet towards the shapes illustrated in Fig. 1.24

1.6 Using the Freeform Origami software by Tomohiro Tachi (http://www.tsg.ne.jp/TT/software/), simulate the folding motion of origami sheets having the *Miura-Ori* fold pattern and the *waterbomb* fold pattern (shown in Fig. 1.25). Also use

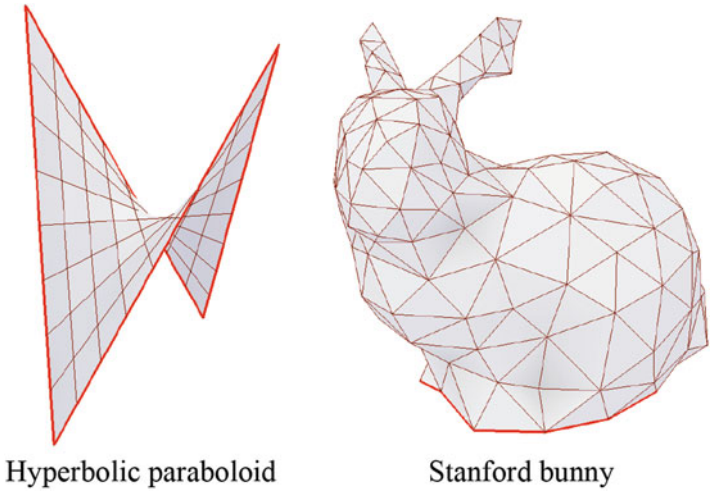


Fig. 1.24 Goal shapes of a hyperbolic paraboloid and the Stanford bunny [321]. These goal shapes can be obtained from the sample files of the Origamizer software [4, 263] by Tomohiro Tachi at http://www.tsg.ne.jp/TT/software/

Freeform Origami to modify these fold patterns and to simulate the folding motion of the origami sheets with the modified fold patterns.

1.7 Using the Rigid Origami Simulator by Tomohiro Tachi (http://www.tsg.ne.jp/TT/software/), simulate the folding motion of the origami flasher shown in Fig. 1.22. Also simulate the folding motion of other fold patterns available in the sample files of the Rigid Origami Simulator

1.8 Using the TreeMaker software by Robert Lang (http://www.langorigami.com/article/treemaker), determine the fold patterns for the tree line graphs of two insects (your choice).

1.9 Refer to the active origami structures presented in [162]. Identify the following: (a) the active material that drives the folding motion of such structures, (b) the characteristic thickness of the active origami structures, and (c) the activation field (e.g., thermal, chemical, electromagnetic) and the methodology used to alter such an activation field.

1.10 Repeat Problem 1.9 considering the active origami structures presented in [23].

1.11 Repeat Problem 1.9 considering the active origami structures presented in [86].

1.12 Repeat Problem 1.9 considering the active origami structure presented in Fig. 1 of [322].

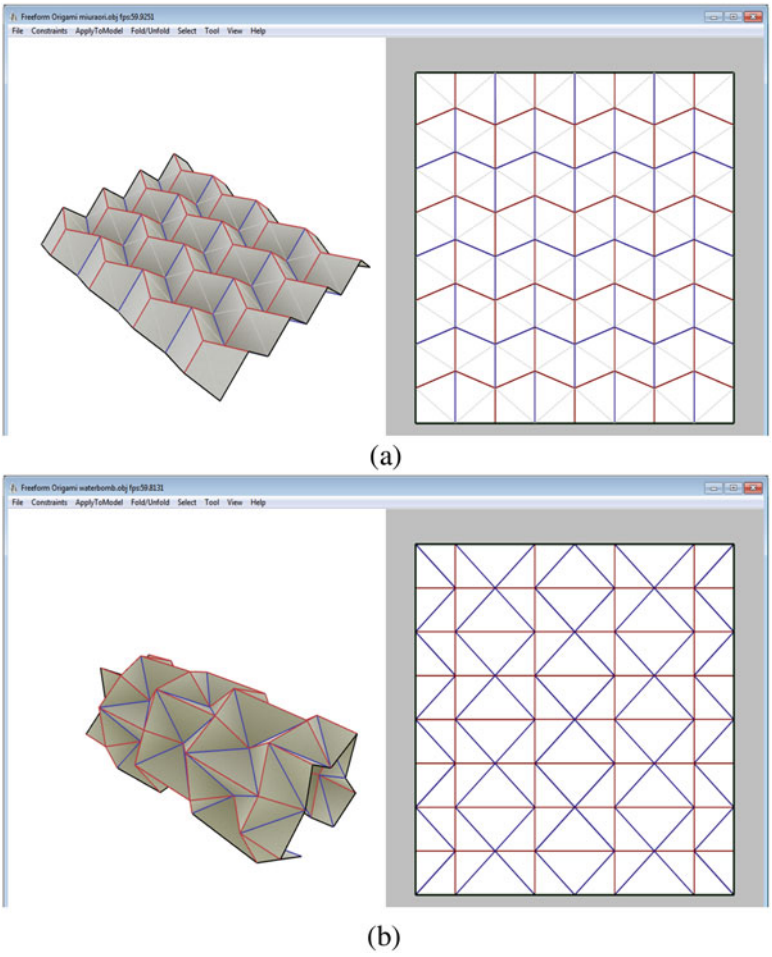


Fig. 1.25 Origami sheets having the (**a**) *Miura-Ori* fold pattern and the (**b**) *waterbomb* fold pattern. These fold patterns can be obtained from the sample files of the Freeform Origami software by Tomohiro Tachi at http://www.tsg.ne.jp/TT/software/

1.13 Repeat Problem 1.9 considering the active origami structures presented in [323].

1.14 Repeat Problem 1.9 considering the active origami structures presented in [324].

References

1. E.D. Demaine, J. O'Rourke, *Geometric Folding Algorithms* (Cambridge University Press, Cambridge, 2007)
2. R.J. Lang, The science of origami. Phys. World **20**(2), 30–31 (2007)
3. R.J. Lang, Origami: complexity in creases (again). Eng. Sci. **67**(1), 5–19 (2004)
4. T. Tachi, Origamizing polyhedral surfaces. IEEE Trans. Vis. Comput. Graph. **16**(2), 298–311 (2010)
5. Y. Liu, J.K. Boyles, J. Genzer, M.D. Dickey, Self-folding of polymer sheets using local light absorption. Soft Matter **8**(6), 1764–1769 (2012)
6. S.M. Felton, M.T. Tolley, B.H. Shin, C.D. Onal, E.D. Demaine, D. Rus, R. Wood, Self-folding with shape memory composites. Soft Matter **9**(32), 7688–7694 (2013)
7. L.J. Fei, D. Sujan, Origami theory and its applications: a literature review. Int. J. Soc. Hum. Sci. Eng. **7**(1), 113–117 (2013)
8. B.A. Cipra, In the fold: origami meets mathematics. SIAM News **34**(8), 1–4 (2001)
9. T. Tarnai, Origami in structural engineering, in *Proceedings of the IASS Symposium 2001: International Symposium on Theory, Design and Realization of Shell and Spatial Structures*, Nagoya, Japan, 9–13 Oct 2001, pp. 298–299
10. X. Zhou, H. Wang, Z. You, Design of three-dimensional origami structures based on a vertex approach. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. **471**(2181), 20150407 (2015)
11. D. Dureisseix, An overview of mechanisms and patterns with origami. Int. J. Space Struct. **27**(1), 1–14 (2012)
12. E.D. Demaine, M.L. Demaine, V. Hart, G.N. Price, T. Tachi, (Non)existence of pleated folds: how paper folds between creases. Graphs Comb. **27**(3), 377–397 (2011)
13. C. Cromvik, K. Eriksson, Airbag folding based on origami mathematics, in *Origami 4: Fourth International Meeting of Origami Science, Mathematics, and Education*, 2006, pp. 129–139
14. R. Hoffman, Airbag folding: origami design applied to an engineering problem, in *Third International Meeting of Origami Science Math and Education*, Asilomar, CA, 2001
15. S. Gray, N. Zeichner, V. Kumar, M. Yim, A simulator for origami-inspired self-reconfigurable robots, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (CRC Press, Boca Raton, 2011), pp. 323–333
16. W. Gao, K. Ramani, R. J. Cipra, T. Siegmund, Kinetogami: a reconfigurable, combinatorial, and printable sheet folding. J. Mech. Des. **135**(11), 111009 (2013)
17. S. Pandey, E. Gultepe, D.H. Gracias. Origami inspired self-assembly of patterned and reconfigurable particles. J. Vis. Exp. (72), e50022 (2013)
18. N.S. Shaar, G. Barbastathis, C. Livermore, Integrated folding, alignment, and latching for reconfigurable origami microelectromechanical systems. J. Microelectromech. Syst. **24**(4), 1043–1051 (2015)
19. E.T. Filipov, G.H. Paulino, T. Tachi, Origami tubes with reconfigurable polygonal cross-sections. Proc. R. Soc. A Math. Phys. Eng. Sci. **472**(2185), 20150607 (2016)
20. S. Yao, X. Liu, S.V. Georgakopoulos, M.M. Tentzeris, A novel reconfigurable origami spring antenna, in *Proceedings of the 2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)* (IEEE, Piscataway, 2014), pp. 374–375
21. Z. You, Folding structures out of flat materials. Science **345**(6197), 623–624 (2014)
22. C.D. Saintsing, B.S. Cook, M.M. Tentzeris, An origami inspired reconfigurable spiral antenna, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2014-35353 (American Society of Mechanical Engineers, New York, 2014), p. V05BT08A050
23. N. Bassik, G.M. Stern, D.H. Gracias, Microassembly based on hands free origami with bidirectional curvature. Appl. Phys. Lett. **95**(9), 091901 (2009)

24. S. Mueller, B. Kruck, P. Baudisch, LaserOrigami: laser-cutting 3D objects, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, 2013), pp. 2585–2592
25. B. Shin, S.M. Felton, M.T. Tolley, R.J. Wood, Self-assembling sensors for printable machines, in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2014), pp. 4417–4422
26. J. Morgan, S.P. Magleby, R.J. Lang, L.L. Howell, A preliminary process for understanding origami-adapted design, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/-CIE*, Paper No. DETC2015-47559 (American Society of Mechanical Engineers, New York, 2015), p. V05BT08A053
27. B. An, S. Miyashita, M.T. Tolley, D.M. Aukes, L. Meeker, E.D. Demaine, M.L. Demaine, R.J. Wood, D. Rus, An end-to-end approach to making self-folded 3D surface shapes by uniform heating, in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2014), pp. 1466–1473
28. A. Piqué, S.A. Mathews, N.A. Charipar, A.J. Birnbaum, Laser origami: a new technique for assembling 3D microstructures, in *Proceedings of SPIE*, vol. 8244 (International Society for Optics and Photonics, San Diego, 2012), p. 82440B-82440B-7
29. P.W.K. Rothemund, Folding DNA to create nanoscale shapes and patterns. Nature **440**(7082), 297–302 (2006)
30. T. Tørring, N.V. Voigt, J. Nangreave, H. Yan, K.V. Gothelf, DNA origami: a quantum leap for self-assembly of complex structures. Chem. Soc. Rev. **40**(12), 5636–5646 (2011)
31. J. Nangreave, D. Han, Y. Liu, H. Yan, DNA origami: a history and current perspective. Curr. Opin. Chem. Biol. **14**(5), 608–615 (2010)
32. A. Edwards, H. Yan, DNA origami, in *Nucleic Acid Nanotechnology* (Springer, Berlin, 2014), pp. 93–133
33. A.E. Marras, L. Zhou, H.-J. Su, C.E. Castro, Programmable motion of DNA origami mechanisms. Proc. Natl. Acad. Sci. **112**(3), 713–718 (2015)
34. C. Cao, Y. Feng, J. Zang, G.P. López, X. Zhao, Tunable lotus-leaf and rose-petal effects via graphene paper origami. Extreme Mech. Lett. **4**, 18–25 (2015)
35. K. Miura, Map fold a la Miura style, its physical characteristics and application to the space science, in *Research of Pattern Formation*, ed. by R. Takaki (KTK Scientific Publishers, Tokyo, 1994), pp. 77–90
36. R.J. Lang, Computational origami: from flapping birds to space telescopes, in *Proceedings of the Twenty-fifth Annual Symposium on Computational Geometry* (ACM, New York, 2009), pp. 159–162
37. L. Wilson, S. Pellegrino, R. Danner, Origami sunshield concepts for space telescopes, in *Proceedings of the 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2013, pp. 2013–1594
38. D. Pohl, W.D. Wolpert, Engineered spacecraft deployables influenced by nature. in *Proceedings of SPIE*, vol. 7424, 2009, p. 742408-742408-9
39. S.A. Zirbel, R.J. Lang, M.W. Thomson, D.A. Sigel, P.E. Walkemeyer, B.P. Trease, S.P. Magleby, L.L. Howell, Accommodating thickness in origami-based deployable arrays. J. Mech. Des. **135**(11), 111005 (2013)
40. Y. Jung, J. Kim, Flutter speed estimation for folding wing system, in *Proceedings of the 18th International Conference on Composite Materials*, 2011
41. M.P. Snyder, B. Sanders, F.E. Eastep, G.J. Frank, Vibration and flutter characteristics of a folding wing. J. Aircr. **46**(3), 791–799 (2009)
42. D.H. Lee, T.A. Weisshaar, Aeroelastic studies on a folding wing configuration, in *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference*, vol. 1996, 2005, pp. 1–13

43. G. Bunget, S. Seelecke, BATMAV: a biologically inspired micro air vehicle for flapping flight: kinematic modeling, in *Proceedings of the 15th International Symposium on Smart Structures and Materials & Nondestructive Evaluation and Health Monitoring* (International Society for Optics and Photonics, San Diego, 2008), p. 69282F
44. G. Bunget, S. Seelecke, BATMAV: a 2-DOF bio-inspired flapping flight platform, in *Proceedings of the SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring* (International Society for Optics and Photonics, San Diego, 2010), p. 76433B
45. P.E.I. Pounds, Paper plane: towards disposable low-cost folded cellulose-substrate UAVs, in *2012 Australasian Conference on Robotics and Automation* (Australian Robotics and Automation Association (ARAA), Sydney, 2012)
46. Q.-T. Truong, B.W. Argyoganendro, H.C. Park, Design and demonstration of insect mimicking foldable artificial wing using four-bar linkage systems. J. Bionic Eng. **11**(3), 449–458 (2014)
47. S.-M. Baek, D.-Y. Lee, K.-J. Cho, Curved compliant facet origami-based self-deployable gliding wing module for jump-gliding, in *Proceedings of the ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Paper No. DETC2016-60543 (American Society of Mechanical Engineers, New York, 2016), p. V05BT07A028
48. S. Miyashita, S. Guitron, M. Ludersdorfer, C.R. Sung, D. Rus, An untethered miniature origami robot that self-folds, walks, swims, and degrades, in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2015), pp. 1490–1496
49. D.-Y. Lee, J.-S. Kim, S.-R. Kim, J.-S. Koh, K.-J. Cho, The deformable wheel robot using magic-ball origami structure, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-13016 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A040
50. P.J. White, S. Latscha, S. Schlaefer, M. Yim, Dielectric elastomer bender actuator applied to modular robotics, in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Piscataway, 2011), pp. 408–413
51. A.M. Hoover, R.S. Fearing, Fast scale prototyping for folded millirobots, in *Proceedings of the 2008 IEEE International Conference on Robotics and Automation ICRA 2008* (IEEE, Pitscataway, 2008), pp. 886–892
52. E. Vander Hoff, D. Jeong, K. Lee, OrigamiBot-I: a thread-actuated origami robot for manipulation and locomotion, in *Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, Piscataway, 2014), pp. 1421–1426
53. Z. Zhakypov, M. Falahi, M. Shah, J. Paik, The design and control of the multi-modal locomotion origami robot, Tribot, in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Piscataway, 2015), pp. 4349–4355
54. C.D. Onal, M.T. Tolley, R.J. Wood, D. Rus, Origami-inspired printed robots. IEEE/ASME Trans. Mechatron. **20**(5), 2214–2221 (2015)
55. K. Zhang, C. Qiu, J.S. Dai, An extensible continuum robot with integrated origami parallel modules. J. Mech. Robot. **8**(3), 031010 (2016)
56. H. Shigemune, S. Maeda, Y. Hara, N. Hosoya, S. Hashimoto, Origami robot: a self-folding paper robot with an electrothermal actuator created by printing. IEEE/ASME Trans. Mechatron. **21**(6), 1–1 (2016)
57. R. Yan, M. Luo, Z. Wan, Y. Qin, J. Santoso, E. Skorina, C. Onal, OriSnake: design, fabrication and experimental analysis of a 3-D origami snake robot. IEEE Robot. Autom. Lett. **3**(3), 1993–1999 (2018)
58. A. Pagano, T. Yan, B. Chien, A. Wissa, S. Tawfick, A crawling robot driven by multi-stable origami. Smart Mater. Struct. **26**(9), 094007 (2017)

59. S. Felton, M. Tolley, E. Demaine, D. Rus, R. Wood, A method for building self-folding machines. Science **345**(6197), 644–646 (2014)
60. D.-Y. Lee, S.-R. Kim, J.-S. Kim, J.-J. Park, K.-J. Cho, Origami wheel transformer: a variable-diameter wheel drive robot using an origami structure. Soft Robot. **4**(2), 163–180 (2017)
61. K. Kuribayashi, K. Tsuchiya, Z. You, D. Tomus, M. Umemoto, T. Ito, M. Sasaki, Self-deployable origami stent grafts as a biomedical application of Ni-rich TiNi shape memory alloy foil. Mater. Sci. Eng. A **419**(1–2), 131–137 (2006)
62. C.L. Randall, E. Gultepe, D.H. Gracias, Self-folding devices and materials for biomedical applications. Trends Biotechnol. **30**(3), 138–146 (2012)
63. R. Fernandes, D.H. Gracias, Self-folding polymeric containers for encapsulation and delivery of drugs. Adv. Drug Deliv. Rev. **64**(14), 1579–1589 (2012)
64. Y. Wang, L. Ge, P. Wang, M. Yan, J. Yu, S. Ge, A three-dimensional origami-based immuno-biofuel cell for self-powered, low-cost, and sensitive point-of-care testing. Chem. Commun. **50**(16), 1947–1949 (2014)
65. K. Kuribayashi, S. Takeuchi, Foldable Parylene origami sheets covered with cells: toward applications in bio-implantable devices, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (CRC Press, Boca Raton, 2011), p. 385
66. B.J. Edmondson, L.A. Bowen, C.L. Grames, S.P. Magleby, L.L. Howell, T.C. Bateman, Oriceps: origami-inspired forceps, in *Proceedings of the ASME 2013 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, Paper No. SMASIS2013-3299 (American Society of Mechanical Engineers, New York, 2013), p. V001T01A027
67. A. Taylor, T. Slutzky, L. Feuerman, M. Fok, Z.T.H. Tse, Origami endoscope design for MRI-guided therapy, in *2017 Design of Medical Devices Conference*, Paper No. DMD2017-3352 (American Society of Mechanical Engineers, New York, 2017), p. V001T08A006
68. D.C. Lagoudas (ed.), *Shape Memory Alloys: Modeling and Engineering Applications* (Springer, New York, 2008)
69. A.P. Thrall, C.P. Quaglia, Accordion shelters: a historical review of origami-like deployable shelters developed by the US military. Eng. Struct. **59**, 686–692 (2014)
70. F.J. Martínez-Martín, A.P. Thrall, Honeycomb core sandwich panels for origami-inspired deployable shelters: multi-objective optimization for minimum weight and maximum energy efficiency. Eng. Struct. **69**, 158–167 (2014)
71. C.P. Quagli, Z.C. Ballard, A.P. Thrall, Parametric modelling of an air-liftable origami-inspired deployable shelter with a novel erection strategy. Mobile Rapidly Assembled Struct. IV **136**, 23 (2014)
72. C.P. Quaglia, N. Yu, A.P. Thrall, S. Paolucci, Balancing energy efficiency and structural performance through multi-objective shape optimization: case study of a rapidly deployable origami-inspired shelter. Energy Build. **82**, 733–745 (2014)
73. C.P. Quaglia, A.J. Dascanio, A.P. Thrall, Bascule shelters: a novel erection strategy for origami-inspired deployable structures. Eng. Struct. **75**, 276–287 (2014)
74. M.D. Tumbeva, Y. Wang, M.M. Sowar, A.J. Dascanio, A.P. Thrall, Quilt pattern inspired engineering: efficient manufacturing of shelter topologies. Autom. Constr. **63**, 57–65 (2016)
75. T. Tachi, Geometric considerations for the design of rigid origami structures, in *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*, vol. 12, 2010, pp. 458–460
76. T. Hollingshead, BYU engineers built a bulletproof origami shield to protect law enforcement. https://news.byu.edu/news/byu-researchers-built-bullet-proof-origami-shield-protect-law-enforcement
77. Y. Shi, F. Zhang, K. Nan, X. Wang, J. Wang, Y. Zhang, Y. Zhang, H. Luan, K.-C. Hwang, Y. Huang, J.A. Rogers, Y. Zhang, Plasticity-induced origami for assembly of three dimensional metallic structures guided by compressive buckling. Extreme Mech. Lett. **11**, 105–110 (2017)

78. Z. Yan, F. Zhang, J. Wang, F. Liu, X. Guo, K. Nan, Q. Lin, M. Gao, D. Xiao, Y. Shi, Y. Qiu, H. Luan, J.H. Kim, Y. Wang, H. Luo, M. Han, Y. Huang, Y. Zhang, J.A. Rogers, Controlled mechanical buckling for origami-inspired construction of 3D microstructures in advanced materials. Adv. Funct. Mater. **26**(16), 2629–2639 (2016)
79. Y. Zhang, Z. Yan, K. Nan, D. Xiao, Y. Liu, H. Luan, H. Fu, X. Wang, Q. Yang, J. Wang, W. Ren, H. Si, F. Liu, L. Yang, H. Li, J. Wang, X. Guo, H. Luo, L. Wang, Y. Huang, J.A. Rogers, A mechanically driven form of Kirigami as a route to 3D mesostructures in micro/nanomembranes. Proc. Natl. Acad. Sci. **112**(38), 11757–11764 (2015)
80. B. Saccà, C.M. Niemeyer, DNA origami: the art of folding DNA. Angew. Chem. Int. Ed. **51**(1), 58–66 (2012)
81. J.T. Early, R. Hyde, R.L. Baron, Twenty-meter space telescope based on diffractive Fresnel lens, in *Proceedings of the SPIE's 48th Annual Meeting, Optical Science and Technology* (International Society for Optics and Photonics, San Diego, 2004), pp. 148–156
82. J.M. Zanardi Ocampo, P.O. Vaccaro, K. Kubota, T. Fleischmann, T.-S. Wang, T. Aida, T. Ohnishi, A. Sugimura, R. Izumoto, M. Hosoda, et al. Characterization of GaAs-based micro-origami mirrors by optical actuation. Microelectron. Eng. **73**, 429–434 (2004)
83. Q. Cheng, Z. Song, T. Ma, B.B. Smith, R. Tang, H. Yu, H. Jiang, C.K. Chan, Folding paper-based lithium-ion batteries for higher areal energy densities. Nano Lett. **13**(10), 4969–4974 (2013)
84. Z. Song, T. Ma, R. Tang, Q. Cheng, X. Wang, D. Krishnaraju, R. Panat, C.K. Chan, H. Yu, H. Jiang, Origami lithium-ion batteries. Nat. Commun. **5**, Article no. 3140 (2014)
85. I. Nam, G.-P. Kim, S. Park, J.W. Han, J. Yi, All-solid-state, origami-type foldable supercapacitor chips with integrated series circuit analogues. Energy Environ. Sci. **7**(3), 1095–1102 (2014)
86. S. Miyashita, L. Meeker, M.T. Tolley, R.J. Wood, D. Rus, Self-folding miniature elastic electric devices. Smart Mater. Struct. **23**(9), 094005 (2014)
87. R. Tang, H. Huang, H. Tu, H. Liang, M. Liang, Z. Song, Y. Xu, H. Jiang, H. Yu, Origami-enabled deformable silicon solar cells. Appl. Phys. Lett. **104**(8), 083501 (2014)
88. J.W. Hu, Z.P. Wu, S.W. Zhong, W.B. Zhang, S. Suresh, A. Mehta, N. Koratkar, Folding insensitive, high energy density lithium-ion battery featuring carbon nanotube current collectors. Carbon **87**, 292–298 (2015)
89. S. Fischer, K. Drechsler, S. Kilchert, A. Johnson, Mechanical tests for foldcore base material properties. Compos. Part A Appl. Sci. Manuf. **40**(12), 1941–1952 (2009)
90. S. Heimbs, J. Cichosz, M. Klaus, S. Kilchert, A.F. Johnson, Sandwich structures with textile-reinforced composite foldcores under impact loads. Compos. Struct. **92**(6), 1485 –1497 (2010)
91. S. Heimbs, S. Kilchert, S. Fischer, M. Klaus, E. Baranger, Sandwich structures with folded core: mechanical modeling and impact simulations, in *SAMPE Europe International Conference*, Paris, 2009, pp. 324–331
92. M. Klaus, H.-G. Reimerdes , Numerical investigation of different strength after impact test procedures, in *Proceedings of the IMPLAST 2010 conference*, Providence, RI, 2010
93. A.F. Johnson, Novel hybrid structural core sandwich materials for aircraft applications, in *Proceedings of the 11th Euro-Japanese Symposium on Composite Materials*, Porto, Portugal, 9–11 September 2008
94. M. Grzeschik, Performance of foldcores mechanical properties and testing, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-13324, Portland, OR, 2013 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A042
95. J.M. Gattas, Z. You, Quasi-static impact response of alternative origami-core sandwich panels, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-12681, Portland, OR, 2013 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A032

96. S. Fischer, Realistic FE simulation of foldcore sandwich structures.Int. J. Mech. Mater. Eng. **10**(1), 1–11 (2015)
97. R. Sturm, S. Fischer, Virtual design method for controlled failure in foldcore sandwich panels. Appl. Compos. Mater. **22**(6), 791–803 (2015)
98. R.K. Fathers, J.M. Gattas, Z. You, Quasi-static crushing of eggbox, cube, and modified cube foldcore sandwich structures. Int. J. Mech. Sci. **101–102**, 421–428 (2015)
99. J.M. Gattas, Z. You, The behaviour of curved-crease foldcores under low-velocity impact loads. Int. J. Solids Struct. **53**, 80–91 (2015)
100. J. Ma, Z. You, The origami crash box in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education*, 2011, pp. 277–290
101. J. Ma, Z. You, A novel origami crash box with varying profiles, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-13495 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A048
102. S.S. Tolman, I.L. Delimont, L.L. Howell, D.T. Fullwood, Material selection for elastic energy absorption in origami-inspired compliant corrugations. Smart Mater. Struct. **23**(9), 094010 (2014)
103. J. Ma, Z. You, Energy absorption of thin-walled square tubes with a prefolded origami pattern–Part I: geometry and numerical simulation. J. Appl. Mech. **81**(1), 011003 (2014)
104. J. Ma, J. Song, Y. Chen, An origami-inspired structure with graded stiffness. Int. J. Mech. Sci. **136**, 134–142 (2018)
105. K. Yang, S. Xu, S. Zhou, Y.M. Xie, Multi-objective optimization of multi-cell tubes with origami patterns for energy absorption. Thin Walled Struct. **123**, 100–113 (2018)
106. M. Schenk, S.D. Guest, Geometry of Miura-folded metamaterials. Proc. Natl. Acad. Sci. **110**(9), 3276–3281 (2013)
107. K. Fuchi, A.R. Diaz, E.J. Rothwell, R.O. Ouedraogo, J. Tang, An origami tunable metamaterial. J. Appl. Phys. **111**(8), 084905 (2012)
108. J.L. Silverberg, A.A. Evans, L. McLeod, R.C. Hayward, T. Hull, C.D. Santangelo, I. Cohen, Using origami design principles to fold reprogrammable mechanical metamaterials. Science **345**(6197), 647–650 (2014)
109. E.T. Filipov, T. Tachi, G.H. Paulino, Origami tubes assembled into stiff, yet reconfigurable structures and metamaterials. Proc. Natl. Acad. Sci. **112**(40), 12321–12326 (2015)
110. V. Brunck, F. Lechenault, A. Reid, M. Adda-Bedia, Elastic theory of origami-based metamaterials. Phys. Rev. E **93**(3), 033005 (2016)
111. X. Zhou, S. Zang, Z. You, Origami mechanical metamaterials based on the Miura-derivative fold patterns. Proc. R. Soc. A Math. Phys. Eng. Sci. **472**(2191), 20160361 (2016)
112. Y. Shen, Y. Pang, J. Wang, H. Ma, Z. Pei, S. Qu, Origami-inspired metamaterial absorbers for improving the larger-incident angle absorption. J. Phys. D Appl. Phys. **48**(44), 445008 (2015)
113. W. Jiang, H. Ma, M. Feng, L. Yan, J. Wang, J. Wang, S. Qu, Origami-inspired building block and parametric design for mechanical metamaterials. J. Phys. D Appl. Phys. **49**(31), 315302 (2016)
114. M.A.E. Kshad, H.E. Naguib, Characterization of origami shape memory metamaterials (SMMM) made of bio-polymer blends, in *Proceedings of SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring* (International Society for Optics and Photonics, San Diego, 2016), p. 98000H
115. E. Boatti, N. Vasios, K. Bertoldi, Origami metamaterials for tunable thermal expansion. Adv. Mater. **29**(26), 1700360 (2017)
116. Z. Wang, L. Jing, K. Yao, Y. Yang, B. Zheng, C.M. Soukoulis, H. Chen, Y. Liu, Origami-based reconfigurable metamaterials for tunable chirality. Adv. Mater. **29**(27), 1700412 (2017)
117. Y. Zhao, M.S. Nandra, Y.C. Tai, A MEMS intraocular origami coil, in *Proceedings of the 16th International Solid-State Sensors, Actuators and Microsystems Conference (TRANSDUCERS)* (IEEE, Piscataway, 2011), pp. 2172–2175
118. C. Yoon, R. Xiao, J. Park, J. Cha, T.D. Nguyen, D.H. Gracias, Functional stimuli responsive hydrogel devices by self-folding. Smart Mater. Struct. **23**(9), 094008 (2014)

119. A. Vorob'ev, P. Vaccaro, K. Kubota, S. Saravanan, T. Aida, Array of micromachined components fabricated using "micro-origami" method. Jpn. J. Appl. Phys. **42**(6S), 4024 (2003)
120. J.M.Z. Ocampo, P.O. Vaccaro, T. Fleischmann, T.-S. Wang, K. Kubota, T. Aida, T. Ohnishi, A. Sugimura, R. Izumoto, M. Hosoda, S. Nashima, Optical actuation of micromirrors fabricated by the micro-origami technique. Appl. Phys. Lett. **83**(18), 3647–3649 (2003)
121. T.G. Leong, A.M. Zarafshar, D.H. Gracias, Three-dimensional fabrication at small size scales. Small **6**(7), 792–806 (2010)
122. J. Rogers, Y. Huang, O.G. Schmidt, D.H. Gracias, Origami MEMS and NEMS. MRS Bull. **41**(2), 123–129 (2016)
123. A. Efimovskaya, D. Senkal, A.M. Shkel, Miniature origami-like folded MEMS TIMU, in *Proceedings of Transducers-2015 18th International Conference on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS)* (IEEE, Piscataway, 2015), pp. 584–587
124. Y. Morikawa, S. Yamagiwa, H. Sawahata, M. Ishida, T. Kawano, An origami-inspired ultrastretchable bioprobe film device, in *Proceedings of the 2016 IEEE 29th International Conference on Micro Electro Mechanical Systems (MEMS)* (IEEE, Piscataway, 2016), pp. 149–152
125. G.J. Hayes, Y. Liu, J. Genzer, G. Lazzi, M.D. Dickey, Self-folding origami microstrip antennas. IEEE Trans. Antennas Propag. **62**(10), 5416–5419 (2014)
126. A. Lebée, From folds to structures, a review. Int. J. Space Struct. **30**(2), 55–74 (2015)
127. N. Turner, B. Goodwine, M. Sen, A review of origami applications in mechanical engineering. Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci. (2015). https://doi.org/10.1177/0954406215597713
128. T. Hull (ed.), *Origami 3: Third International Meeting of Origami Science, Mathematics, and Education* (AK Peters, Natick, 2002)
129. R. Lang (ed.), *Origami 4: Fourth International Meeting of Origami Science, Mathematics, and Education* (AK Peters, Natick, 2009)
130. P. Wang-Iverson, R. Lang, M. Yim (eds.), *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (CRC Press, Boca Raton, 2011)
131. K. Miura, T. Kawasaki, T. Tachi, R. Uehara, R. Lang, P. Wang-Iverson (eds.), *Origami 6: Sixth International Meeting of Origami Science, Mathematics, and Education* (American Mathematical Society, Providence, 2011)
132. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr, D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
133. L. Ionov, Soft microorigami: self-folding polymer films. Soft Matter **7**(15), 6786–6791 (2011)
134. C. Lauff, T.W. Simpson, M. Frecker, Z. Ounaies, S. Ahmed, P. von Lockette, R. Strzelec, R. Sheridan, J.-M. Lien, Differentiating bending from folding in origami engineering using active materials, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2014-34702 (American Society of Mechanical Engineers, New York, 2014), p. V05BT08A040
135. S.M. Felton, K.P. Becker, D.M. Aukes, R.J. Wood, Self-folding with shape memory composites at the millimeter scale. J. Micromech. Microeng. **25**(8), 085004 (2015)
136. P.K. Kumar, D.C. Lagoudas, Introduction to shape memory alloys, in *Shape Memory Alloys: Modeling and Engineering Applications* (Springer, New York, 2008), pp. 1–51
137. K. Otsuka, C.M. Wayman, *Shape Memory Materials* (Cambridge University Press, Cambridge, 1999)
138. M.V. Gandhi, B.D. Thompson, *Smart Materials and Structures* (Springer, London, 1992)
139. R.C. Smith, *Smart Material Systems: Model Development* (SIAM, Philadelphia, 2005)
140. D.J. Leo, *Engineering Analysis of Smart Material Systems* (Wiley, Hoboken, 2007)
141. Y. Liu, K. Gall, M.L. Dunn, A.R. Greenberg, J. Diani, Thermomechanics of shape memory polymers: uniaxial experiments and constitutive modeling. Int. J. Plast. **22**(2), 279–313 (2006)
142. A. Lendlein, S. Kelch, Shape-memory polymers. Angew. Chem. Int. Ed. **41**(12), 2034–2057 (2002)

143. H. Tobushi, T. Hashimoto, S. Hayashi, E. Yamada, Thermomechanical constitutive modeling in shape memory polymer of polyurethane series. J. Intell. Mater. Syst. Struct. **8**(8), 711–718 (1997)
144. C. Liu, H. Qin, P.T. Mather, Review of progress in shape-memory polymers. J. Mater. Chem. **17**(16), 1543–1558 (2007)
145. L. Ionov, Hydrogel-based actuators: possibilities and limitations. Mater. Today **17**(10), 494–503 (2014)
146. K. Deligkaris, T.S. Tadele, W. Olthuis, A. van den Berg, Hydrogel-based devices for biomedical applications. Sens. Actuators B Chem. **147**(2), 765–774 (2010)
147. A. O'Halloran, F. O'malley, P. McHugh, A review on dielectric elastomer actuators, technology, applications, and challenges. J. Appl. Phys. **104**(7), 9 (2008)
148. M. Farshad, A. Benine, Magnetoactive elastomer composites. Polym. Test. **23**(3), 347–353 (2004)
149. L. Bowen, K. Springsteen, H. Feldstein, M. Frecker, T.W. Simpson, P. von Lockette, Development and validation of a dynamic model of magneto-active elastomer actuation of the origami waterbomb base. J. Mech. Robot. **7**(1), 011010 (2015)
150. E. Hawkes, B. An, N.M. Benbernou, H. Tanaka, S. Kim, E.D. Demaine, D. Rus, R.J. Wood, Programmable matter by folding. Proc. Natl. Acad. Sci. **107**(28), 12441–12445 (2010)
151. J. Paik, B. An, D. Rus, R.J. Wood, Robotic origamis: self-morphing modular robots, in *Proceedings of the 2nd International Conference on Morphological Computation ICMC*, EPFL-CONF-206919, Venice, 2012
152. Z. You, K. Kuribayashi, A novel origami stent, in *Proceedings of Summer Bioengineering Conference*, Key Biscayne, 2003, pp. 0257–0258
153. K. Kuribayashi, A novel foldable stent graft. Thesis, University of Oxford, 2004
154. C.D. Onal, R.J. Wood, D. Rus, An origami-inspired approach to worm robots. IEEE/ASME Trans. Mechatron. **18**(2), 430–438 (2013)
155. C.D. Onal, R.J. Wood, D. Rus, Towards printable robotics: origami-inspired planar fabrication of three-dimensional mechanisms, in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2011), pp. 4608–4613
156. D.-Y. Lee, J.-S. Kim, S.-R. Kim, J.-J. Park, K.-J. Cho, Design of deformable-wheeled robot based on origami structure with shape memory alloy coil spring, in *Proceedings of the 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* (IEEE, Piscataway, 2013), p. 120
157. M.T. Tolley, S.M. Felton, S. Miyashita, D. Aukes, D. Rus, R.J. Wood, Self-folding origami: shape memory composites activated by uniform heating. Smart Mater. Struct. **23**(9), 094006 (2014)
158. M.T. Tolley, S.M. Felton, S. Miyashita, L. Xu, B. Shin, M. Zhou, D. Rus, R.J. Wood, Self-folding shape memory laminates for automated fabrication, in *Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Piscataway, 2013), pp. 4931–4936
159. S.M. Felton, M.T. Tolley, C.D. Onal, D. Rus, R.J. Wood, Robot self-assembly by folding: a printed inchworm robot, in *Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2013), pp. 277–282
160. L.J. Wood, J. Rendon, R.J. Malak, D. Hartl, An origami-inspired, SMA actuated lifting structure, in *Proceedings of the ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Paper No. DETC2016-60261 (American Society of Mechanical Engineers, New York, 2016), p. V05BT07A024
161. A. Roudaut, A. Karnik, M. Löchtefeld, S. Subramanian, Morphees: toward high shape resolution in self-actuated flexible mobile devices, in *Proceedings of the 2013 SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, 2013), pp. 593–602
162. Q. Ge, C.K. Dunn, H. J. Qi, M.L. Dunn, Active origami by 4D printing. Smart Mater. Struct. **23**(9), 094007 (2014)
163. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr, Design and numerical analysis of an SMA mesh-based self-folding sheet. Smart Mater. Struct. **22**(9), 094008 (2013)

164. E.A. Peraza Hernandez, S. Hu, H.W. Kung, D. Hartl, E. Akleman, Towards building smart self-folding structures. Comput. Graph. **37**(6), 730–742 (2013)
165. D. Hartl, K. Lane, R. Malak, Computational design of a reconfigurable origami space structure incorporating shape memory alloy thin films, in *Proceedings of the ASME 2012 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS* (American Society of Mechanical Engineers, New York, 2012), pp. 277–285
166. T. Halbert, P. Moghadas, R. Malak, D. Hartl, Control of a shape memory alloy based self-folding sheet, in *Proceedings of the ASME 2014 International Design Engineering Technical Conference & Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2014-34703 (American Society of Mechanical Engineers, New York, 2014), p. V05BT08A041
167. A. Gomes, A. Nesbitt, R. Vertegaal, MorePhone: a study of actuated shape deformations for flexible thin-film smartphone notifications, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, 2013), pp. 583–592
168. J. Qi, L. Buechley, Animating paper using shape memory alloys, in *Proceedings of the 2012 SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, 2012), pp. 749–752
169. J. Qi, L. Buechley, Electronic popables: exploring paper-based computing through an interactive pop-up book, in *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction* (ACM, New York, 2010), pp. 121–128
170. N. Koizumi, K. Yasu, A. Liu, M. Sugimoto, M. Inami, Animated paper: a toolkit for building moving toys. Comput. Entertain. **8**(2), 7 (2010)
171. A.P. Lee, D.R. Ciarlo, P.A. Krulevitch, S. Lehew, J. Trevino, M.A. Northrup, A practical microgripper by fine alignment, eutectic bonding and SMA actuation. Sens. Actuators A Phys. **54**(1), 755–759 (1996)
172. P. Krulevitch, A.P. Lee, P.B. Ramsey, J.C. Trevino, J. Hamilton, M.A. Northrup, Thin film shape memory alloy microactuators. J. Microelectromech. Syst. **5**(4), 270–282 (1996)
173. Y. Liu, M. Miskiewicz, M.J. Escuti, J. Genzer, M.D. Dickey, Three-dimensional folding of pre-strained polymer sheets via absorption of laser light. J. Appl. Phys. **115**(20), 204911 (2014)
174. Y. Liu, R. Mailen, Y. Zhu, M.D. Dickey, J. Genzer, Simple geometric model to describe self-folding of polymer sheets. Phys. Rev. E **89**(4), 042601 (2014)
175. R. Saunders, D. Hartl, R. Malak, D. Lagoudas, Design and analysis of a self-folding SMA-SMP composite laminate, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2014-35151 (American Society of Mechanical Engineers, New York, 2014), p. V05BT08A048
176. K.E. Laflin, C.J. Morris, T. Muqeem, D.H. Gracias, Laser triggered sequential folding of microstructures. Appl. Phys. Lett. **101**(13), 131901 (2012)
177. K. Kalaitzidou, A.J. Crosby, Adaptive polymer particles. Appl. Phys. Lett. **93**(4), 041910 (2008)
178. B. Simpson, G. Nunnery, R. Tannenbaum, K. Kalaitzidou, Capture/release ability of thermo-responsive polymer particles. J. Mater. Chem. **20**(17), 3496–3501 (2010)
179. C. Sung, R. Lin, S. Miyashita, S. Yim, S. Kim, D. Rus, Self-folded soft robotic structures with controllable joints, in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017)
180. K. Fuchi, T.H. Ware, P.R. Buskohl, G.W. Reich, R.A. Vaia, T.J. White, J.J. Joo, Topology optimization for the design of folding liquid crystal elastomer actuators. Soft Matter **11**(37), 7288–7295 (2015)
181. K. Fuchi, P.R. Buskohl, T. Ware, R.A. Vaia, T.J. White, G.W. Reich, J.J. Joo, Inverse design of LCN films for origami applications using topology optimization, in *Proceedings of the ASME 2014 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, Paper No. SMASIS2014-7497 (American Society of Mechanical Engineers, New York, 2014), p. V001T01A011

182. L.T. de Haan, V. Gimenez-Pinto, A. Konya, T.-S. Nguyen, J. Verjans, C. Sánchez-Somolinos, J.V. Selinger, R.L.B. Selinger, D.J. Broer, A.P.H.J. Schenning, Accordion-like actuators of multiple 3D patterned liquid crystal polymer films. Adv. Funct. Mater. **24**(9), 1251–1258 (2014)
183. K.-U. Jeong, J.-H. Jang, D.-Y. Kim, C. Nah, J.H. Lee, M.-H. Lee, H.-J. Sun, C.-L. Wang, S.Z.D. Cheng, E.L. Thomas, Three-dimensional actuators transformed from the programmed two-dimensional structures via bending, twisting and folding mechanisms. J. Mater. Chem. **21**(19), 6824–6830 (2011)
184. S. Zakharchenko, N. Puretskiy, G. Stoychev, M. Stamm, L. Ionov, Temperature controlled encapsulation and release using partially biodegradable thermo-magneto-sensitive self-rolling tubes. Soft Matter **6**(12), 2633–2636 (2010)
185. G. Stoychev, N. Puretskiy, L. Ionov, Self-folding all-polymer thermoresponsive microcapsules. Soft Matter **7**(7), 3277–3279 (2011)
186. G. Stoychev, S. Zakharchenko, S. Turcaud, J.W.C. Dunlop, L. Ionov, Shape-programmed folding of stimuli-responsive polymer bilayers. ACS Nano **6**(5), 3925–3934 (2012)
187. K. Kumar, B. Nandan, V. Luchnikov, F. Simon, A. Vyalikh, U. Scheler, M. Stamm, A novel approach for the fabrication of silica and silica/metal hybrid microtubes. Chem. Mater. **21**(18), 4282–4287 (2009)
188. K. Kumar, B. Nandan, V. Luchnikov, E.B. Gowd, M. Stamm, Fabrication of metallic microtubes using self-rolled polymer tubes as templates. Langmuir **25**(13), 7667–7674 (2009)
189. K. Kumar, V. Luchnikov, V. Nandan, V. Senkovskyy, M. Stamm, Formation of self-rolled polymer microtubes studied by combinatorial approach. Eur. Polym. J. **44**(12), 4115–4121 (2008)
190. V. Luchnikov, K. Kumar, M. Stamm, Toroidal hollow-core microcavities produced by self-rolling of strained polymer bilayer films. J. Micromech. Microeng. **18**(3), 035041 (2008)
191. W. Guo, M. Li, J. Zhou, Modeling programmable deformation of self-folding all-polymer structures with temperature-sensitive hydrogels. Smart Mater. Struct. **22**(11), 115028 (2013)
192. H. He, J. Guan, J.L. Lee, An oral delivery device based on self-folding hydrogels. J. Control. Release **110**(2), 339–346 (2006)
193. H. He, X. Cao, L.J. Lee, Design of a novel hydrogel-based intelligent system for controlled drug release. J. Control. Release **95**(3), 391–402 (2004)
194. T.S. Shim, S.-H. Kim, C.-J. Heo, H.C. Jeon, S.-M. Yang, Controlled origami folding of hydrogel bilayers with sustained reversibility for robust microcarriers. Angew. Chem. **124**(6), 1449–1452 (2012)
195. J. Guan, H. He, D.J. Hansford, L.J. Lee, Self-folding of three-dimensional hydrogel microstructures. J. Phys. Chem. B **109**(49), 23134–23137 (2005)
196. S. Zakharchenko, E. Sperling, L. Ionov, Fully biodegradable self-rolled polymer tubes: a candidate for tissue engineering scaffolds. Biomacromolecules **12**(6), 2211–2215 (2011)
197. N. Bassik, A. Brafman, A.M. Zarafshar, M. Jamal, D. Luvsanjav, F.M. Selaru, D.H. Gracias, Enzymatically triggered actuation of miniaturized tools. J. Am. Chem. Soc. **132**(46), 16314–16317 (2010)
198. J.S. Randhawa, T.G. Leong, N. Bassik, B.R. Benson, M.T. Jochmans, D.H. Gracias, Pick-and-place using chemically actuated microgrippers. J. Am. Chem. Soc. **130**(51), 17238–17239 (2008)
199. W. Sun, F. Liu, Z. Ma, C. Li, J. Zhou, Soft mobile robots driven by foldable dielectric elastomer actuators. J. Appl. Phys. **120**(8), 084901 (2016)
200. H. Okuzaki, T. Saido, H. Suzuki, Y. Hara, H. Yan, A biomorphic origami actuator fabricated by folding a conducting paper. J. Phys. Conf. Ser. **127**(1), 012001 (2008)
201. H. Okuzaki, T. Kuwabara, K. Funasaka, T. Saido, Humidity-sensitive polypyrrole films for electro-active polymer actuators. Adv. Funct. Mater. **23**(36), 4400–4407 (2013)
202. P. von Lockette, R. Sheridan, Folding actuation and locomotion of novel magneto-active elastomer (MAE) composites, in *Proceedings of the ASME 2013 Conference on Smart Materials, Adaptive Structures, and Intelligent Systems SMASIS*, Paper No. SMASIS2013-3222, 16–18 September 2013, p. V001T01A020

203. P. von Lockette, Fabrication and performance of magneto-active elastomer composite structures, in *Proceedings of the ASME 2014 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, Paper No. SMASIS2014-7590 (American Society of Mechanical Engineers, New York, 2014), p. V001T01A019
204. S. Ahmed, C. Lauff, A. Crivaro, K. McGough, R. Sheridan, M. Frecker, P. von Lockette, Z. Ounaies, T. Simpson, J.-M. Lien, R. Strzelec, Multi-field responsive origami structures: preliminary modeling and experiments, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-12405, Portland, OR, 4–7 August 2013, p. V06BT07A028
205. A. Crivaro, R. Sheridan, M. Frecker, T.W. Simpson, P. Von Lockette, Bistable compliant mechanism using magneto active elastomer actuation. J. Intell. Mater. Syst. Struct. (2015). https://doi.org/1045389X15620037
206. L. Bowen, K. Springsteen, M. Frecker, T. Simpson, Optimization of a dynamic model of magnetic actuation of an origami mechanism, in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2015-47458 (American Society of Mechanical Engineers, New York, 2015), p. V05BT08A05
207. L. Bowen, K. Springsteen, M. Frecker, T. Simpson, Trade space exploration of magnetically actuated origami mechanisms. J. Mech. Robot. **8**(3), 031012 (2016)
208. K. McGough, S. Ahmed, M. Frecker, Z. Ounaies, Finite element analysis and validation of dielectric elastomer actuators used for active origami. Smart Mater. Struct. **23**(9), 094002 (2014)
209. S. Ahmed, Z. Ounaies, M. Frecker, Investigating the performance and properties of dielectric elastomer actuators as a potential means to actuate origami structures. Smart Mater. Struct. **23**(9), 094003 (2014)
210. R.N. Saunders, D.J. Hartl, J.G. Boyd, D.C. Lagoudas, Modeling and development of a twisting wing using inductively heated shape memory alloy actuators, in *Proceedings of SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring* (International Society for Optics and Photonics, San Diego, 2015), p. 94310U–94310U–8
211. R.N. Saunders, J.G. Boyd, D.J. Hartl, J.K. Brown, F.T. Calkins, D.C. Lagoudas, A validated model for induction heating of shape memory alloy actuators. Smart Mater. Struct. **25**(4), 045022 (2016)
212. T.J. Cognata, D. Hartl, R. Sheth, C. Dinsmore, A morphing radiator for high-turndown thermal control of crewed space exploration vehicles, in *Proceedings of 23rd AIAA/AHS Adaptive Structures Conference*, AIAA 2015-1509, 2015
213. C.L. Bertagne, R.B. Sheth, D.J. Hartl, J.D. Whitcomb, Simulating coupled thermal-mechanical interactions in morphing radiators, in *Proceedings of SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring* (International Society for Optics and Photonics, San Diego, 2015), p. 94312F–94312F–10
214. B. An, N. Benbernou, E.D. Demaine, D. Rus, Planning to fold multiple objects from a single self-folding sheet. Robotica **29**(1), 87–102 (2011)
215. B. An, D. Rus, Designing and programming self-folding sheets. Robot. Auton. Syst. **62**(7), 976–1001 (2014)
216. E.A. Peraza Hernandez, D.J. Hartl, K.R. Frei, E. Akleman, Connectivity of shape memory alloy-based self-folding structures, in *Proceedings of the 22nd AIAA/ASME/AHS Adaptive Structures Conference, in AIAA SciTech* (National Harbor, Maryland, 2014), p. 1415
217. E. Peraza Hernandez, D. Hartl, R. Malak, E. Akleman, O. Gonen, H. Kung, Design tools for patterned self-folding reconfigurable structures based on programmable active laminates. J. Mech. Robot. **8**(3), 031015 (2016)

218. P. Moghadas, R. Malak, D. Hartl, Reinforcement learning for control of a shape memory alloy based self-folding sheet, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/-CIE*, Paper No. DETC2015-46980 (American Society of Mechanical Engineers, New York, 2015), p. V05BT08A044
219. T.R. Halbert, An improved algorithm for sequential information-gathering decisions in design under uncertainty. Master's thesis, Texas A&M University, College Station, TX, USA, 2015
220. D. Hartl, K. Lane, R. Malak, Design of a massively reconfigurable origami space structure incorporating shape memory alloys, in *Proceedings of the ASME 2012 International Mechanical Engineering Congress and Exposition IMECE*, Paper No. IMECE2012-86391 (American Society of Mechanical Engineers, New York, 2012), p. 115–122
221. T. Halbert, E. Peraza Hernandez, R. Malak, D. Hartl, Numerically validated reduced-order model for laminates containing shape memory alloy wire meshes. J. Intell. Mater. Syst. Struct. (2015). https://doi.org/1045389X15595295
222. E.A. Peraza Hernandez, B. Kiefer, D.J. Hartl, A. Menzel, D.C. Lagoudas, Analytical investigation of structurally stable configurations in shape memory alloy-actuated plates. Int. J. Solids Struct. **69**, 442–458 (2015)
223. E. Peraza Hernandez, D. Hartl, D. Lagoudas, Modeling of shape memory alloy wire meshes using effective lamina properties for improved analysis efficiency, in *Proceedings of the ASME 2013 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS 2013*, Paper No. SMASIS2013-3094 (American Society of Mechanical Engineers, New York, 2013), p. V001T01A009
224. V. Brailovski, S. Prokoshkin, P. Terriault, F. Trochu, *Shape Memory Alloys: Fundamentals, Modeling and Applications* (Université du Québec. École de technologie supérieure, Québec, 2003)
225. L.G. Machado, M.A. Savi, Medical applications of shape memory alloys. Braz. J. Med. Biol. Res. **36**(6), 683–691 (2003)
226. N.B. Morgan, Medical shape memory alloy applications—the market and its products. Mater. Sci. Eng. A **378**(1), 16–23 (2004)
227. J.M. Jani, M. Leary, A. Subic, M.A. Gibson, A review of shape memory alloy research, applications and opportunities. Mater. Des. **56**, 1078–1113 (2014)
228. G. Esquivel, D. Weiser, D. Hartl, D. Whitten, POP-OP: a shape memory-based morphing wall. Int. J. Archit. Comput. **11**(3), 347–362 (2013)
229. J. Berry, J.H. Seo, Incorporation of shape memory polymers in interactive design, in *Proceedings of the 21st International Symposium of Electronic Art ISEA*, 2015
230. C. Schwesig, I. Poupyrev, E. Mori, Gummi: a bendable computer, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, 2004), pp. 263–270
231. J.-S. Park, T.-W. Kim, D. Stryakhilev, J.-S. Lee, S.-G. An, Y.-S. Pyo, D.-B. Lee, Y.G. Mo, D.-U. Jin, H.K. Chung, Flexible full color organic light-emitting diode display on polyimide plastic substrate driven by amorphous indium gallium zinc oxide thin-film transistors. Appl. Phys. Lett. **95**(1), 013503–013503 (2009)
232. B. Lahey, A. Girouard, W. Burleson, R. Vertegaal, PaperPhone: understanding the use of bend gestures in mobile devices with flexible electronic paper displays, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, 2011), pp. 1303–1312
233. A. Minuto, A. Nijholt, Smart material interfaces as a methodology for interaction: a survey of SMIs' state of the art and development, in *Proceedings of the Second International Workshop on Smart Material Interfaces: Another Step to a Material Future* (ACM, New York, 2013), pp. 1–6
234. A. Gomes, R. Vertegaal, PaperFold: evaluating shape changes for viewport transformations in foldable thin-film display devices, in *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction* (ACM, New York, 2015), pp. 153–160

235. D. Tan, M. Kumorek, A.A. Garcia, A. Mooney, D. Bekoe, Projectagami: a foldable mobile device with shape interactive applications, in *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (ACM, New York, 2015), pp. 1555–1560
236. L.-W. Kang, M.-F. Weng, C.-L. Jheng, C.-Y. Tseng, S.K. Ramesh, A. Gureja, H.-C. Hsu, C.-H. Yeh, Content-aware image retargeting for image display on foldable mobile devices. Procedia Comput. Sci. **56**, 104–110 (2015)
237. H. Meng, G. Li, A review of stimuli-responsive shape memory polymer composites. Polymer **54**(9), 2199–2221 (2013)
238. Q. Li, *Intelligent Stimuli-Responsive Materials: From Well-Defined Nanostructures to Applications* (Wiley, Hoboken, 2013)
239. A. Mata, A.J. Fleischman, S. Roy, Characterization of polydimethylsiloxane (PDMS) properties for biomedical micro/nanosystems. Biomed. Microdevices **7**(4), 281–293 (2005)
240. F. Schneider, J. Draheim, R. Kamberger, U. Wallrabe, Process and material properties of polydimethylsiloxane (PDMS) for optical MEMS. Sens. Actuators A Phys. **151**(2), 95–99 (2009)
241. V. Luchnikov, L. Ionov, M. Stamm, Self-rolled polymer tubes: novel tools for microfluidics, microbiology, and drug-delivery systems. Macromol. Rapid Commun. **32**(24), 1943–1952 (2011)
242. L. Ionov, Nature-inspired stimuli-responsive self-folding materials, in *Intelligent Stimuli-responsive Materials: From Well-defined Nanostructures to Applications* (Wiley, Hoboken, 2013), pp. 1–16
243. A.V. Prinz, V.Y. Prinz, Application of semiconductor micro-and nanotubes in biology. Surf. Sci. **532**, 911–915 (2003)
244. S. Ahmed, Z. Ounaies, E.A.F. Arrojado, Electric field-induced bending and folding of polymer sheets. Sens. Actuators A Phys. **260**, 68–80 (2017)
245. A. Diaz, J.I. Castillo, J.A. Logan, W.-Y. Lee, Electrochemistry of conducting polypyrrole films. J. Electroanal. Chem. Interfacial Electrochem. **129**(1), 115–132 (1981)
246. A.F. Diaz, B. Hall, Mechanical properties of electrochemically prepared polypyrrole films. IBM J. Res. Dev. **27**(4), 342–347 (1983)
247. M. Farshad, A. Benine, Magnetoactive elastomer composites. Polym. Test. **23**(3), 347–353 (2004)
248. B.M. Cowan, Magnetically induced actuation and optimization of the Miura-ori structure. Master's thesis, The Pennsylvania State University, 2015
249. R.J. Lang, A computational algorithm for origami design, in *Proceedings of the Twelfth Annual Symposium on Computational Geometry* (ACM, New York, 1996), pp. 98–105
250. T.A. Evans, R.J. Lang, S.P. Magleby, L.L. Howell, Rigidly foldable origami gadgets and tessellations. R. Soc. Open Sci. **2**(9), 150067 (2015)
251. R.J. Lang, The tree method of origami design, in *The Second International Meeting of Origami Science and Scientific Origami*, ed. by K. Miura (Seian University of Art of Design, Otsu, 1994), pp. 72–82
252. R. Lang, Trees and circles: an efficient algorithm for origami design, in *Proceedings of the 3rd International Meeting of Origami Science, Math, and Education*, 2001
253. R.J. Lang, *Origami Design Secrets: Mathematical Methods for an Ancient Art* (CRC Press, Boca Raton, 2011)
254. R. Lang, Treemaker, available at http://www.langorigami.com/article/treemaker, 1998
255. E.D. Demaine, M.L. Demaine, Recent results in computational origami, in *Origami 3: Third International Meeting of Origami Mathematics, Science, and Education*, 2002, pp. 3–16
256. E.D. Demaine, M.L. Demaine, J.S.B. Mitchell, Folding flat silhouettes and wrapping polyhedral packages: new results in computational origami. Comput. Geom. **16**(1), 3–21 (2000)
257. K. Fuchi, A.R. Diaz, Origami design by topology optimization. J. Mech. Des. **135**(11), 111003 (2013)
258. M.P. Bendsoe, *Topology Optimization: Theory, Methods and Applications* (Springer, Berlin, 2003)

259. W. Liu, K. Tai, Optimal design of flat patterns for 3D folded structures by unfolding with topological validation. Comput. Aided Des. **39**(10), 898–913 (2007)
260. W. Schlickenrieder, Nets of polyhedra. Master's thesis, Technische Universität Berlin, 1997
261. M. Bern, E.D. Demaine, D. Eppstein, E. Kuo, A. Mantler, J. Snoeyink, Ununfoldable polyhedra with convex faces. Comput. Geom. **24**(2), 51–62 (2003)
262. T. Tachi, 3D origami design based on tucking molecule, in *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education*, 2009, pp. 259–272
263. T. Tachi, Origamizer, available at http://www.tsg.ne.jp/TT/software/, 2008
264. B.J. Edmondson, R.J. Lang, M.R. Morgan, S.P. Magleby, L.L. Howell, Thick rigidly foldable structures realized by an offset panel technique, in *Origami 6: I. Mathematics*, 2015, p. 149
265. T. Tachi, Rigid-foldable thick origami, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education*, 2011, pp. 253–264
266. Y. Chen, R. Peng, Z. You, Origami of thick panels. Science **349**(6246), 396–400 (2015)
267. S.A. Zirbel, M.E. Wilson, S.P. Magleby, L.L. Howell, An origami-inspired self-deployable array, in *Proceedings of the ASME 2013 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, Paper No. SMASIS2013-3296, Snowbird, UT, 2013 (American Society of Mechanical Engineers, New York, 2013), p. V001T01A026
268. S.A. Zirbel, B.P. Trease, M.W. Thomson, R.J. Lang, S.P. Magleby, L.H. Howell, HanaFlex: a large solar array for space applications, in *Proceedings of SPIE*, vol. 9467, 2015, p. 94671C–94671C–9
269. J.S. Ku, E.D. Demaine, Folding flat crease patterns with thick materials, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2015-48039 (American Society of Mechanical Engineers, New York, 2015), p. V05BT08A056
270. J.S. Ku, E.D. Demaine, Folding flat crease patterns with thick materials. J. Mech. Robot. **8**(3), 031003 (2016)
271. K. Fuchi, P.R. Buskohl, G. Bazzan, M.F. Durstock, G.W. Reich, R.A. Vaia, J.J. Joo, Origami actuator design and networking through crease topology optimization. J. Mech. Des. **137**(9), 091401 (2015)
272. K. Saito, A. Tsukahara, Y. Okabe, Designing of self-deploying origami structures using geometrically misaligned crease patterns. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. **472**(2185), 20150235 (2016)
273. D.A. McAdams, W. Li, A novel method to design and optimize flat-foldable origami structures through a genetic algorithm. J. Comput. Inf. Sci. Eng. **14**(3), 031008 (2014)
274. S. Ishida, T. Nojima, I. Hagiwara, Application of conformal maps to origami-based structures: New method to design deployable circular membranes, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-12725, Portland, OR, 2013 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A035
275. K. Zhu, C. Deshan, O.N.N. Fernando, Snap-n-Fold: origami pattern generation based real-life object structure, in *CHI'12 Extended Abstracts on Human Factors in Computing Systems* (ACM, New York, 2012), pp. 2345–2350
276. T. Tachi, Designing rigidly foldable horns using Bricard's octahedron. J. Mech. Robot. **8**(3), 031008 (2016)
277. R.J. Lang, S. Magleby, L. Howell, Single degree-of-freedom rigidly foldable cut origami flashers. J. Mech. Robot. **8**(3), 031005 (2016)
278. T. Tachi, Simulation of rigid origami. *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education*, 2009, pp. 175–187
279. T. Ida, H. Takahashi, M. Marin, A. Kasem, F. Ghourabi, Computational origami system Eos, in *Origami 4: Proceedings of 4th International Conference on Origami, Science, Mathematics and Education*, 2009, pp. 285–293
280. H. Huzita, Axiomatic development of origami geometry, in *Proceedings of the First International Meeting of Origami Science and Technology*, 1989, pp. 143–158

281. R.C. Alperin, R.J. Lang, One-, two-, and multi-fold origami axioms, in *Origami 4: Proceedings of 4th International Conference on Origami, Science, Mathematics and Education*, 2009, pp. 371–393
282. T. Hull, *Project origami: Activities for Exploring Mathematics* (CRC Press, Boca Raton, 2012)
283. A. Kasem, F. Ghourabi, T. Ida, Origami axioms and circle extension, in *Proceedings of the 2011 ACM Symposium on Applied Computing* (ACM, New York, 2011), pp. 1106–1111
284. F. Ghourabi, T. Ida, H. Takahashi, M. Marin, A. Kasem, Logical and algebraic view of Huzita's origami axioms with applications to computational origami, in *Proceedings of the 2007 ACM Symposium on Applied Computing* (ACM, New York, 2007), pp. 767–772
285. W. Wu, Z. You, Modelling rigid origami with quaternions and dual quaternions. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. **466**(2119), 2155–2174 (2010)
286. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Kinematics of origami structures with smooth folds. J. Mech. Robot. **8**(6), 061019 (2016)
287. E.D. Demaine, M.L. Demaine, Recent results in computational origami, in *Proceedings of the 3rd International Meeting of Origami Science, Math, and Education* Citeseer, 2001, pp. 3–16
288. M.S. Moses, M.K. Ackerman, G.S. Chirikjian, Origami rotors: imparting continuous rotation to a moving platform using compliant flexure hinges, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, paper No. DETC2013-12753, Portland, OR, 2013 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A037
289. S.-M. Belcastro, T.C. Hull, A mathematical model for non-flat origami, in *Origami 3: Third International Meeting of Origami Mathematics, Science, and Education*, 2002, pp. 39–51
290. S.-M. Belcastro, T.C. Hull, Modelling the folding of paper into three dimensions using affine transformations. Linear Algebra Appl. **348**(1–3), 273–282 (2002)
291. T. Tachi, Rigid origami simulator, available at http://www.tsg.ne.jp/TT/software/, 2007
292. T. Tachi, Freeform origami, available at http://www.tsg.ne.jp/TT/software/, 2013
293. T. Tachi, Freeform variations of origami. J. Geom. Graph. **14**(2), 203–215 (2010)
294. A. Kasem, T. Ida, H. Takahashi, M. Marin, F. Ghourabi, E-origami system Eos, in *Proceedings of the Annual Symposium of Japan Society for Software Science and Technology, JSSST*, Tokyo, Japan (September 2006)
295. T. Ida, H. Takahashi, M. Marin, F. Ghourabi, Modeling origami for computational construction and beyond, in *Computational Science and Its Applications–ICCSA 2007* (Springer, 2007), pp. 653–665
296. A. Kasem, T. Ida, Computational origami environment on the web. Front. Comput. Sci. China **2**(1), 39–54 (2008)
297. M. Schenk, S.D. Guest, Origami folding: a structural engineering approach, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education*, 2011, pp. 291–304
298. A.R. Diaz, Origami folding and bar frameworks, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2014-34149 (American Society of Mechanical Engineers, New York, 2014), p. V05BT08A031
299. A.A. Evans, J.L. Silverberg, C.D. Santangelo, Lattice mechanics of origami tessellations. Phys. Rev. E **92**(1), 013205 (2015)
300. K. Fuchi, P.R. Buskohl, G. Bazzan, M.F. Durstock, G.W. Reich, R.A. Vaia, J.J. Joo, Design optimization challenges of origami-based mechanisms with sequenced folding. J. Mech. Robot. **8**(5), 051011 (2016)
301. K. Saito, A. Tsukahara, Y. Okabe, New deployable structures based on an elastic origami model. J. Mech. Des. **137**(2), 021402 (2015)

302. Y. Nakada, Y. Fujieda, T. Mori, H. Iwai, K. Nakaya, R. Uehara, M. Yamabe, On a stiffness model for origami folding, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2015-46731 (American Society of Mechanical Engineers, New York, 2015), p. V05BT08A039
303. S. Heimbs, Virtual testing of sandwich core structures using dynamic finite element simulations. Comput. Mater. Sci. **45**(2), 205–216 (2009)
304. J. Ma, Z. You, Energy absorption of thin-walled beams with a pre-folded origami pattern. Thin Walled Struct. **73**, 198–206 (2013)
305. J. Song, Y. Chen, G. Lu, Axial crushing of thin-walled structures with origami patterns. Thin Walled Struct. **54**, 65–71 (2012)
306. Y. Li, Z. You, Thin-walled open-section origami beams for energy absorption, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2014-35204 (American Society of Mechanical Engineers, New York, 2014), p. V003T01A014
307. C. Qiu, V. Aminzadeh, J.S. Dai, Kinematic analysis and stiffness validation of origami cartons. J. Mech. Des. **135**(11), 111004 (2013)
308. C. Qiu, V. Aminzadeh, J.S. Dai, Kinematic and stiffness analysis of an origami-type carton, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2013-12343, Portland, OR, 2013 (American Society of Mechanical Engineers, New York, 2013), p. V06BT07A026
309. J.T. Oden, J.N. Reddy, *An Introduction to the Mathematical Theory of Finite Elements* (Dover Publications, Mineola, 2012)
310. J.N. Reddy, *An Introduction to the Finite Element Method*, vol. 2 (McGraw-Hill, New York, 1993)
311. J.N. Reddy, *Mechanics of Laminated Composite Plates: Theory and Analysis* (CRC Press, Boca Raton, 1997)
312. E. Carrera, Theories and finite elements for multilayered, anisotropic, composite plates and shells. Arch. Comput. Methods Eng. **9**(2), 87–140 (2002)
313. H.T.Y. Yang, S. Saigal, A. Masud, R.K. Kapania, A survey of recent shell finite elements. Int. J. Numer. Methods Eng. **47**(1–3), 101–127 (2000)
314. J. Ma, D. Hou, Y. Chen, Z. You, Quasi-static axial crushing of thin-walled tubes with a kite-shape rigid origami pattern: Numerical simulation. Thin Walled Struct. **100**, 38–47 (2016)
315. D. Hou, Y. Chen, J. Ma, Z. You, Axial crushing of thin-walled tubes with kite-shape pattern, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2015-46671 (American Society of Mechanical Engineers, New York, 2015), p. V05BT08A037
316. K. Yang, S. Xu, J. Shen, S. Zhou, Y.M. Xie, Energy absorption of thin-walled tubes with pre-folded origami patterns: numerical simulation and experimental verification. Thin Walled Struct. **103**, 33–44 (2016)
317. S. Heimbs, P. Middendorf, S. Kilchert, A.F. Johnson, M. Maier, Experimental and numerical analysis of composite folded sandwich core structures under compression. Appl. Compos. Mater. **14**(5–6), 363–377 (2007)
318. R.W. Mailen, M.D. Dickey, J. Genzer, M.A. Zikry, A fully coupled thermo-viscoelastic finite element model for self-folding shape memory polymer sheets. J. Polym. Sci. Part B Polym. Phys. **55**(16), 1207–1219 (2017)
319. E. Peraza Hernandez, D. Hartl, E. Galvan, R. Malak, Design and optimization of a shape memory alloy-based self-folding sheet. J. Mech. Des. **135**(11), 111007 (2013)
320. E. Peraza Hernandez, K. Frei, D. Hartl, D. Lagoudas, Folding patterns and shape optimization using SMA-based self-folding laminates. Proc. SPIE **9057**, 90571G–90571G–13 (2014)
321. G. Turk, M. Levoy, The Stanford Bunny. The Stanford 3D Scanning Repository, http://graphics.stanford.edu/data/3Dscanrep/, 1994

322. M. Behl, M.Y. Razzaq, A. Lendlein, Multifunctional shape-memory polymers. Adv. Mater. **22**(31), 3388–3410 (2010)
323. A. Firouzeh, Y. Sun, H. Lee, J. Paik, Sensor and actuator integrated low profile robotic origami, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 4937–4944
324. J.C. Athas, C.P. Nguyen, B.C. Zarket, A. Gargava, Z. Nie, S.R. Raghavan, Enzyme-triggered folding of hydrogels: Toward a mimic of the Venus Flytrap. ACS Appl. Mater. Interfaces **8**(29), 19066–19074 (2016)

Chapter 2
Kinematics of Origami Structures with Creased Folds

Abstract Having reviewed the existing and potential engineering applications of active origami structures in Chap. 1, it is evident that such structures have significant complexity in their geometry and motion. Therefore, mathematical models for origami are needed to enable the development of advanced active origami structures. The following assumptions generally apply to the development of mathematical models for origami: folds are straight creases (termed as *creased folds*), and planar faces bounded by the folds and the sheet boundary are rigid (i.e., these faces do not stretch or bend). In this chapter, we present a kinematic model for origami based on such assumptions. We also address the implementation of the model in a computational environment.

2.1 Introduction

Traditionally, origami is known as the art of folding sheets of paper into decorative and often intriguing shapes, either abstract in form or representative of realistic objects (e.g., plants, animals) [1]. As reviewed in Chap. 1, origami is currently being applied to the development of novel engineering structures comprised of complex materials. *Kinematic modeling* of origami structures, *which is the study of their motion without considering their constituent materials or the physical stimuli causing such a motion*, is of high importance as it permits the understanding of the complex deformation of origami structures and the development of computational tools for their simulation [2–4]. Kinematic modeling of origami structures is also the first step towards the full physical modeling of such structures. The reader is referred to Sect. 1.4 for a review on previous efforts in kinematic modeling and simulation of origami structures. In this chapter, we present a kinematic model for origami and we also address the implementation of such a model in a computational framework.

Electronic Supplementary Material The online version of this article (https://doi.org/10.1007/978-3-319-91866-2_2) contains supplementary material, which is available to authorized users.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_2

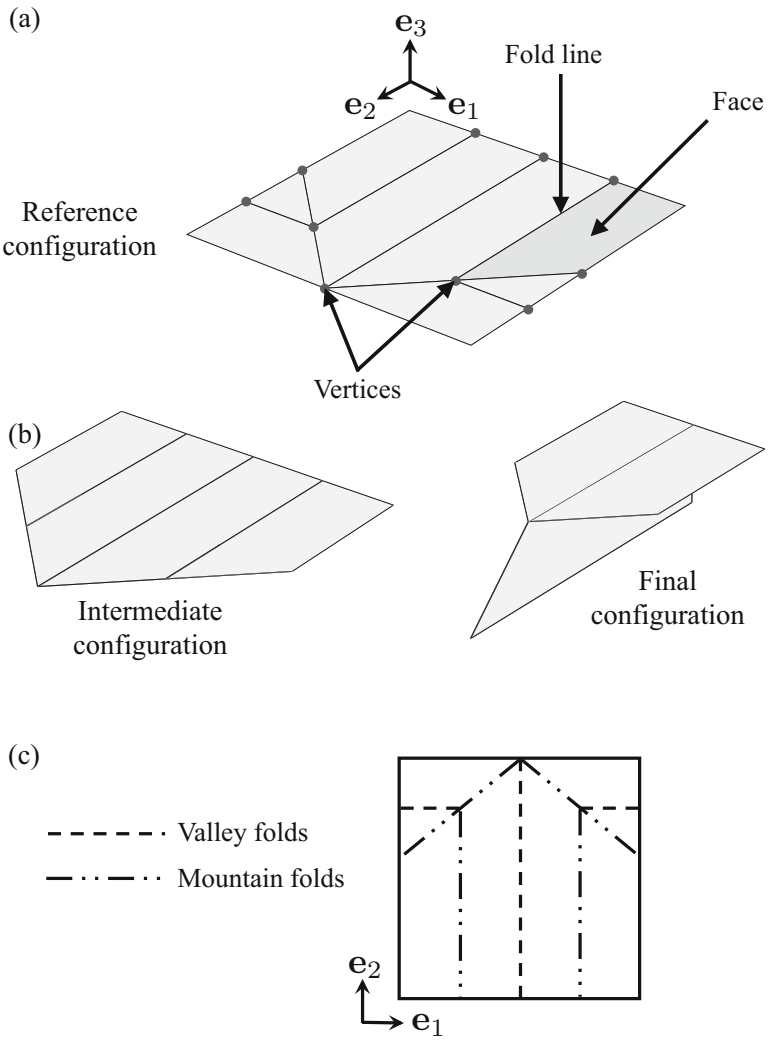


Fig. 2.1 Schematic of a simple origami airplane: (**a**) Planar *reference configuration* of the origami sheet showing its various elements; (**b**) An intermediate configuration and the *final configuration*; (**c**) Mountain-valley assignment

Throughout the development of the kinematic model presented in this chapter, we examine the various elements that allow us to mathematically model the folding motion of an origami structure such as the origami airplane shown in Fig. 2.1. In such a figure, the final shape of the airplane is obtained from an initially planar sheet exclusively through folding.

The origami airplane in Fig. 2.1 illustrates the assumptions made in the kinematic model for origami to be developed in this chapter. First, it is noted that the initial configuration of the sheet shown in Fig. 2.1a, which is termed here as the *reference configuration*, is *a bounded planar surface that does not have overlaps*. Second, it is observed in Fig. 2.1a, b that the sheet is folded along straight creases, denoted as *creased folds*. Third, the sheet regions bounded by the creased folds and the sheet boundary, which are denoted as the *faces*, remain *rigid* during folding as implicitly indicated in Fig. 2.1a, b (i.e., they do not stretch or bend). Models of origami making these assumptions have been widely employed for the analysis of deployable structures and mechanisms [5–26].

Figure 2.1 also illustrates concepts of origami that are extensively used throughout this chapter. The line segments coincident with the creased folds in the reference configuration are called *fold lines*, which are typically defined by their end points, formally called *vertices*. The layout of the fold lines is known as the *fold pattern* [27]. The fold pattern, along with the history of folding motion (Fig. 2.1b), determine the final configuration of the sheet. A *"mountain-valley" assignment* such as that shown in Fig. 2.1c is a useful schematic that indicates the folding direction of each creased fold in the sheet. For *mountain folds*, faces on either side of the fold line can be thought of as rotating into the page, while for *valley folds* they can be thought of as rotating out of it.

The origami model to be presented in this chapter allows us to mathematically describe the folding motion of origami sheets with creased folds having arbitrary fold patterns. In Sect. 2.2, we formalize the concepts and assumptions previously mentioned in this section and illustrated in Fig. 2.1. In Sect. 2.3, we provide the mathematical description of the fold pattern in origami sheets with creased folds. Subsequently, we address the fundamental constraints on the fold pattern and folding motion of origami sheets in Sect. 2.4. The formulation of the mapping that relates the reference configuration to any other configuration of an origami sheet is examined in Sect. 2.5. The implementation of the model in a computational framework is addressed in Sect. 2.6. Simulation results obtained using the implemented model are provided in Sect. 2.7. To help clarify some of the more complex aspects of the model, step-by-step examples are provided throughout the chapter. Additionally, we review the mathematical notation, concepts, and formulas used throughout the development of the kinematic model in Appendix A. Upon the study of this chapter, the reader will be able to derive the kinematic model for origami with creased folds presented here and to implement it in a computational framework. The Supplemental Material associated with this chapter provides the reader with a number of MATLAB® scripts that represent such an implementation.

2.2 Fundamental Concepts

Here, we address the fundamental concepts of origami with creased folds that are used in the model presented in this chapter. The data required to define the fold pattern such as the position of the vertices and their connectivity are assumed given.

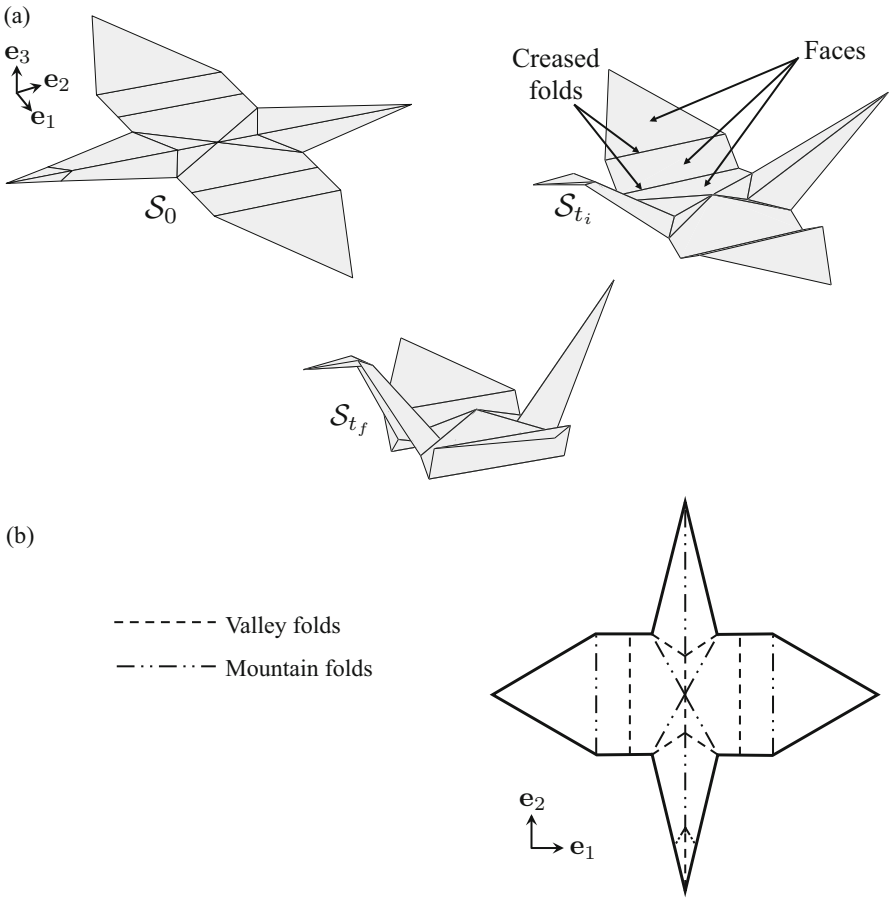


Fig. 2.2 Example of an origami crane: (**a**) *Reference configuration* $\mathcal{S}_0$, an intermediate configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and *final configuration* $\mathcal{S}_{t_f}$; (**b**) Mountain-valley assignment

We address two methods for the design of fold patterns to achieve goal shapes in Chaps. 3 and 4.

Let us consider the example of an origami crane illustrated in Fig. 2.2a. The *sheet* comprising the origami crane is mathematically described as a *surface in three-dimensional space*. Such a sheet is a *connected* surface, which means that any two points in the sheet can be joined by a path lying entirely in the sheet. The sheet also has an outer closed boundary and thus it is a *bounded* surface (Fig. 2.2b). The sheet is divided into various *faces* that are joined at straight edges corresponding to the *creased folds*.

Figures 2.2a and 2.3a show different *configurations* of origami sheets during their folding motion. We consider the folding motion of such origami sheets from $t = 0$ to $t = t_{t_f}$ where $t \in [0, t_f]$ is a *time parameter* that tracks the history

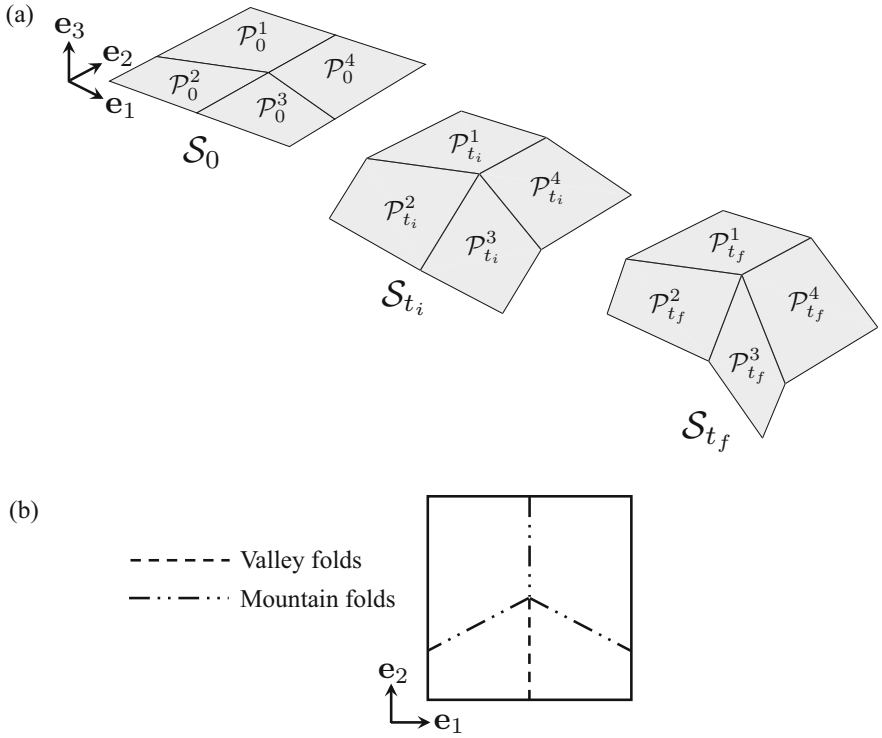


Fig. 2.3 Illustration of a sheet having a *Miura-Ori* fold pattern: (**a**) Reference configuration $\mathcal{S}_0$, an intermediate configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$. The configuration of each face at $t = 0, t_i, t_f$ is indicated; (**b**) Mountain-valley assignment

of folding motion. In a given *configuration*, each point in the sheet is located at some definite position in three-dimensional space $\mathbb{R}^3$. The current configuration of a sheet at any t is denoted $\mathcal{S}_t$. The configurations of the *faces* comprising $\mathcal{S}_t$ are denoted $\mathcal{P}_t^1, \ldots, \mathcal{P}_t^{N_\mathcal{P}} \subset \mathcal{S}_t$, where $N_\mathcal{P}$ *is the number of faces in the sheet* (i.e., $\mathcal{S}_t = \bigcup_{i=1}^{N_\mathcal{P}} \mathcal{P}_t^i$). This is illustrated in Fig. 2.3a.

To describe the position of the points in a configuration of the sheet, we introduce the vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \in \mathbb{R}^3$ that form the natural basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ of $\mathbb{R}^3$ (see Figs. 2.1, 2.2, and 2.3). These vectors are given as follows:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{2.1}$$

The initial configuration of the sheet at $t = 0$ is termed as the *reference configuration* $\mathcal{S}_0$ and is assumed *planar*. As shown in Figs. 2.2a and 2.3a, the reference configuration $\mathcal{S}_0$ *is fully contained in the plane spanned by* $\mathbf{e}_1$ *and*

$\mathbf{e}_2$ *and has no overlaps*. Although certain applications that utilize origami do not consider an initially planar configuration (e.g., [28–31]), the planar reference configuration $\mathcal{S}_0$ assumed here is in agreement with most conventional origami modeling approaches [32, 33] and is maintained for the sake of simplicity. The side of $\mathcal{S}_0$ with normal vector $\mathbf{e}_3$ is selected as the *positive side* of the sheet. The configuration of the sheet at $t = t_f$ is termed as the *final configuration* $\mathcal{S}_{t_f}$.

In the kinematic model presented in this chapter, we are interested in configurations having the following properties [32, 33]:

1. The faces have undergone only rigid deformations (i.e., they neither stretch nor bend),
2. The sheet is not torn (initially joined faces remain joined), and
3. The sheet does not self-intersect.

A configuration that has the previous three properties is said to be a *valid configuration*. The examples presented in Figs. 2.1, 2.2, and 2.3 show origami sheets attaining valid configurations during their folding motion. As indicated in such figures, the only non-rigid body deformations of an origami sheet are achieved by rotating adjacent faces relative to one another along their connecting creased fold such that the sheet is not torn and does not self-intersect during such deformations. Thus, the configuration of an origami sheet is fully described by the only kinematic variable associated with a creased fold, which describes the relative rotation between the two faces that are joined by such a fold and is denoted as *fold angle*. The *fold angle* $\hat{\theta}_i(t)$ *is defined as* π *radians less the dihedral angle between the positive sides of the two faces joined by the* i*th creased fold*.

Schematics illustrating the concept of fold angle are provided in Fig. 2.4. We note that each fold angle is a function of the time parameter t. Specifically, each fold angle $\hat{\theta}_1(t), \ldots, \hat{\theta}_{N_\mathcal{F}}(t)$, where $N_\mathcal{F}$ *is the number of creased folds in the sheet*, is a *continuous* function with respect to the time parameter t since the motion of the sheet must be continuous. For the remainder of this book, the dependence of $\hat{\theta}_i$ on t is not shown explicitly to simplify notation.

Upon folding, *valley folds* attain fold angles greater than 0 while *mountain folds* attain fold angles lower than 0 as shown in Fig. 2.4. To prevent self-intersection of any pair of faces connected by a creased fold, the value of the associated fold angle must be contained in the interval $[-\pi, \pi]$.

2.3 Fold Pattern Description

In this section, we examine the geometric description of the *fold pattern*,[1] which is the layout of the creased folds in the reference configuration $\mathcal{S}_0$ of an origami sheet. We consider again the Miura-Ori sheet illustrated in Fig. 2.3 to help the reader

[1] Also referred to as *crease pattern* in the more restrictive context of origami with creased folds [27, 32, 33].

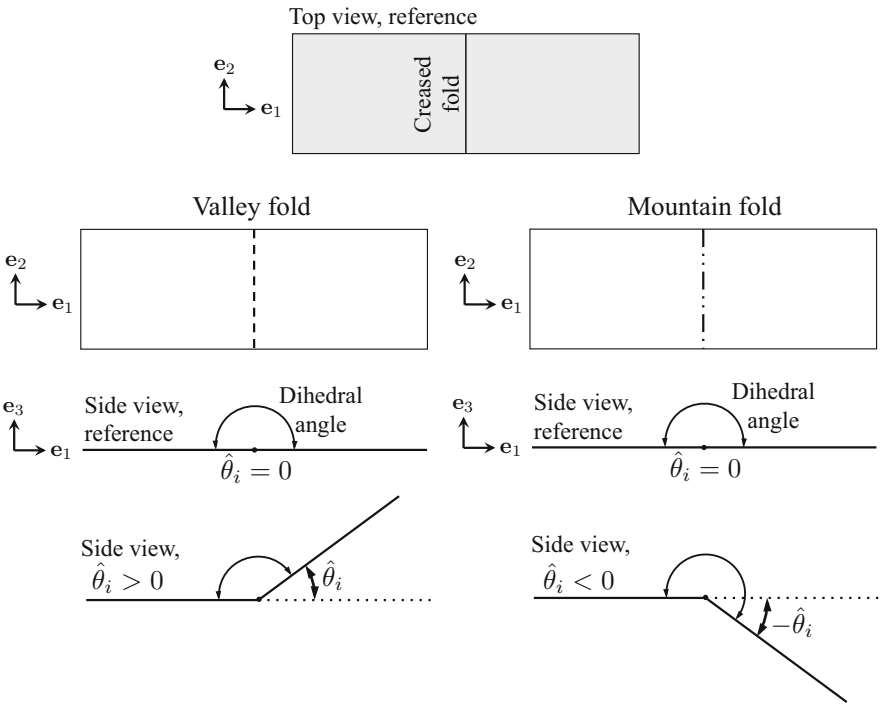


Fig. 2.4 Schematics showing reference and current configurations of valley and mountain creased folds and their adjacent faces. Refer to Figs. 2.1, 2.2, and 2.3 for mountain-valley assignments of different origami sheets

visualize the concepts introduced in this section. The reference configuration $\mathcal{S}_0$ of the Miura-Ori sheet is shown in more detail in Fig. 2.5. As previously stated, the *fold lines* are the line segments coincident with the creased folds in $\mathcal{S}_0$. Each fold line is defined by its end points, denoted as the *vertices* (Fig. 2.5a). Each vertex has an associated position vector denoted $\mathbf{v}^j \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ (Fig. 2.5b).

Example 2.1 Position vectors of the vertices of the Miura-Ori sheet (Fig. 2.5).
Statement: Based on Fig. 2.5, provide the components of the position vectors of each vertex in the Miura-Ori sheet.
Solution: The position vectors of the vertices $\mathbf{v}^1, \ldots, \mathbf{v}^5$ for the Miura-Ori sheet are given as follows:

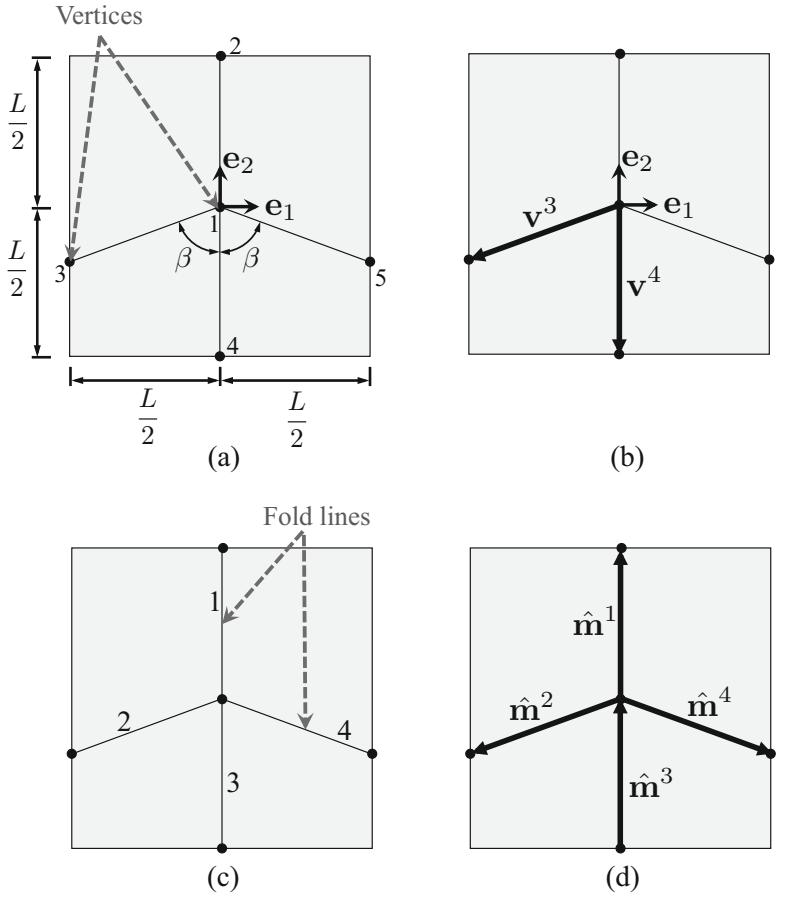


Fig. 2.5 Reference configuration $\mathcal{S}_0$ of a sheet with the Miura-Ori pattern illustrated in Fig. 2.3: (**a**) Numbering of the vertices and dimensions of the sheet; (**b**) Position vectors of the third and fourth vertices; (**c**) Numbering of the fold lines; (**d**) Fold vectors along the length of each fold line (the choice for the start point and end point of a given fold vector can be swapped and the same fold pattern would result, but the choice must be consistent throughout the various input parameters required to define the fold pattern)

$$\mathbf{v}^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{v}^2 = \begin{bmatrix} 0 \\ \frac{L}{2} \\ 0 \end{bmatrix}, \qquad \mathbf{v}^3 = \begin{bmatrix} -\frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix},$$
$$\mathbf{v}^4 = \begin{bmatrix} 0 \\ -\frac{L}{2} \\ 0 \end{bmatrix}, \qquad \mathbf{v}^5 = \begin{bmatrix} \frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix}. \tag{2.2}$$

The number of vertices located at the interior of $\mathcal{S}_0$ is denoted $N_{\mathcal{I}}$ and the number of vertices located at the boundary of $\mathcal{S}_0$ is denoted $N_{\mathcal{B}}$. For the Miura-Ori sheet in Fig. 2.5, $N_{\mathcal{I}} = 1$ and $N_{\mathcal{B}} = 4$. We enumerate the vertices starting from those located at the interior of $\mathcal{S}_0$ (with corresponding position vectors $\mathbf{v}^1, \ldots, \mathbf{v}^{N_{\mathcal{I}}}$) followed by those located at the boundary $\mathcal{S}_0$ (with corresponding position vectors $\mathbf{v}^{N_{\mathcal{I}}+1}, \ldots, \mathbf{v}^{N_{\mathcal{I}}+N_{\mathcal{B}}}$). This convention is observed in the enumeration of the vertices in Fig. 2.5a.

To identify which vertices are the start points and end points of each fold line in the fold pattern, we introduce the *fold connectivity matrix* $\mathbf{C}^{\mathcal{F}} \in \mathbb{R}^{N_{\mathcal{F}} \times 2}$ with components $C^{\mathcal{F}}_{ij}$ defined as follows:

$$\begin{aligned} C^{\mathcal{F}}_{i1} &= \text{Index of the vertex corresponding to the start point of the } i\text{th fold line,} \\ C^{\mathcal{F}}_{i2} &= \text{Index of the vertex corresponding to the end point of the } i\text{th fold line,} \\ i &= 1, \ldots, N_{\mathcal{F}}. \end{aligned} \tag{2.3}$$

The choice for the start point and end point of a given fold line can be swapped and the same fold pattern would result. However, such a choice must be consistent throughout the various input parameters required to define the fold pattern, which are presented subsequently in this section.

Example 2.2 Fold connectivity matrix for the Miura-Ori sheet (Fig. 2.5).
Statement: Based on Fig. 2.5, provide the fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$ for the Miura-Ori sheet.
Solution: The Miura-Ori sheet has four folds (i.e., $N_{\mathcal{F}} = 4$); thus, its associated fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$ is a 4×2 matrix having components given as follows:

$$\mathbf{C}^{\mathcal{F}} = \begin{bmatrix} 1 & 2 \\ 1 & 3 \\ 4 & 1 \\ 1 & 5 \end{bmatrix}. \tag{2.4}$$

Refer to Fig. 2.5c for the numbering of the folds.

Let $\hat{\mathbf{v}}^{i1}, \hat{\mathbf{v}}^{i2} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, $i = 1, \ldots, N_{\mathcal{F}}$, be the position vectors of the vertices from which each fold line in the sheet emanates and ends, respectively. These vectors are determined from the fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$ as follows (recall the definition of the fold connectivity matrix in (2.3)):

$$\hat{\mathbf{v}}^{i1} = \mathbf{v}^{C^{\mathcal{F}}_{i1}}, \qquad \hat{\mathbf{v}}^{i2} = \mathbf{v}^{C^{\mathcal{F}}_{i2}} \qquad\qquad i = 1, \ldots, N_{\mathcal{F}}. \tag{2.5}$$

Let $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_{\mathcal{F}}} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the *fold vectors*, which are those along the length of each fold line. These vectors are determined as follows (see Fig. 2.5d):

$$\hat{\mathbf{m}}^i = \hat{\mathbf{v}}^{i2} - \hat{\mathbf{v}}^{i1} \qquad\qquad i = 1, \ldots, N_{\mathcal{F}}. \tag{2.6}$$

Example 2.3 Fold parameters of the Miura-Ori sheet (Fig. 2.5).
Statement: Determine the position vectors of the start points and end points of the folds lines and the fold vectors of the Miura-Ori sheet.
Solution: Using the position vectors $\mathbf{v}^1, \ldots, \mathbf{v}^5$ from (2.2) and the fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$ from (2.4), we determine the position vectors of the start points and end points of folds 1 and 2 of the Miura-Ori sheet via (2.5) as follows:

$$\begin{aligned}
\hat{\mathbf{v}}^{11} &= \mathbf{v}^{C_{11}^{\mathcal{F}}} = \mathbf{v}^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} && \text{start point of fold 1,} \\
\hat{\mathbf{v}}^{12} &= \mathbf{v}^{C_{12}^{\mathcal{F}}} = \mathbf{v}^2 = \begin{bmatrix} 0 \\ \frac{L}{2} \\ 0 \end{bmatrix} && \text{end point of fold 1,} \\
\hat{\mathbf{v}}^{21} &= \mathbf{v}^{C_{21}^{\mathcal{F}}} = \mathbf{v}^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} && \text{start point of fold 2,} \\
\hat{\mathbf{v}}^{22} &= \mathbf{v}^{C_{22}^{\mathcal{F}}} = \mathbf{v}^3 = \begin{bmatrix} -\frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix} && \text{end point of fold 2.}
\end{aligned} \tag{2.7}$$

Similarly:

$$\begin{aligned}
\hat{\mathbf{v}}^{31} &= \mathbf{v}^{C_{31}^{\mathcal{F}}} = \mathbf{v}^4 = \begin{bmatrix} 0 \\ -\frac{L}{2} \\ 0 \end{bmatrix}, \qquad \hat{\mathbf{v}}^{32} = \mathbf{v}^{C_{32}^{\mathcal{F}}} = \mathbf{v}^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \\
\hat{\mathbf{v}}^{41} &= \mathbf{v}^{C_{41}^{\mathcal{F}}} = \mathbf{v}^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \hat{\mathbf{v}}^{42} = \mathbf{v}^{C_{42}^{\mathcal{F}}} = \mathbf{v}^5 = \begin{bmatrix} \frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix}.
\end{aligned} \tag{2.8}$$

The fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^4$ are determined via (2.6) (refer to Fig. 2.5d):

$$\begin{aligned}
\hat{\mathbf{m}}^1 &= \hat{\mathbf{v}}^{12} - \hat{\mathbf{v}}^{11} = \begin{bmatrix} 0 \\ \frac{L}{2} \\ 0 \end{bmatrix} && \text{vector along the length of fold line 1,} \\
\hat{\mathbf{m}}^2 &= \hat{\mathbf{v}}^{22} - \hat{\mathbf{v}}^{21} = \begin{bmatrix} -\frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix} && \text{vector along the length of fold line 2.}
\end{aligned} \tag{2.9}$$

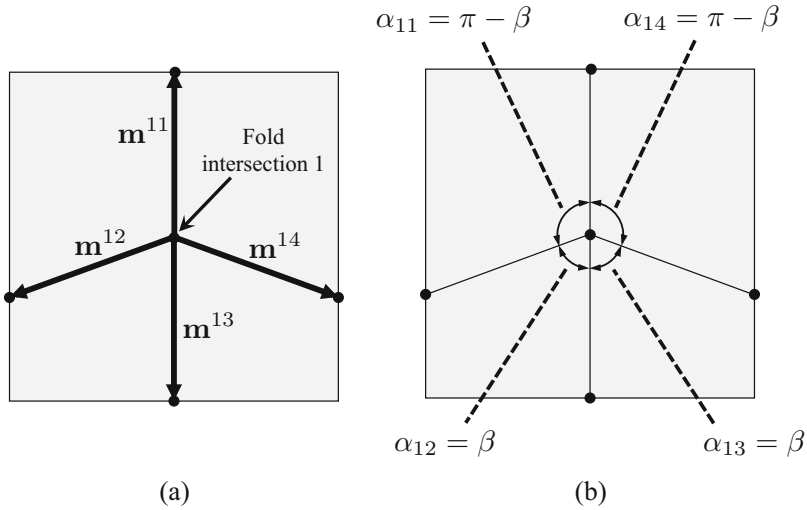


Fig. 2.6 Parameters associated with the interior fold intersection of the Miura-Ori sheet: (**a**) Vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ along the length of the fold lines incident to the interior fold intersection and that emanate from the intersection; (**b**) Face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ around the interior fold intersection

Similarly:

$$\hat{\mathbf{m}}^3 = \hat{\mathbf{v}}^{32} - \hat{\mathbf{v}}^{31} = \begin{bmatrix} 0 \\ \frac{L}{2} \\ 0 \end{bmatrix}, \qquad \hat{\mathbf{m}}^4 = \hat{\mathbf{v}}^{42} - \hat{\mathbf{v}}^{41} = \begin{bmatrix} \frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix}. \tag{2.10}$$

Let $n_1, \ldots, n_{N_\mathcal{I}}$ be the number of fold lines incident to each *interior fold intersection* of $\mathcal{S}_0$ (corresponding to the interior vertices having position vectors $\mathbf{v}^1, \ldots, \mathbf{v}^{N_\mathcal{I}}$). Also, let $\mathbf{m}^{jk} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, $j = 1, \ldots, N_\mathcal{I}$, $k = 1, \ldots, n_j$, be the vector along the length of the kth fold line incident to the jth interior fold intersection that emanates from such an intersection. The vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ associated with the interior fold intersection of the Miura-Ori sheet shown in Fig. 2.5 are illustrated in Fig. 2.6a.

For the jth interior fold intersection, the associated vectors $\mathbf{m}^{j1}, \ldots, \mathbf{m}^{jn_j}$ are arranged in counterclockwise order (see Figs. 2.6 and 2.7). The *fold intersection connectivity matrix* $\mathbf{C}^\mathcal{I} \in \mathbb{R}^{N_\mathcal{I} \times \max(n_j)}$ is used for the identification and ordering of the folds incident to the interior fold intersections and its components $C^\mathcal{I}_{jk}$ are defined as follows:

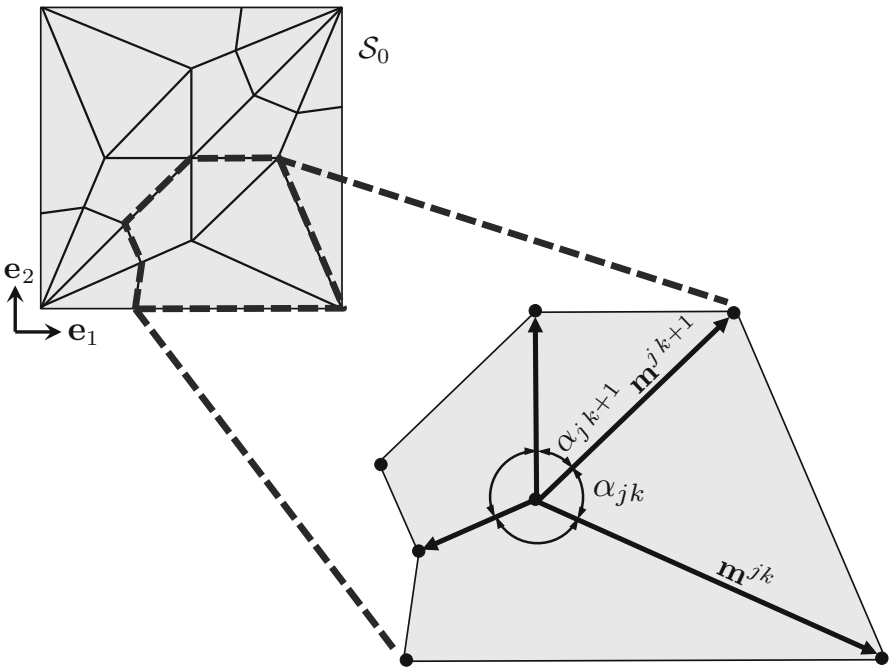


Fig. 2.7 Schematic showing faces and folds connected to an interior fold intersection and their associated geometric parameters. The overall domain $\mathcal{S}_0$ has 11 interior vertices and 8 boundary vertices ($N_{\mathcal{I}} = 11$, $N_{\mathcal{B}} = 8$)

$$C^{\mathcal{I}}_{jk} = \text{Index of the } k\text{th fold line incident to the } j\text{th interior fold intersection (multiplied by } -1 \text{ if the fold line ends at the interior fold intersection),}$$
$$j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots n_j. \tag{2.11}$$

The mapping from the fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_{\mathcal{F}}}$ (see (2.6)) to the vectors $\mathbf{m}^{j1}, \ldots, \mathbf{m}^{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$, is given as follows:

$$\mathbf{m}^{jk} = \begin{cases} \hat{\mathbf{m}}^{C^{\mathcal{I}}_{jk}}; & C^{\mathcal{I}}_{jk} > 0 \\ -\hat{\mathbf{m}}^{|C^{\mathcal{I}}_{jk}|}; & C^{\mathcal{I}}_{jk} < 0, \end{cases} \qquad j = 1, \ldots, N_{\mathcal{I}}, \; k = 1, \ldots, n_j. \tag{2.12}$$

Note that $\mathbf{m}^{jk}$ has the opposite direction of its associated fold vector $\hat{\mathbf{m}}^{|C^{\mathcal{I}}_{jk}|}$ if $C^{\mathcal{I}}_{jk} < 0$. This is applied such that the vectors $\mathbf{m}^{jk}$ always emanate from the interior fold intersection (since $C^{\mathcal{I}}_{jk} < 0$ if the considered fold line *ends* at the interior fold intersection, recall the definition of $\mathbf{C}^{\mathcal{I}}$ in (2.11)).

Let us define $\varphi(\mathbf{y}, \mathbf{z})$ as the angle from a vector $\mathbf{z} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ to a vector $\mathbf{y} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ that is determined as follows[2]:

$$\varphi(\mathbf{y}, \mathbf{z}) = \begin{cases} \cos^{-1}\left(\dfrac{\mathbf{y} \cdot \mathbf{z}}{\|\mathbf{y}\| \|\mathbf{z}\|}\right); & (\mathbf{e}_3 \times \mathbf{z}) \cdot \mathbf{y} \geq 0 \\ 2\pi - \cos^{-1}\left(\dfrac{\mathbf{y} \cdot \mathbf{z}}{\|\mathbf{y}\| \|\mathbf{z}\|}\right); & (\mathbf{e}_3 \times \mathbf{z}) \cdot \mathbf{y} < 0. \end{cases} \tag{2.13}$$

To simplify the notation, if only one argument is provided in the function φ, it is implicitly assumed that $\mathbf{z} = \mathbf{e}_1$ (i.e., $\varphi(\mathbf{y}) = \varphi(\mathbf{y}, \mathbf{e}_1)$).

The face corner angles surrounding each interior fold intersection are denoted as $\alpha_{j1}, \ldots, \alpha_{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$, and are calculated as follows (see Fig. 2.7):

$$\alpha_{jk} = \begin{cases} \varphi(\mathbf{m}^{j\,k+1}, \mathbf{m}^{jk}); & k = 1, \ldots, n_j - 1 \\ \varphi(\mathbf{m}^{j1}, \mathbf{m}^{jk}); & k = n_j. \end{cases} \tag{2.14}$$

The face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ surrounding the interior fold intersection of the Miura-Ori sheet are shown in Fig. 2.6b.

Example 2.4 Fold intersection parameters for the Miura-Ori sheet (Figs. 2.5 and 2.6).
Statement: Provide the fold intersection connectivity matrix $\mathbf{C}^{\mathcal{I}}$ for the Miura-Ori sheet shown in Figs. 2.5 and 2.6. Then, determine the vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ and the face corner angles $\alpha_{11}, \ldots, \alpha_{14}$.
Solution: The fold intersection connectivity matrix $\mathbf{C}^{\mathcal{I}}$ for the Miura-Ori sheet is a 1×4 matrix with components given as follows:

$$\mathbf{C}^{\mathcal{I}} = \begin{bmatrix} 1 & 2 & -3 & 4 \end{bmatrix}. \tag{2.15}$$

Using the fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^4$ determined in Example 2.3, the vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ are determined via (2.12) as follows (refer to Fig. 2.6a):

[2] $\|\cdot\|$ denotes the 2-norm, i.e. $\|\mathbf{y}\| = (\mathbf{y} \cdot \mathbf{y})^{\frac{1}{2}}$.

$$
\begin{aligned}
\mathbf{m}^{11} &= \hat{\mathbf{m}}^{C_{11}^{\mathcal{I}}} = \hat{\mathbf{m}}^{1} = \begin{bmatrix} 0 \\ \frac{L}{2} \\ 0 \end{bmatrix}, \\
\mathbf{m}^{12} &= \hat{\mathbf{m}}^{C_{12}^{\mathcal{I}}} = \hat{\mathbf{m}}^{2} = \begin{bmatrix} -\frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix}, \\
\mathbf{m}^{13} &= -\hat{\mathbf{m}}^{\left|C_{13}^{\mathcal{I}}\right|} = -\hat{\mathbf{m}}^{|-3|} = -\hat{\mathbf{m}}^{3} = -\begin{bmatrix} 0 \\ \frac{L}{2} \\ 0 \end{bmatrix}, \\
\mathbf{m}^{14} &= \hat{\mathbf{m}}^{C_{14}^{\mathcal{I}}} = \hat{\mathbf{m}}^{4} = \begin{bmatrix} \frac{L}{2} \\ -\frac{L}{2\tan(\beta)} \\ 0 \end{bmatrix}.
\end{aligned}
\tag{2.16}
$$

The face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ are determined using (2.14) (refer to Fig. 2.6b):

$$
\begin{aligned}
\alpha_{11} &= \varphi\left(\mathbf{m}^{12}, \mathbf{m}^{11}\right) = \pi - \beta, \qquad & \alpha_{12} &= \varphi\left(\mathbf{m}^{13}, \mathbf{m}^{12}\right) = \beta, \\
\alpha_{13} &= \varphi\left(\mathbf{m}^{14}, \mathbf{m}^{13}\right) = \beta, & \alpha_{14} &= \varphi\left(\mathbf{m}^{11}, \mathbf{m}^{14}\right) = \pi - \beta.
\end{aligned}
\tag{2.17}
$$

We summarize the input and calculated parameters required to define the fold pattern of an origami sheet with creased folds in Tables 2.1 and 2.2.

Table 2.1 Input parameters required to define the fold pattern of an origami sheet with creased folds

Parameter	Definition
Position vectors of the vertices $\mathbf{v}^1, \ldots, \mathbf{v}^{N_{\mathcal{I}}+N_{\mathcal{B}}}$	
Fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$	(2.3)
Fold intersection connectivity matrix $\mathbf{C}^{\mathcal{I}}$	(2.11)

Table 2.2 Calculated parameters required to define the fold pattern of an origami sheet with creased folds

Parameter	Equation
Fold line start points $\hat{\mathbf{v}}^{11}, \ldots, \hat{\mathbf{v}}^{N_{\mathcal{F}}1}$	(2.5)
Fold line end points $\hat{\mathbf{v}}^{12}, \ldots, \hat{\mathbf{v}}^{N_{\mathcal{F}}2}$	(2.5)
Fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_{\mathcal{F}}}$	(2.6)
Vectors along fold lines incident to interior fold intersections $\mathbf{m}^{j1}, \ldots, \mathbf{m}^{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$	(2.12)
Face corner angles $\alpha_{j1}, \ldots, \alpha_{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$	(2.14)

2.4 Kinematic Constraints for Origami with Creased Folds

In this section, we describe the fundamental kinematic constraints for origami with creased folds. These are the *developability constraint* (Sect. 2.4.1) and the *loop closure constraint* (Sect. 2.4.2).

2.4.1 Developability Constraint

After the geometry of the fold pattern is defined, constraints on the fold kinematic variables (corresponding to the fold angles for creased folds) are formulated such that every current configuration attained by an origami sheet is valid according to Sect. 2.2.[3] In addition to constraints allowing for valid configurations (to be addressed in Sect. 2.4.2), the condition of *developability* [34] is also conventionally imposed in origami.

A developable surface has zero Gaussian curvature everywhere [35]. Developability allows a surface to be flattened onto a plane without stretching or overlapping. Since valid configurations of the sheet consist of planar faces joined at straight creased folds, the only locations where the Gaussian curvature is non-trivially zero is at the singular points corresponding to the interior fold intersections. The conventional differential geometry definition of Gaussian curvature as the product of the two principal curvatures is not applicable at such singular interior fold intersections [36] and therefore the *discrete Gaussian curvature*, denoted K_j, is considered [36–38]. It is given as 2π radians less the sum of the face corner angles surrounding each interior fold intersection [39]. *For any interior fold intersection in the sheet, the discrete Gaussian curvature must be zero for it to be developable*:

$$K_j = 2\pi - \sum_{k=1}^{n_j} \alpha_{jk} = 0 \qquad j = 1, \ldots, N_{\mathcal{I}}, \tag{2.18}$$

where the face corner angles α_{jk} are calculated using (2.14). In the model presented in this chapter, the face corner angles α_{jk} are defined in the reference configuration $\mathcal{S}_0$, which is planar and free of face overlaps. Thus, these angles sum to 2π for each interior fold intersection and the developability constraint (2.18) is satisfied in $\mathcal{S}_0$. No further consideration of this constraint is required because the face corner angles are constant during the motion of the sheet (since the faces undergo only rigid

[3] Self-intersection avoidance is an essential restriction in origami as stated in Sect. 2.2. It remains an open problem to provide rigorous mathematical constraints on fold angles that would allow for general three-dimensional folded configurations free of self-intersection [27, 33]. This restriction is thus not considered.

deformations for valid configurations) and thus they hold their associated values α_{jk} as defined in $\mathcal{S}_0$.

Example 2.5 Developability of the Miura-Ori sheet (Fig. 2.6).
Statement: Verify that the developability constraint is satisfied for the Miura-Ori sheet shown in Fig. 2.6.
Solution: We substitute the face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ determined in (2.17) into the developability constraint (2.18):

$$\begin{aligned} K_j &= 2\pi - \sum_{k=1}^{n_j} \alpha_{jk} \\ &= 2\pi - (\alpha_{11} + \alpha_{12} + \alpha_{13} + \alpha_{14}) \\ &= 2\pi - \big((\pi - \beta) + \beta + \beta + (\pi - \beta)\big) \\ &= 2\pi - 2\pi \;=\; 0. \end{aligned} \tag{2.19}$$

Thus, the developability constraint (2.18) is satisfied for the Miura-Ori sheet.

2.4.2 Loop Closure Constraint

Following the process presented by Belcastro and Hull in [32, 33] to formulate constraints on the fold angles allowing for valid configurations, the *mapping* from the reference to current configurations considering only the faces adjacent to an interior fold intersection is first derived. This represents a kind of "local" case, in which only the faces adjacent to a particular interior fold intersection of a more complex fold pattern are examined [32, 33]. The goal is to determine the constraints that allow any such set of faces to attain valid configurations.

The fold angle associated with the kth fold incident to the jth interior fold intersection is denoted θ_{jk}, $j = 1, \ldots, N_{\mathcal{I}}$, $k = 1, \ldots, n_j$. The mapping from all the fold angles in the sheet ($\hat{\theta}_1, \ldots, \hat{\theta}_{N_{\mathcal{F}}}$; see Sect. 2.2) to the fold angles of only the folds incident to the jth interior fold intersection ($\theta_{j1}, \ldots, \theta_{jn_j}$) is given as follows:

$$\theta_{jk} = \hat{\theta}_{\left|C^{\mathcal{I}}_{jk}\right|} \qquad j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots, n_j, \tag{2.20}$$

where components of the fold intersection connectivity matrix $\mathbf{C}^{\mathcal{I}}$ are defined in (2.11).

Let $\boldsymbol{\gamma}^j(\eta) : [0, 1] \to \mathcal{S}_0$ be an arbitrary simple closed path enclosing the jth interior fold intersection and crossing each of its incident folds once as illustrated in Fig. 2.8. The point having position $\boldsymbol{\gamma}^j(0) = \boldsymbol{\gamma}^j(1)$ is defined such that it is located

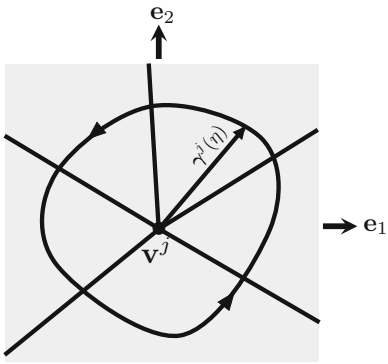


Fig. 2.8 Path $\boldsymbol{\gamma}^j(\eta)$ crossing the creased folds incident to the jth interior fold intersection

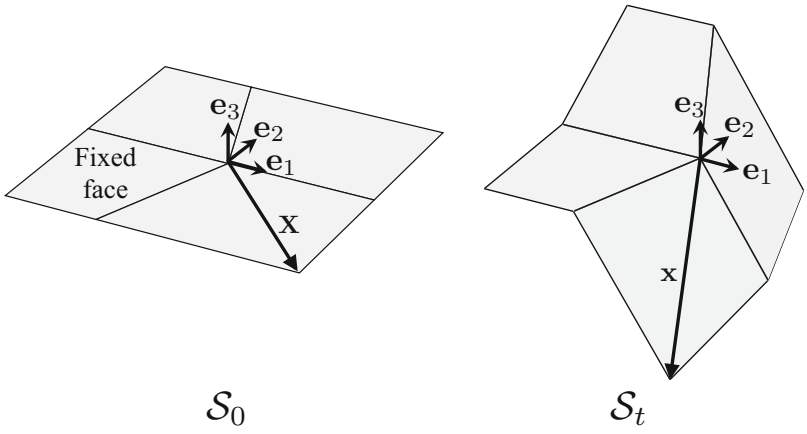


Fig. 2.9 Position vector $\mathbf{X}$ of a point in the reference configuration $\mathcal{S}_0$ and position vector $\mathbf{x}$ of the same point in a current configuration $\mathcal{S}_t$

at the face adjacent to the creased folds with corresponding vectors $\mathbf{m}^{j1}$ and $\mathbf{m}^{jn_j}$. Also, the path $\boldsymbol{\gamma}^j(\eta)$ is defined such that it crosses the creased folds with associated vectors $\mathbf{m}^{jk}$ in counterclockwise order (i.e., $\mathbf{m}^{j1}, \mathbf{m}^{j2}, \ldots, \mathbf{m}^{jn_j}$). The following assumptions are made to simplify the derivation of kinematic constraints for the case of a single interior fold intersection: (1) the vertex at the fold intersection is assumed to be located at the origin and (2) the face containing the point with position $\boldsymbol{\gamma}^j(0)$ is assumed *fixed* in space (not translating or rotating).

Let $\mathbf{X} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the position vector of a point in a face in the reference configuration $\mathcal{S}_0$ and let $\mathbf{x} \in \mathbb{R}^3$ be the position vector of the same point in a current configuration $\mathcal{S}_t$. This is illustrated in Fig. 2.9.

To define the mapping between reference and current configurations of the faces crossed by the path $\boldsymbol{\gamma}^j(\eta)$ (i.e., the map $\mathbf{X} \mapsto \mathbf{x}$), we must define certain rotation matrices first. Let $\mathbf{R}_1(\phi) \in \mathbb{R}^{3\times 3}$ be the transformation matrix associated with a rotation by ϕ radians about an axis of rotation aligned to $\mathbf{e}_1$:

$$\mathbf{R}_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}, \tag{2.21}$$

and $\mathbf{R}_3(\phi) \in \mathbb{R}^{3\times 3}$ be the transformation matrix associated with a rotation by ϕ radians about an axis of rotation aligned to $\mathbf{e}_3$:

$$\mathbf{R}_3(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.22}$$

Considering an axis aligned with a vector $\mathbf{y} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, the transformation associated with a rotation by ϕ about such an axis can be represented as follows [32, 33] (recall the definition of $\varphi(\cdot)$ in (2.13)):

$$\mathbf{R}_3(\varphi(\mathbf{y}))\,\mathbf{R}_1(\phi)\,\mathbf{R}_3^{-1}(\varphi(\mathbf{y})). \tag{2.23}$$

In this transformation, the axis of rotation aligned with vector $\mathbf{y}$ is first aligned to $\mathbf{e}_1$ via the rotation matrix $\mathbf{R}_3^{-1}(\varphi(\mathbf{y}))$. Then, a rotation of ϕ about $\mathbf{e}_1$ is performed via the rotation matrix $\mathbf{R}_1(\phi)$. Finally, the axis of rotation is aligned back to its original orientation through the rotation matrix $\mathbf{R}_3(\varphi(\mathbf{y}))$.

Example 2.6 Matrix associated with a rotation about an axis aligned with $\mathbf{y} = [\ 1\ \ 1\ \ 0]^\top$.
Statement: Determine the transformation matrix associated with a rotation of ϕ about an axis aligned with $\mathbf{y} = [\ 1\ \ 1\ \ 0]^\top$.
Solution: We can derive the transformation matrix associated with such a rotation using (2.23):

$$\mathbf{R}_3(\varphi(\mathbf{y}))\,\mathbf{R}_1(\phi)\,\mathbf{R}_3^{-1}(\varphi(\mathbf{y})) = \mathbf{R}_3(\tfrac{\pi}{4})\,\mathbf{R}_1(\phi)\,\mathbf{R}_3^{-1}(\tfrac{\pi}{4}), \tag{2.24}$$

where $\mathbf{R}_3(\frac{\pi}{4})$ is determined using (2.22):

$$\mathbf{R}_3(\tfrac{\pi}{4}) = \begin{bmatrix} \frac{2^{1/2}}{2} & -\frac{2^{1/2}}{2} & 0 \\ \frac{2^{1/2}}{2} & \frac{2^{1/2}}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.25}$$

Substituting (2.25) into (2.24), we determine the explicit form of the transformation matrix associated with a rotation of ϕ about an axis aligned with $\mathbf{y} = [\ 1\ \ 1\ \ 0]^\top$:

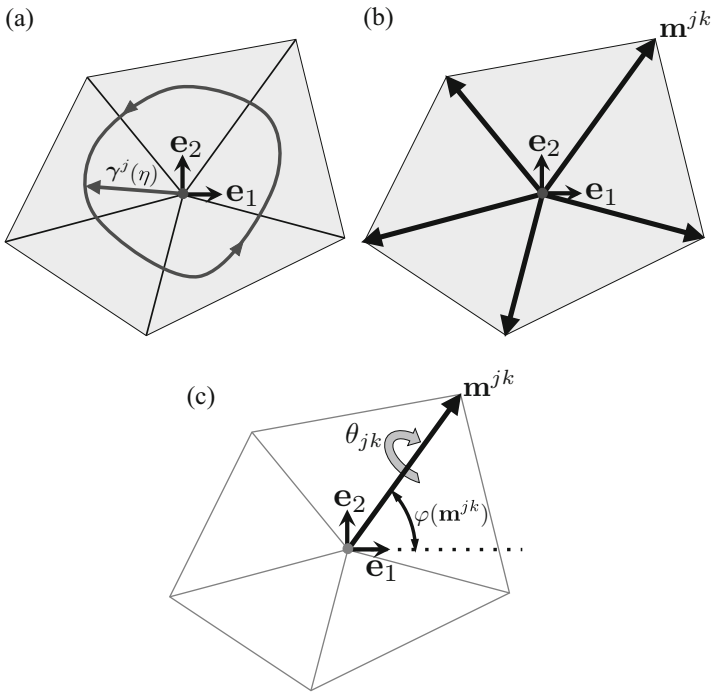


Fig. 2.10 (**a**) Path $\boldsymbol{\gamma}^j(\eta)$ crossing the creased folds incident an interior fold intersection. (**b**) Vector $\mathbf{m}^{jk}$ along the kth fold crossed by $\boldsymbol{\gamma}^j(\eta)$. (**c**) Transformation described by (2.27): Rotation of θ_{jk} about an axis of rotation aligned to $\mathbf{m}^{jk}$

$$\mathbf{R}_3(\tfrac{\pi}{4})\,\mathbf{R}_1(\phi)\,\mathbf{R}_3^{-1}(\tfrac{\pi}{4}) = \begin{bmatrix} \frac{1}{2}+\frac{1}{2}\cos(\phi) & \frac{1}{2}-\frac{1}{2}\cos(\phi) & \frac{2^{1/2}}{2}\sin(\phi) \\ \frac{1}{2}-\frac{1}{2}\cos(\phi) & \frac{1}{2}+\frac{1}{2}\cos(\phi) & -\frac{2^{1/2}}{2}\sin(\phi) \\ -\frac{2^{1/2}}{2}\sin(\phi) & \frac{2^{1/2}}{2}\sin(\phi) & \cos(\phi) \end{bmatrix}. \tag{2.26}$$

We can use (2.23) to represent the transformation associated with folding a creased fold. The transformation representing the deformation associated with the folding of the kth creased fold crossed by the path $\boldsymbol{\gamma}^j(\eta)$ can be described as a rotation of θ_{jk} about an axis of rotation aligned to $\mathbf{m}^{jk}$ as follows (refer to Fig. 2.10):

$$\mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk})). \tag{2.27}$$

The mapping $\mathbf{X} \mapsto \mathbf{x}$ is constructed as the composition of such transformations associated with the folds crossed by the segment of path $\boldsymbol{\gamma}^j(\eta)$ that connects $\boldsymbol{\gamma}^j(0)$ to the face containing the point with initial position $\mathbf{X}$:

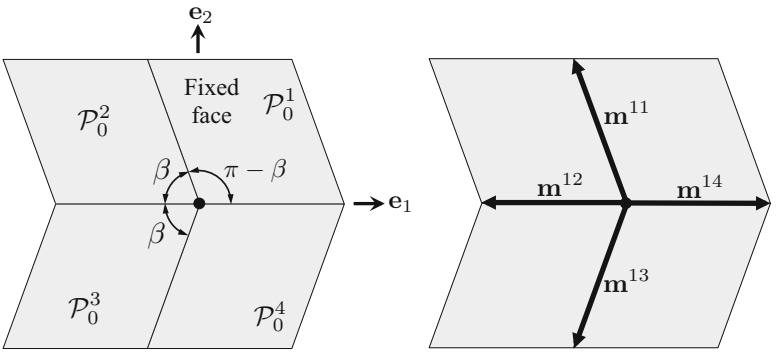


Fig. 2.11 Schematic for Example 2.7 corresponding to a Miura-Ori unit cell

$$\mathbf{x} = \left(\prod_{k=1}^{n_\gamma} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1(\theta_{jk})\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk})) \right) \mathbf{X}, \tag{2.28}$$

where n_γ *is the number of folds crossed by the segment of the path* $\boldsymbol{\gamma}^j(\eta)$ *connecting* $\boldsymbol{\gamma}^j(0)$ *and the face containing the point with position vector* $\mathbf{X}$. Note that $\mathbf{x}$ is the position vector of such a point in a current configuration fully determined by $\theta_{j1} \ldots, \theta_{jn_j}$.

Example 2.7 Determining the mapping between reference and current configurations of a Miura-Ori sheet.
Statement: Considering the fold pattern presented in Fig. 2.11 (a unit cell in a Miura-Ori tessellation), determine the mapping between reference and current configurations in the form of (2.28) for each face shown in the figure.
Solution: From the schematic in Fig. 2.11, the angles $\varphi(\mathbf{m}^{11})$, $\varphi(\mathbf{m}^{12})$, $\varphi(\mathbf{m}^{13})$, $\varphi(\mathbf{m}^{14})$ are given as: $\varphi(\mathbf{m}^{11}) = \pi - \beta$, $\varphi(\mathbf{m}^{12}) = \pi$, $\varphi(\mathbf{m}^{13}) = \pi + \beta$, $\varphi(\mathbf{m}^{14}) = 0$. The transformation matrices in the form of (2.27) associated with each of the folds in Fig. 2.11 can be given as:

$$\begin{aligned}
&\text{Fold between } \mathcal{P}_0^1 \text{ and } \mathcal{P}_0^2: \quad \mathbf{R}_3(\pi - \beta)\, \mathbf{R}_1(\theta_{11})\, \mathbf{R}_3^{-1}(\pi - \beta),\\
&\text{Fold between } \mathcal{P}_0^2 \text{ and } \mathcal{P}_0^3: \quad \mathbf{R}_3(\pi)\, \mathbf{R}_1(\theta_{12})\, \mathbf{R}_3^{-1}(\pi),\\
&\text{Fold between } \mathcal{P}_0^3 \text{ and } \mathcal{P}_0^4: \quad \mathbf{R}_3(\pi + \beta)\, \mathbf{R}_1(\theta_{13})\, \mathbf{R}_3^{-1}(\pi + \beta),\\
&\text{Fold between } \mathcal{P}_0^4 \text{ and } \mathcal{P}_0^1: \quad \mathbf{R}_3(0)\, \mathbf{R}_1(\theta_{14})\, \mathbf{R}_3^{-1}(0).
\end{aligned} \tag{2.29}$$

Therefore, the mappings in the form of (2.28) are determined as follows for $\mathbf{X} \in \mathcal{P}_0^2$:

$$\begin{aligned}\mathbf{x} &= \left(\prod_{k=1}^{1} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\right)\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3^{-1}(\pi-\beta)\,\mathbf{X},\end{aligned} \tag{2.30}$$

for $\mathbf{X} \in \mathcal{P}_0^3$:

$$\begin{aligned}\mathbf{x} &= \left(\prod_{k=1}^{2} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\right)\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3^{-1}(\pi-\beta) \\ &\quad\times \mathbf{R}_3(\pi)\,\mathbf{R}_1(\theta_{12})\,\mathbf{R}_3^{-1}(\pi)\,\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{12})\,\mathbf{R}_3^{-1}(\pi)\,\mathbf{X},\end{aligned} \tag{2.31}$$

for $\mathbf{X} \in \mathcal{P}_0^4$:

$$\begin{aligned}\mathbf{x} &= \left(\prod_{k=1}^{3} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\right)\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3^{-1}(\pi-\beta) \\ &\quad\times \mathbf{R}_3(\pi)\,\mathbf{R}_1(\theta_{12})\,\mathbf{R}_3^{-1}(\pi) \\ &\quad\times \mathbf{R}_3(\pi+\beta)\,\mathbf{R}_1(\theta_{13})\,\mathbf{R}_3^{-1}(\pi+\beta)\,\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{12}) \\ &\quad\times \mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{13})\,\mathbf{R}_3(\pi-\beta)\,\mathbf{X},\end{aligned} \tag{2.32}$$

and for $\mathbf{X} \in \mathcal{P}_0^1$:

$$\begin{aligned}\mathbf{x} &= \left(\prod_{k=1}^{4} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\right)\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3^{-1}(\pi-\beta) \\ &\quad\times \mathbf{R}_3(\pi)\,\mathbf{R}_1(\theta_{12})\,\mathbf{R}_3^{-1}(\pi) \\ &\quad\times \mathbf{R}_3(\pi+\beta)\,\mathbf{R}_1(\theta_{13})\,\mathbf{R}_3^{-1}(\pi+\beta) \\ &\quad\times \mathbf{R}_3(0)\,\mathbf{R}_1(\theta_{14})\,\mathbf{R}_3^{-1}(0)\,\mathbf{X} \\ &= \mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{12}) \\ &\quad\times \mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{13})\,\mathbf{R}_3(\pi-\beta)\,\mathbf{R}_1(\theta_{14})\,\mathbf{X}.\end{aligned} \tag{2.33}$$

As observed in (2.33), the mapping of the position vector for a point in $\mathcal{P}_0^1$ can be expressed as a function of the fold angles θ_{11}, θ_{12}, θ_{13}, and θ_{14}. However, it is specified here that points in $\mathcal{P}_0^1$ remain *fixed* upon folding (i.e., $\mathbf{x} = \mathbf{X}$ for $\mathbf{X} \in \mathcal{P}_0^1$). Therefore, the fold angles $\theta_{11}, \theta_{12}, \theta_{13}$, and θ_{14} are *not* independent from one another and certain constraints must hold to ensure that points in $\mathcal{P}_0^1$ are fixed. It is observed that $\mathbf{x} = \mathbf{X}$ for $\mathbf{X} \in \mathcal{P}_0^1$ when the product of the rotation matrices multiplying $\mathbf{X}$ in (2.33) is equal to the identity matrix. Thus, the fold angles θ_{11}, θ_{12}, θ_{13}, and θ_{14} for the Miura-Ori unit cell shown in Fig. 2.11 must be subject to the following constraint:

$$\begin{aligned}\mathbf{I}_3 = \mathbf{R}_3(\pi - \beta)\,\mathbf{R}_1(\theta_{11})\,\mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{12}) \\ \times \mathbf{R}_3(\beta)\,\mathbf{R}_1(\theta_{13})\,\mathbf{R}_3(\pi - \beta)\,\mathbf{R}_1(\theta_{14}),\end{aligned} \tag{2.34}$$

where $\mathbf{I}_n$ denotes the identity matrix in $\mathbb{R}^{n\times n}$. If the constraint in (2.34) does not hold, the mapping between reference and current configurations would not be spatially continuous. For instance, the mapping of the points at the edge of $\mathcal{P}_0^1$ corresponding to the crease between $\mathcal{P}_0^1$ and $\mathcal{P}_0^4$ (highlighted in Fig. 2.12a) could be given as $\mathbf{x} = \mathbf{X}$ (since it is specified that points in $\mathcal{P}_0^1$ are fixed) or through (2.33), which does not result in $\mathbf{x} = \mathbf{X}$ if the constraint in (2.34) is not satisfied. In that case where (2.34) is not satisfied, the sheet will be *torn* at such an edge in a current configuration as shown in Fig. 2.12c and thus it would not be a valid configuration (see Sect. 2.2).

This same approach used to determine the constraints on the fold angles for the Miura-Ori unit cell in (2.34) is likewise used to determine constraints on the fold angles for any general interior fold intersection with an arbitrary number of incident creased folds.

As noted in Example 2.7, to prevent tearing between the faces joined to the jth interior fold intersection (i.e., to allow the mapping between reference and current configurations in (2.28) to be spatially continuous), a constraint on the fold angles $\theta_{j1} \ldots, \theta_{jn_j}$ must be imposed. Such a constraint is derived by noting that the mapping between reference and current configurations formulated in (2.28) must result in $\mathbf{x} = \mathbf{X}$ for any point in the fixed face (i.e., if we substitute $n_\gamma = 0$ or $n_\gamma = n_j$ in (2.28), then $\mathbf{x} = \mathbf{X}$). This requires the following:

$$\prod_{k=1}^{n_j} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\mathbf{R}_1(\theta_{jk})\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk})) = \mathbf{I}_3. \tag{2.35}$$

We can simplify (2.35) by noting that:

$$\begin{aligned}\mathbf{I}_3 &= \prod_{k=1}^{n_j} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\mathbf{R}_1(\theta_{jk})\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk})) \\ &= \mathbf{R}_3(\varphi(\mathbf{m}^{j1}))\left(\prod_{k=1}^{n_j-1} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk})\right)\mathbf{R}_1(\theta_{jn_j})\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jn_j})),\end{aligned} \tag{2.36}$$

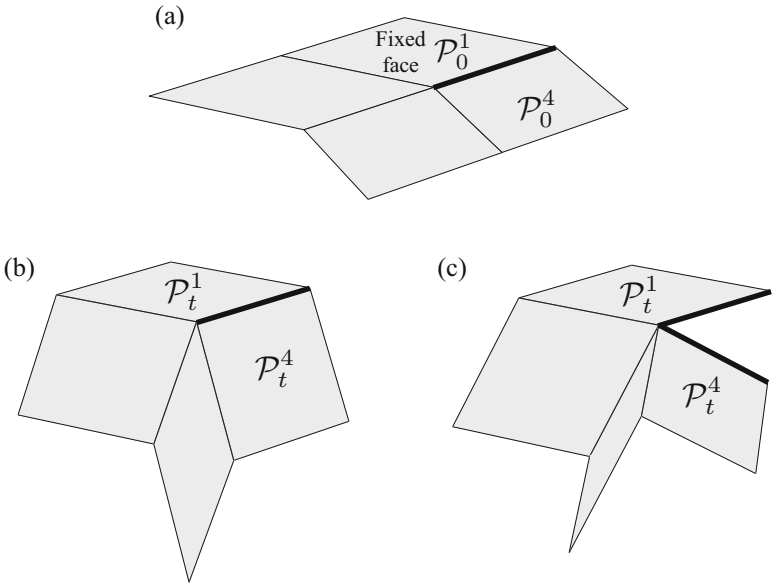


Fig. 2.12 (**a**) Reference configuration of a Miura-Ori unit cell. (**b**) Current configuration for which the constraint in (2.34) is satisfied. (**c**) Current configuration for which the constraint in (2.34) is *not* satisfied and the sheet is *torn*

where the following equality was used (refer to the definition of α_{jk} provided in (2.14)):

$$\mathbf{R}_3(\alpha_{jk}) = \begin{cases} \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_3(\varphi(\mathbf{m}^{j\,k+1})); & k = 1, \ldots, n_j - 1 \\ \\ \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{R}_3(\varphi(\mathbf{m}^{j1})); & k = n_j. \end{cases} \tag{2.37}$$

Multiplying the last expression in (2.36) by $\mathbf{R}_3^{-1}\big(\varphi(\mathbf{m}^{j1})\big)$ from the left and by $\mathbf{R}_3\big(\varphi(\mathbf{m}^{j1})\big)$ from the right, the following is obtained:

$$\begin{aligned} \mathbf{I}_3 &= \left(\prod_{k=1}^{n_j-1} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk})\right) \mathbf{R}_1(\theta_{jn_j})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jn_j}))\,\mathbf{R}_3(\varphi(\mathbf{m}^{j1})) \\ &= \left(\prod_{k=1}^{n_j-1} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk})\right) \mathbf{R}_1(\theta_{jn_j})\,\mathbf{R}_3(\alpha_{jn_j}) \\ &= \prod_{k=1}^{n_j} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk}). \end{aligned} \tag{2.38}$$

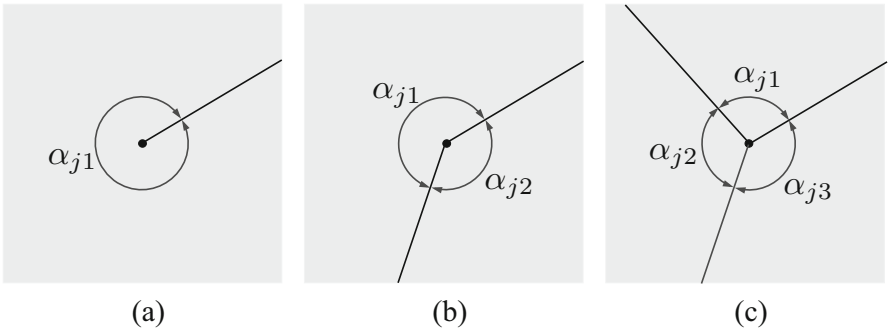


Fig. 2.13 Interior fold intersection having (**a**) one incident fold, (**b**) two incident folds, and (**c**) three incident folds

We then obtain the final form of the kinematic constraint for origami with creased folds [9, 27]:

$$\mathbf{R}^j = \prod_{k=1}^{n_j} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk}) = \mathbf{I}_3 \qquad j = 1, \ldots, N_{\mathcal{I}}. \tag{2.39}$$

In Problem 2.3, the reader is asked to derive the kinematic constraint for the folds adjacent to an interior fold intersection using a path $\boldsymbol{\gamma}^j(\eta)$ oriented clockwise as opposed to the counterclockwise convention adopted in this chapter.

It can be shown that each face in a sheet having multiple interior fold intersections undergoes a rigid deformation and no tearing occurs provided the local constraint (2.39) is satisfied for each interior fold intersection (see Problems 2.8, 2.9, and 2.10 for details).

Various trivial cases occur when only one, two, or three creased folds are incident to an interior fold intersection. These cases are the following:

1. *One incident fold* (Fig. 2.13a). If the jth interior fold intersection has a single incident creased fold, satisfaction of the kinematic constraint (2.39) requires:

$$\text{For } n_j = 1: \qquad \theta_{j1} = 0. \tag{2.40}$$

2. *Two incident folds* (Fig. 2.13b). If the jth interior fold intersection has two incident creased folds, satisfaction of the kinematic constraint (2.39) requires:

$$\text{For } n_j = 2: \qquad \begin{aligned} \theta_{j1} = \theta_{j2} = 0; &\quad \alpha_{j1} \neq \pi \\ \theta_{j1} = \theta_{j2}; &\quad \alpha_{j1} = \pi, \end{aligned} \tag{2.41}$$

i.e., the two folds are not collinear and the union of the faces adjacent to the jth interior fold intersection must remain flat *or* the two folds are collinear and a single combined fold results

3. *Three incident folds* (Fig. 2.13c). If the jth interior fold intersection has three incident creased folds, satisfaction of the kinematic constraint (2.39) requires:

$$\text{For } n_j = 3: \quad \begin{aligned} \theta_{j1} = \theta_{j2} = \theta_{j3} = 0; &\quad \alpha_{j1} \neq \pi, \alpha_{j2} \neq \pi, \alpha_{j3} \neq \pi \\ \theta_{j1} = \theta_{j2}, \theta_{j3} = 0; &\quad \alpha_{j1} = \pi \\ \theta_{j2} = \theta_{j3}, \theta_{j1} = 0; &\quad \alpha_{j2} = \pi \\ \theta_{j3} = \theta_{j1}, \theta_{j2} = 0; &\quad \alpha_{j3} = \pi, \end{aligned} \tag{2.42}$$

i.e., no pair of folds is collinear and the union of the faces adjacent to the jth interior fold intersection must remain flat *or* two of the folds are collinear and a single combined fold results while the third fold remains flat

Thus, for non-trivial folding motion, any interior fold intersection must have at least *four* incident creased folds.

Example 2.8 Fold intersection having a single incident fold (Fig. 2.13a).
Statement: Verify that (2.40) must hold for an interior fold intersection having a single incident fold.
Solution: The matrix $\mathbf{R}^j$ in (2.39) associated with an interior fold intersection having one incident creased fold is $\mathbf{R}^j = \mathbf{R}_1(\theta_{j1})\mathbf{R}_3(\alpha_{j1})$. The angle α_{j1} is equal to 2π for a single fold incident to an interior fold intersection, thus $\mathbf{R}_3(\alpha_{j1}) = \mathbf{R}_3(2\pi) = \mathbf{I}_3$. This requires that $\mathbf{R}_1(\theta_{j1}) = \mathbf{I}_3$, which for $\theta_{j1} \in [-\pi, \pi]$ holds true only when $\theta_{j1} = 0$. Cases for two or three folds, respectively, provided in (2.41) and (2.42) are verified in a similar manner (see Problem 2.6).

Example 2.9 Folding the corner of a cube [32, 33].
Statement: Figure 2.14 shows the fold pattern and fold angles used to fold the corner of a cube from a square sheet. The reference and final configurations are also shown. Verify that the kinematic constraint for origami with creased folds in (2.39) is satisfied for the fold angles shown.
Solution: The fold angles as enumerated in Fig. 2.14 are given as: $\theta_{11} = \pi/2$, $\theta_{12} = \pi/2$, $\theta_{13} = \pi/2$, $\theta_{14} = -\pi$, and $\theta_{15} = \pi$. The face corner angles can be determined from Fig. 2.14 and are as follows: $\alpha_{11} = \pi/2$, $\alpha_{12} = \pi/2$, $\alpha_{13} = \pi/4$, $\alpha_{14} = \pi/4$, and $\alpha_{15} = \pi/2$. The matrices $\mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk})$, $k = 1, \ldots, 5$, are then given as follows:

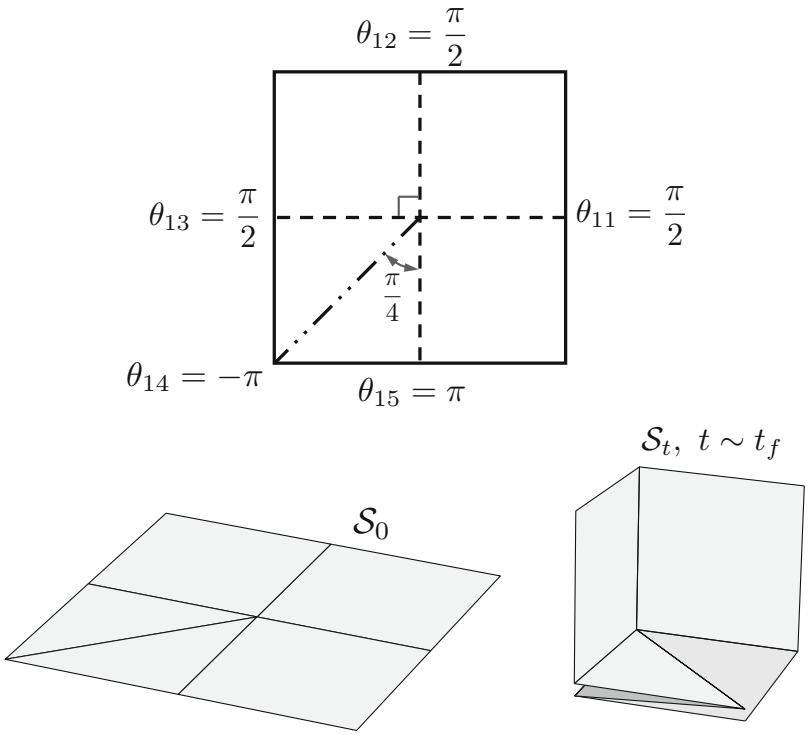


Fig. 2.14 Schematic for Example 2.9. Top: Fold pattern and fold angles for each fold at the final configuration [32, 33]. Bottom: Reference configuration $\mathcal{S}_0$ and configuration $\mathcal{S}_t$ (for visualization $t \sim t_f$)

$$\begin{aligned}\mathbf{R}_1(\theta_{11})\mathbf{R}_3(\alpha_{11}) = \mathbf{R}_1\left(\tfrac{\pi}{2}\right)\mathbf{R}_3\left(\tfrac{\pi}{2}\right) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix},\end{aligned} \tag{2.43}$$

$$\mathbf{R}_1\left(\theta_{12}\right)\mathbf{R}_3\left(\alpha_{12}\right) = \mathbf{R}_1\left(\tfrac{\pi}{2}\right)\mathbf{R}_3\left(\tfrac{\pi}{2}\right) = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}, \tag{2.44}$$

$$\mathbf{R}_1\left(\theta_{13}\right)\mathbf{R}_3\left(\alpha_{13}\right) = \mathbf{R}_1\left(\tfrac{\pi}{2}\right)\mathbf{R}_3\left(\tfrac{\pi}{4}\right) = \begin{bmatrix} 2^{-\frac{1}{2}} & -2^{-\frac{1}{2}} & 0 \\ 0 & 0 & -1 \\ 2^{-\frac{1}{2}} & 2^{-\frac{1}{2}} & 0 \end{bmatrix}, \tag{2.45}$$

$$\mathbf{R}_1(\theta_{14})\mathbf{R}_3(\alpha_{14}) = \mathbf{R}_1(-\pi)\mathbf{R}_3\left(\tfrac{\pi}{4}\right) = \begin{bmatrix} 2^{-\frac{1}{2}} & -2^{-\frac{1}{2}} & 0 \\ -2^{-\frac{1}{2}} & -2^{-\frac{1}{2}} & 0 \\ 0 & 0 & -1 \end{bmatrix}, \tag{2.46}$$

$$\mathbf{R}_1(\theta_{15})\mathbf{R}_3(\alpha_{15}) = \mathbf{R}_1(\pi)\mathbf{R}_3\left(\tfrac{\pi}{2}\right) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{2.47}$$

Finally, to verify if the fold angles shown in Fig. 2.14 satisfy the constraint (2.39), the previously calculated matrices are multiplied to check if their product is the identity matrix:

$$\begin{aligned} \prod_{k=1}^{5} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk}) = &\begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} 2^{-\frac{1}{2}} & -2^{-\frac{1}{2}} & 0 \\ 0 & 0 & -1 \\ 2^{-\frac{1}{2}} & 2^{-\frac{1}{2}} & 0 \end{bmatrix} \\ &\times \begin{bmatrix} 2^{-\frac{1}{2}} & -2^{-\frac{1}{2}} & 0 \\ -2^{-\frac{1}{2}} & -2^{-\frac{1}{2}} & 0 \\ 0 & 0 & -1 \end{bmatrix}\begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \\ = &\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathbf{I}_3. \end{aligned} \tag{2.48}$$

2.5 Folding Map Formulation

In this section, we provide a method to determine the mapping of the position vector of any point in the sheet from the reference configuration $\mathcal{S}_0$ to a current configuration $\mathcal{S}_t$ (termed *folding map*). A local folding map considering only the faces adjacent to an interior fold intersection was provided in (2.28). Such a formulation is extended here for the derivation of a folding map considering *all* the

faces in the sheet. The method to determine the folding map requires the following steps:

1. A single arbitrary face in the sheet is assumed fixed in its reference configuration
2. Paths $\breve{\boldsymbol{\gamma}}^j(\eta) : [0, 1] \to \mathcal{S}_0$, $j = 1, \ldots, N_{\mathcal{P}}$, connecting the fixed face to any other face with reference configuration $\mathcal{P}_0^j$, $j = 1, \ldots, N_{\mathcal{P}}$, are assumed (refer to Fig. 2.15 for examples). The paths $\breve{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_{\mathcal{P}}$, may not cross any fold intersection (i.e., they travel only through faces and creased folds of $\mathcal{S}_0$). Each path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses a number of $\breve{n}_j$ creased folds
3. The formulation of the transformation associated with folding each of the creased folds crossed by the paths $\breve{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_{\mathcal{P}}$ (termed *folding transformation* for simplicity) is determined
4. The folding map for any point in each face with reference configuration $\mathcal{P}_0^j$ is then obtained through the composition of folding transformations of the folds crossed by the corresponding path $\breve{\boldsymbol{\gamma}}^j(\eta)$

Steps 1 and 2 from the previous list can be readily performed. Sections 2.5.1 and 2.5.2 provide the information required to perform Step 3. Specifically, Sect. 2.5.1 addresses the determination of the parameters required to define the folding transformation while Sect. 2.5.2 addresses the formulation of such a transformation. The formulation of the folding map for Step 4 is also addressed in Sect. 2.5.2.

2.5.1 Parameters Required to Derive the Folding Map

To derive the folding transformation associated with each fold crossed by a path $\breve{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_{\mathcal{P}}$, certain geometric parameters and kinematic variables associated with such folds must be determined. Our focus in this section is to provide equations for all such parameters.

As shown in Fig. 2.16, a path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses the ith fold in the sheet *positively* if it crosses from the face adjacent to the fold in the direction $-\mathbf{e}_3 \times \hat{\mathbf{m}}^i$ to the face in the direction $\mathbf{e}_3 \times \hat{\mathbf{m}}^i$ (where $\hat{\mathbf{m}}^i$ is fold vector along the length of the ith fold line, see (2.6)). A path $\breve{\boldsymbol{\gamma}}^l(\eta)$ crosses a fold *negatively* if it crosses such a fold in the opposite direction as also illustrated in Fig. 2.16.

The *face connectivity matrix* $\mathbf{C}^{\mathcal{P}} \in \mathbb{R}^{N_{\mathcal{P}} \times \max(\breve{n}_j)}$ with components $C^{\mathcal{P}}_{jk}$ is used for the identification and ordering of the folds crossed by each path $\breve{\boldsymbol{\gamma}}^j(\eta)$:

$$
\begin{aligned}
C^{\mathcal{P}}_{jk} &= \text{Index of the } k\text{th fold crossed by the path } \breve{\boldsymbol{\gamma}}^j(\eta) \\
&\quad \text{(multiplied by } -1 \text{ if the fold is } \textit{negatively} \text{ crossed by } \breve{\boldsymbol{\gamma}}^j(\eta)), \\
j &= 1, \ldots, N_{\mathcal{P}}, \quad k = 1, \ldots, \breve{n}_j.
\end{aligned}
\tag{2.49}
$$

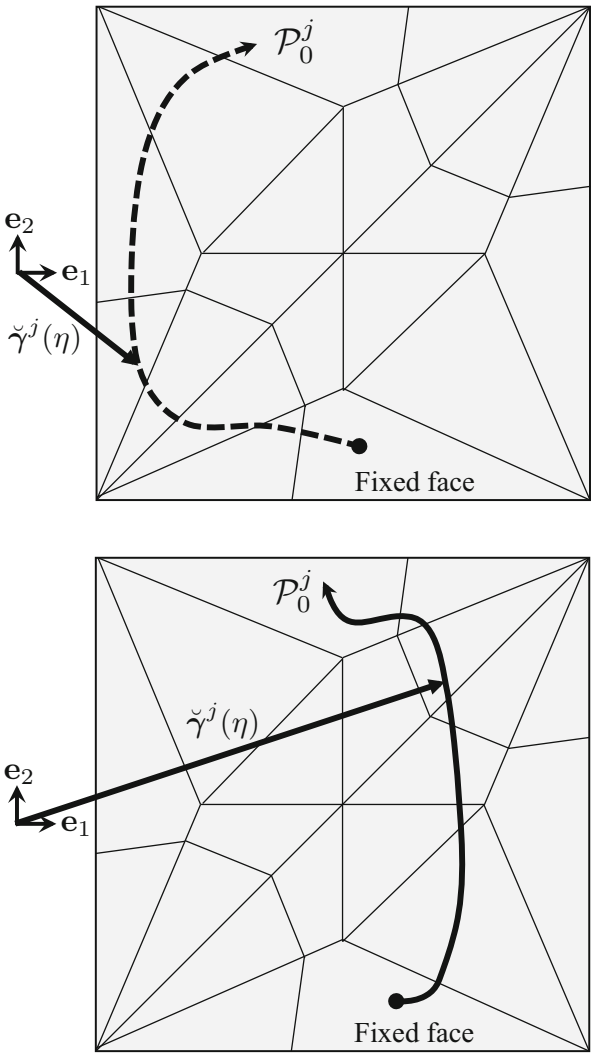


Fig. 2.15 Two equivalent paths $\breve{\boldsymbol{\gamma}}^j(\eta)$ connecting the fixed face to $\mathcal{P}_0^j$

Let $\breve{\mathbf{m}}^{jk} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the vector along the length of the kth fold line crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ (see Fig. 2.17b). This vector is defined such that the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses it positively (i.e., from the face in the direction $-\mathbf{e}_3 \times \breve{\mathbf{m}}^{jk}$ to the face in the direction $\mathbf{e}_3 \times \breve{\mathbf{m}}^{jk}$). The mapping from the fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_\mathcal{F}}$ to the vectors $\breve{\mathbf{m}}^{j1}, \ldots, \breve{\mathbf{m}}^{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$, is given as follows:

$$\breve{\mathbf{m}}^{jk} = \begin{cases} \hat{\mathbf{m}}^{C^\mathcal{P}_{jk}}; & C^\mathcal{P}_{jk} > 0 \\ -\hat{\mathbf{m}}^{|C^\mathcal{P}_{jk}|}; & C^\mathcal{P}_{jk} < 0, \end{cases} \qquad j = 1, \ldots, N_\mathcal{P},\ k = 1, \ldots, \breve{n}_j. \tag{2.50}$$

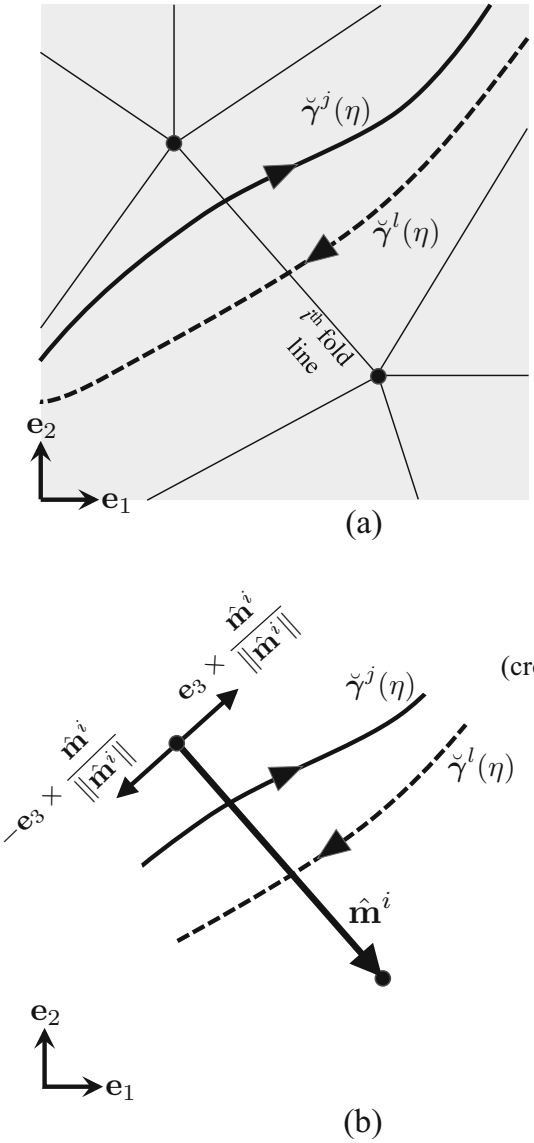


Fig. 2.16 (**a**) Two paths crossing the ith fold line in $\mathcal{S}_0$. (**b**) Fold vector $\hat{\mathbf{m}}^i$ along the length of the ith fold line in $\mathcal{S}_0$. The path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses the fold *positively* (i.e., from the side in the direction $-\mathbf{e}_3 \times \hat{\mathbf{m}}^i$ to the side in the direction $\mathbf{e}_3 \times \hat{\mathbf{m}}^i$) and the path $\breve{\boldsymbol{\gamma}}^l(\eta)$ crosses the fold *negatively*

Note that $\breve{\mathbf{m}}^{jk}$ has the opposite direction of its associated fold vector $\hat{\mathbf{m}}^{|C^{\mathcal{P}}_{jk}|}$ if $C^{\mathcal{P}}_{jk} < 0$. This is applied such that the vectors $\breve{\mathbf{m}}^{jk}$ are always crossed positively by $\breve{\boldsymbol{\gamma}}^j(\eta)$.

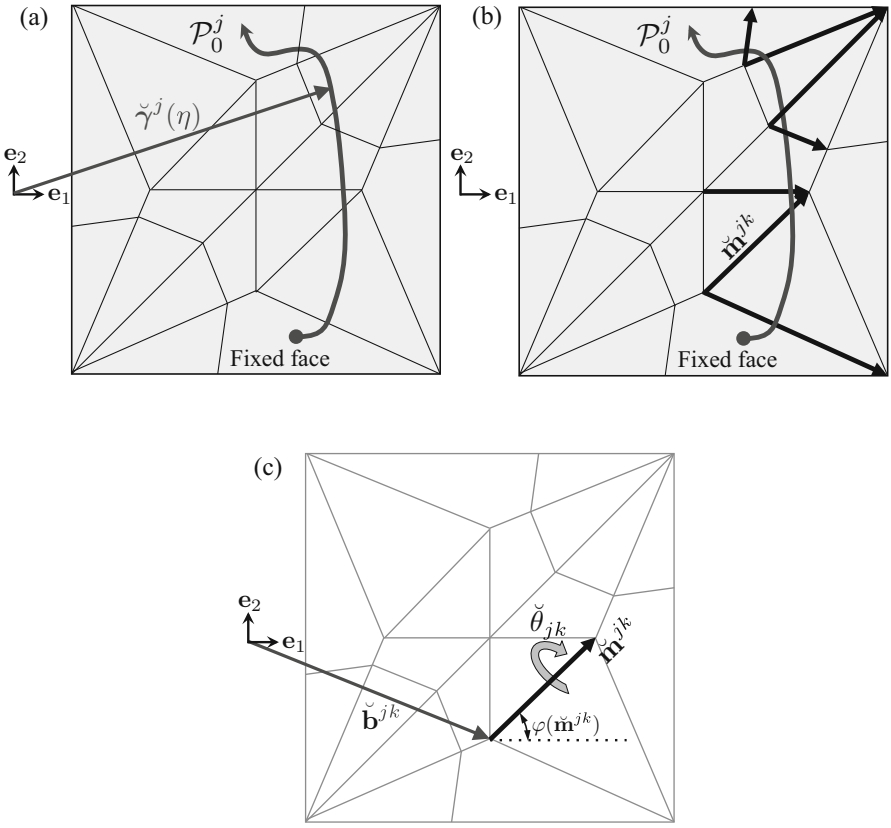


Fig. 2.17 An origami sheet illustrating the parameters that define the folding transformation: (**a**) Path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crossing creased folds between the fixed face and $\mathcal{P}_0^j$; (**b**) Vector $\breve{\mathbf{m}}^{jk}$ along the kth fold line crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$; (**c**) *Folding transformation* consisting of a rotation of $\breve{\theta}_{jk}$ about an axis crossing a point with position vector $\breve{\mathbf{b}}^{jk}$ and aligned to $\breve{\mathbf{m}}^{jk}$

Let $\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$ be the fold angles associated with the ordered creased folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$. The mapping from all the fold angles in the sheet ($\hat{\theta}_1, \ldots, \hat{\theta}_{N_\mathcal{F}}$, see Sect. 2.2) to the fold angles of only those crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$) is given as follows:

$$\breve{\theta}_{jk} = \hat{\theta}_{\left|C_{jk}^{\mathcal{P}}\right|} \qquad j = 1, \ldots, N_\mathcal{P}, \quad k = 1, \ldots, \breve{n}_j. \tag{2.51}$$

Thus far, the vectors $\breve{\mathbf{m}}^{j1}, \ldots, \breve{\mathbf{m}}^{j\breve{n}_j}$ that provide the orientation of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ in $\mathcal{S}_0$ and their associated fold angles $\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$ have been determined. The final parameters that are required in Sect. 2.5.2 to formulate the folding transformation are the position vectors of points located at the folds. The

Table 2.3 Input parameters required to define the folding map of an origami sheet with creased folds

Parameter	Definition
Fold angles $\hat{\theta}_1, \ldots, \hat{\theta}_{N_\mathcal{F}}$	
Face connectivity matrix $\mathbf{C}^\mathcal{P}$	(2.49)

These parameters are required in addition to those associated with the fold pattern listed in Tables 2.1 and 2.2

Table 2.4 Calculated parameters required to define the folding map of an origami sheet with creased folds

Parameter	Equation
Vectors along the fold lines crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\mathbf{m}}^{j1}, \ldots, \breve{\mathbf{m}}^{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$)	(2.50)
Fold angles of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$)	(2.51)
Position vectors of points on the fold lines crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\mathbf{b}}^{j1}, \ldots, \breve{\mathbf{b}}^{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$)	(2.52)

position vectors of such points are denoted $\breve{\mathbf{b}}^{j1}, \ldots, \breve{\mathbf{b}}^{j\breve{n}_j} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ (see Fig. 2.17c). The start points of the fold lines in $\mathcal{S}_0$ defined in (2.5) provide an obvious choice for the points along each fold crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$. The position vectors of such start points ($\hat{\mathbf{v}}^{11}, \ldots, \hat{\mathbf{v}}^{N_\mathcal{F}1}$) are then used to define the vectors $\breve{\mathbf{b}}^{j1}, \ldots, \breve{\mathbf{b}}^{j\breve{n}_j}$ as follows:

$$\breve{\mathbf{b}}^{jk} = \hat{\mathbf{v}}^{\left|C^\mathcal{P}_{jk}\right| 1} \qquad j = 1, \ldots, N_\mathcal{P}, \quad k = 1, \ldots, \breve{n}_j. \tag{2.52}$$

We summarize the input and calculated parameters required to define the folding map of an origami sheet with creased folds in Tables 2.3 and 2.4.

2.5.2 *Folding Map Formulation*

Our goal in this section is to derive the folding transformation associated with each fold crossed by the paths $\breve{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_\mathcal{P}$. Such a folding transformation is illustrated in Fig. 2.17c and corresponds to a rotation of $\breve{\theta}_{jk}$ about an axis crossing a point with position vector $\breve{\mathbf{b}}^{jk}$ and aligned to vector $\breve{\mathbf{m}}^{jk}$. The *folding map* will then be obtained through the composition of the folding transformations associated with the folds crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$. The folding transformation can be formulated as a composition of rotation and translation transformations in the following order:

1. The axis of rotation is translated such that it crosses the origin (via a translation by $-\breve{\mathbf{b}}^{jk}$)
2. The axis of rotation is then aligned with $\mathbf{e}_1$ (via a rotation of $-\varphi(\breve{\mathbf{m}}^{jk})$ about $\mathbf{e}_3$)

3. Once the axis of rotation is aligned with $\mathbf{e}_1$, a rotation of $\breve{\theta}_{jk}$ about $\mathbf{e}_1$ is applied
4. The axis of rotation is then aligned to its initial orientation (via a rotation of $\varphi(\breve{\mathbf{m}}^{jk})$ about $\mathbf{e}_3$)
5. The axis of rotation is finally translated to its initial position (via a translation by $\breve{\mathbf{b}}^{jk}$)

To formulate the previous composition of transformations in matrix form for notational convenience and computational implementation, *homogeneous coordinates* are introduced. Such coordinates allow for the formulation of both translation and rotation transformations in matrix form [32, 33, 40].

Let $\mathbf{T}(\mathbf{b}) \in \mathbb{R}^{4\times 4}$ be the matrix in homogeneous coordinates representing the transformation associated with a translation by vector $\mathbf{b} \in \mathbb{R}^3$ with components b_1, b_2, b_3:

$$\mathbf{T}(\mathbf{b}) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & b_1 \\ 0 & 1 & 0 & b_2 \\ 0 & 0 & 1 & b_3 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \begin{bmatrix} \mathbf{I}_3 & \mathbf{b} \\ \mathbf{0}_3^\top & 1 \end{bmatrix}, \tag{2.53}$$

where $\mathbf{0}_n \in \mathbb{R}^n$ is the zero vector. The translation by $\mathbf{b}$ of a position vector $\mathbf{y} \in \mathbb{R}^3$ with components y_1, y_2, y_3 is determined as follows:

$$\begin{aligned} \mathbf{T}(\mathbf{b}) \begin{bmatrix} \mathbf{y} \\ 1 \end{bmatrix} &= \begin{bmatrix} \mathbf{I}_3 & \mathbf{b} \\ \mathbf{0}_3^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 & b_1 \\ 0 & 1 & 0 & b_2 \\ 0 & 0 & 1 & b_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ 1 \end{bmatrix} = \begin{bmatrix} y_1 + b_1 \\ y_2 + b_2 \\ y_3 + b_3 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{y} + \mathbf{b} \\ 1 \end{bmatrix}. \end{aligned} \tag{2.54}$$

Let $\mathbf{Q}_1(\phi) \in \mathbb{R}^{4\times 4}$ be the transformation matrix in homogeneous coordinates associated with a rotation by ϕ radians about an axis of rotation aligned to $\mathbf{e}_1$:

$$\mathbf{Q}_1(\phi) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) & 0 \\ 0 & \sin(\phi) & \cos(\phi) & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \begin{bmatrix} \mathbf{R}_1(\phi) & \mathbf{0}_3 \\ \mathbf{0}_3^\top & 1 \end{bmatrix}, \tag{2.55}$$

The rotation of a vector $\mathbf{y} \in \mathbb{R}^3$ by ϕ radians about an axis of rotation aligned to $\mathbf{e}_1$ is determined as follows:

$$\mathbf{Q}_1(\phi)\begin{bmatrix}\mathbf{y}\\1\end{bmatrix}=\begin{bmatrix}\mathbf{R}_1(\phi) & \mathbf{0}_3\\ \mathbf{0}_3^\top & 1\end{bmatrix}\begin{bmatrix}\mathbf{y}\\1\end{bmatrix}=\begin{bmatrix}\mathbf{R}_1(\phi)\,\mathbf{y}\\1\end{bmatrix}. \tag{2.56}$$

Also, let $\mathbf{Q}_3(\phi) \in \mathbb{R}^{4\times4}$ be the transformation matrix in homogeneous coordinates associated with a rotation by ϕ radians about an axis of rotation aligned to $\mathbf{e}_3$:

$$\mathbf{Q}_3(\phi)=\left[\begin{array}{rrr|r}\cos(\phi) & -\sin(\phi) & 0 & 0\\ \sin(\phi) & \cos(\phi) & 0 & 0\\ 0 & 0 & 1 & 0\\ \hline 0 & 0 & 0 & 1\end{array}\right]=\begin{bmatrix}\mathbf{R}_3(\phi) & \mathbf{0}_3\\ \mathbf{0}_3^\top & 1\end{bmatrix}. \tag{2.57}$$

The rotation of a vector $\mathbf{y} \in \mathbb{R}^3$ by ϕ radians about an axis of rotation aligned to $\mathbf{e}_3$ is determined in the same manner as that shown in (2.56).

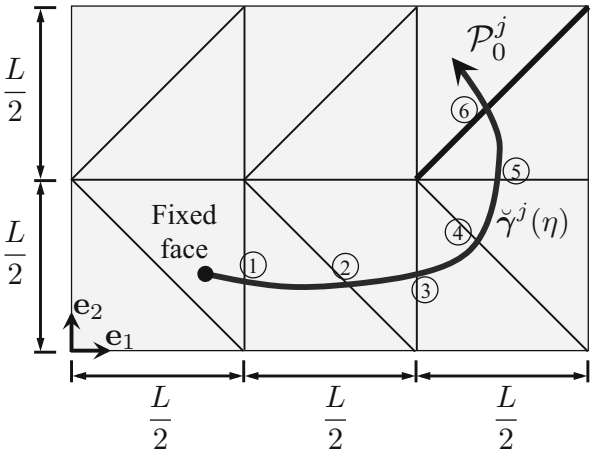
After providing a formulation for both rotation and translation transformations in matrix form, the folding transformation associated with each fold crossed by the paths $\breve{\boldsymbol{\gamma}}^j(\eta)$ can be expressed as a matrix denoted $\breve{\mathbf{H}}^{jk} \in \mathbb{R}^{4\times4}$, $j = 1, \ldots, N_{\mathcal{P}}$, $k = 1, \ldots, \breve{n}_j$ (i.e., the *folding transformation matrix*). Recalling from the beginning of this section, such a transformation is a composition of the ordered transformations corresponding to: (1) a translation by $-\breve{\mathbf{b}}^{jk}$, (2) a rotation by $-\varphi(\breve{\mathbf{m}}^{jk})$ about $\mathbf{e}_3$, (3) a rotation by $\breve{\theta}_{jk}$ about $\mathbf{e}_1$, (4) a rotation by $\varphi(\breve{\mathbf{m}}^{jk})$ about $\mathbf{e}_3$, and (5) a translation by $\breve{\mathbf{b}}^{jk}$. The composition of such transformations corresponding to the *folding transformation matrix* is then compactly expressed as follows:

$$\boxed{\breve{\mathbf{H}}^{jk}=\mathbf{T}\big(\breve{\mathbf{b}}^{jk}\big)\,\mathbf{Q}_3\big(\varphi\big(\breve{\mathbf{m}}^{jk}\big)\big)\,\mathbf{Q}_1\big(\breve{\theta}_{jk}\big)\,\mathbf{Q}_3^{-1}\big(\varphi\big(\breve{\mathbf{m}}^{jk}\big)\big)\,\mathbf{T}^{-1}\big(\breve{\mathbf{b}}^{jk}\big).} \tag{2.58}$$

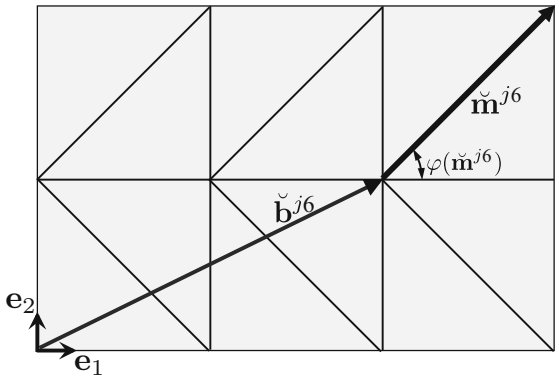
Example 2.10 Determining a folding transformation matrix.
Statement: Determine the folding transformation matrix associated with the sixth fold crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ shown in Fig. 2.18a.
Solution: The parameters required to describe the folding transformation matrix are illustrated in Fig. 2.18b. We note that $\breve{\mathbf{m}}^{j6} = [\ \frac{L}{2}\ \ \frac{L}{2}\ \ 0]^\top$ (therefore $\varphi(\breve{\mathbf{m}}^{j6}) = \frac{\pi}{4}$) and also that $\breve{\mathbf{b}}^{j6} = [\ L\ \ \frac{L}{2}\ \ 0]^\top$.

To determine the folding transformation matrix $\breve{\mathbf{H}}^{j6}$ via the formulation provided in (2.58), we require the following matrices:

$$\mathbf{T}\big(\breve{\mathbf{b}}^{j6}\big)=\mathbf{T}\big([\ L\ \ \tfrac{L}{2}\ \ 0]^\top\big)=\begin{bmatrix}1 & 0 & 0 & L\\ 0 & 1 & 0 & \frac{L}{2}\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}, \tag{2.59}$$



(a)



(b)

Fig. 2.18 (**a**) Path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crossing creased folds between the fixed face and $\mathcal{P}_0^j$. The numbering of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ is labeled. (**b**) Vector $\breve{\mathbf{m}}^{j6}$ along the sixth fold line crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$

$$\mathbf{Q}_1\big(\breve{\theta}^{j6}\big) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\big(\breve{\theta}^{j6}\big) & -\sin\big(\breve{\theta}^{j6}\big) & 0 \\ 0 & \sin\big(\breve{\theta}^{j6}\big) & \cos\big(\breve{\theta}^{j6}\big) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.60}$$

$$\mathbf{Q}_3\big(\varphi\big(\breve{\mathbf{m}}^{j6}\big)\big) = \mathbf{Q}_3\big(\tfrac{\pi}{4}\big) = \begin{bmatrix} \frac{2^{1/2}}{2} & -\frac{2^{1/2}}{2} & 0 & 0 \\ \frac{2^{1/2}}{2} & \frac{2^{1/2}}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.61}$$

We then use (2.59)–(2.61) to determine the explicit form for the folding transformation matrix $\breve{\mathbf{H}}^{j6}$:

$$\breve{\mathbf{H}}^{j6} = \mathbf{T}\big(\breve{\mathbf{b}}^{j6}\big)\,\mathbf{Q}_3\big(\varphi\big(\breve{\mathbf{m}}^{j6}\big)\big)\,\mathbf{Q}_1\big(\breve{\theta}_{j6}\big)\,\mathbf{Q}_3^{-1}\big(\varphi\big(\breve{\mathbf{m}}^{j6}\big)\big)\,\mathbf{T}^{-1}\big(\breve{\mathbf{b}}^{j6}\big)$$

$$= \begin{bmatrix} \frac{1}{2}+\frac{1}{2}\cos\big(\breve{\theta}^{j6}\big) & \frac{1}{2}-\frac{1}{2}\cos\big(\breve{\theta}^{j6}\big) & \frac{2^{1/2}}{2}\sin\big(\breve{\theta}^{j6}\big) & -\frac{L}{4}\big(\cos\big(\breve{\theta}^{j6}\big)-1\big) \\ \frac{1}{2}-\frac{1}{2}\cos\big(\breve{\theta}^{j6}\big) & \frac{1}{2}+\frac{1}{2}\cos\big(\breve{\theta}^{j6}\big) & -\frac{2^{1/2}}{2}\sin\big(\breve{\theta}^{j6}\big) & \frac{L}{4}\big(\cos\big(\breve{\theta}^{j6}\big)-1\big) \\ -\frac{2^{1/2}}{2}\sin\big(\breve{\theta}^{j6}\big) & \frac{2^{1/2}}{2}\sin\big(\breve{\theta}^{j6}\big) & \cos\big(\breve{\theta}^{j6}\big) & \frac{2^{1/2}L}{4}\sin\big(\breve{\theta}^{j6}\big) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.62}$$

The *folding map* used to transform the position vector of any point in the sheet from the reference configuration $\mathcal{S}_0$ to a current configuration $\mathcal{S}_t$ is formulated here based on the fold transformation matrices $\breve{\mathbf{H}}^{jk}$ provided in (2.58).

Let $\mathbf{X} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the position vector of a point in a face with reference configuration $\mathcal{P}_0^j \subset \mathcal{S}_0$ within which the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ terminates and $\mathbf{x} \in \mathbb{R}^3$ be the position vector of such a point in $\mathcal{P}_t^j \subset \mathcal{S}_t$. The mapping $\mathbf{X} \mapsto \mathbf{x}$ is determined as the multiplication of the ordered matrices $\breve{\mathbf{H}}^{jk}$ of the $\breve{n}_j$ creased folds crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$:

$$\boxed{\begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = \left(\prod_{k=1}^{\breve{n}_j} \breve{\mathbf{H}}^{jk} \right) \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}.} \tag{2.63}$$

The reader is asked to show that the folding map provided in (2.63) is independent from the chosen path $\breve{\boldsymbol{\gamma}}^j(\eta)$ in Problem 2.10.

Example 2.11 Mapping between reference and current configurations for a simple strip with two folds.

Statement: Consider the origami sheet presented in Fig. 2.19. (a) Determine the folding map in the form of (2.63) for each face shown in the figure. (b) Determine the position vector $\mathbf{x}$ in a current configuration with fold angles $\hat{\theta}_1 = \frac{\pi}{2}$ and $\hat{\theta}_2 = -\frac{\pi}{2}$ of the point with reference position vector $\mathbf{X} = [L_1 + L_2 + L_3 \quad 0 \quad 0]^\top$.

Solution: (a) Face 1 is the fixed face. Therefore, the folding map for points in this face is trivially given as follows:

$$\mathbf{x} = \mathbf{X} \qquad \mathbf{X} \in \mathcal{P}_0^1. \tag{2.64}$$

A path $\breve{\boldsymbol{\gamma}}^2(\eta)$ starting from $\mathcal{P}_0^1$ crosses Fold 1 to reach $\mathcal{P}_0^2$. The vector $\breve{\mathbf{m}}^{21}$ along the length of this fold is given as $\breve{\mathbf{m}}^{21} = -L_v\mathbf{e}_2$. Thus, $\varphi(\breve{\mathbf{m}}^{21}) = \varphi(-L_v\mathbf{e}_2) = \frac{3\pi}{2}$.

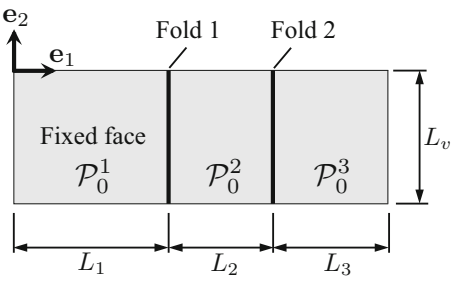


Fig. 2.19 Schematic for Example 2.11

The position vector $\breve{\mathbf{b}}^{21}$ of a point in Fold 1 can be given as $\breve{\mathbf{b}}^{21} = [\ L_1\ \ 0\ \ 0]^\top$. Therefore, the transformation matrix $\breve{\mathbf{H}}^{21}$ associated with Fold 1 can be expressed as a rotation by $\hat{\theta}_1$ about an axis aligned with $-L_v\mathbf{e}_2$ and crossing a point with position vector $[\ L_1\ \ 0\ \ 0]^\top$. The folding map for points in $\mathcal{P}_0^2$ is then given as follows:

$$\begin{bmatrix}\mathbf{x}\\ 1\end{bmatrix} = \breve{\mathbf{H}}^{21}\begin{bmatrix}\mathbf{X}\\ 1\end{bmatrix} \qquad \mathbf{X} \in \mathcal{P}_0^2, \tag{2.65}$$

where:

$$\begin{aligned}\breve{\mathbf{H}}^{21} &= \mathbf{T}\big([\ L_1\ \ 0\ \ 0]^\top\big)\,\mathbf{Q}_3\big(\tfrac{3\pi}{2}\big)\,\mathbf{Q}_1\big(\hat{\theta}_1\big)\,\mathbf{Q}_3^{-1}\big(\tfrac{3\pi}{2}\big)\,\mathbf{T}^{-1}\big([\ L_1\ \ 0\ \ 0]^\top\big)\\ &= \begin{bmatrix} \cos\big(\hat{\theta}_1\big) & 0 & -\sin\big(\hat{\theta}_1\big) & L_1\big(1-\cos\big(\hat{\theta}_1\big)\big)\\ 0 & 1 & 0 & 0\\ \sin\big(\hat{\theta}_1\big) & 0 & \cos\big(\hat{\theta}_1\big) & -L_1\sin\big(\hat{\theta}_1\big)\\ 0 & 0 & 0 & 1\end{bmatrix}.\end{aligned} \tag{2.66}$$

A path $\breve{\boldsymbol{\gamma}}^3(\eta)$ starting from $\mathcal{P}_0^1$ crosses Fold 1 and Fold 2 to reach $\mathcal{P}_0^3$. The vectors $\breve{\mathbf{m}}^{31}$ and $\breve{\mathbf{m}}^{32}$ along the length of these folds are given as $\breve{\mathbf{m}}^{31} = \breve{\mathbf{m}}^{32} = -L_v\mathbf{e}_2$. Therefore, $\varphi(\breve{\mathbf{m}}^{31}) = \varphi(\breve{\mathbf{m}}^{32}) = \varphi(-L_v\mathbf{e}_2) = \frac{3\pi}{2}$. The position vector $\breve{\mathbf{b}}^{31}$ of a point in Fold 1 can be given as $\breve{\mathbf{b}}^{31} = [\ L_1\ \ 0\ \ 0]^\top$ and the position vector $\breve{\mathbf{b}}^{32}$ of a point in Fold 2 can be given as $\breve{\mathbf{b}}^{32} = [L_1 + L_2\ \ 0\ \ 0]^\top$. Thus, the transformation matrix $\breve{\mathbf{H}}^{31}$ associated with Fold 1 can be expressed as a rotation by $\hat{\theta}_1$ about an axis aligned with $-L_v\mathbf{e}_2$ and crossing a point with position vector $[\ L_1\ \ 0\ \ 0]^\top$. Also, the transformation matrix $\breve{\mathbf{H}}^{32}$ associated with Fold 2 can be expressed as a rotation by $\hat{\theta}_2$ about an axis aligned with $-L_v\mathbf{e}_2$ and crossing a point with position vector $[L_1 + L_2\ \ 0\ \ 0]^\top$. The folding map for points in $\mathcal{P}_0^3$ is then given as follows (cf. (2.63)):

$$\begin{bmatrix}\mathbf{x}\\ 1\end{bmatrix} = \breve{\mathbf{H}}^{31}\,\breve{\mathbf{H}}^{32}\begin{bmatrix}\mathbf{X}\\ 1\end{bmatrix} \qquad \mathbf{X} \in \mathcal{P}_0^3, \tag{2.67}$$

where:

$$\begin{aligned}\breve{\mathbf{H}}^{31} &= \mathbf{T}\left([\,L_1\;\;0\;\;0]^\top\right)\mathbf{Q}_3\left(\tfrac{3\pi}{2}\right)\mathbf{Q}_1\left(\hat{\theta}_1\right)\mathbf{Q}_3^{-1}\left(\tfrac{3\pi}{2}\right)\mathbf{T}^{-1}\left([\,L_1\;\;0\;\;0]^\top\right)\\ &= \begin{bmatrix} \cos\left(\hat{\theta}_1\right) & 0 & -\sin\left(\hat{\theta}_1\right) & L_1\left(1-\cos\left(\hat{\theta}_1\right)\right)\\ 0 & 1 & 0 & 0\\ \sin\left(\hat{\theta}_1\right) & 0 & \cos\left(\hat{\theta}_1\right) & -L_1\sin\left(\hat{\theta}_1\right)\\ 0 & 0 & 0 & 1\end{bmatrix},\end{aligned} \tag{2.68}$$

$$\begin{aligned}\breve{\mathbf{H}}^{32} &= \mathbf{T}\left([L_1+L_2\;\;0\;\;0]^\top\right)\mathbf{Q}_3\left(\tfrac{3\pi}{2}\right)\mathbf{Q}_1\left(\hat{\theta}_2\right)\\ &\quad\times\mathbf{Q}_3^{-1}\left(\tfrac{3\pi}{2}\right)\mathbf{T}^{-1}\left([L_1+L_2\;\;0\;\;0]^\top\right)\\ &= \begin{bmatrix} \cos\left(\hat{\theta}_2\right) & 0 & -\sin\left(\hat{\theta}_2\right) & (L_1+L_2)\left(1-\cos\left(\hat{\theta}_2\right)\right)\\ 0 & 1 & 0 & 0\\ \sin\left(\hat{\theta}_2\right) & 0 & \cos\left(\hat{\theta}_2\right) & -(L_1+L_2)\sin\left(\hat{\theta}_2\right)\\ 0 & 0 & 0 & 1\end{bmatrix}.\end{aligned} \tag{2.69}$$

(b) The point with position vector $\mathbf{X} = [L_1+L_2+L_3\quad 0\quad 0]^\top$ is located in $\mathcal{P}_0^3$. Thus, the mapping provided in (2.67) is used to determine its position vector $\mathbf{x}$ in a current configuration with $\hat{\theta}_1 = \frac{\pi}{2}$ and $\hat{\theta}_2 = -\frac{\pi}{2}$ as follows:

$$\begin{aligned}\begin{bmatrix}\mathbf{x}\\1\end{bmatrix} &= \breve{\mathbf{H}}^{31}\big|_{\hat{\theta}_1=\frac{\pi}{2}}\,\breve{\mathbf{H}}^{32}\big|_{\hat{\theta}_2=-\frac{\pi}{2}}\begin{bmatrix}\mathbf{X}\\1\end{bmatrix}\\ &= \begin{bmatrix}0&0&-1&L_1\\0&1&0&0\\1&0&0&-L_1\\0&0&0&1\end{bmatrix}\begin{bmatrix}0&0&1&L_1+L_2\\0&1&0&0\\-1&0&0&L_1+L_2\\0&0&0&1\end{bmatrix}\begin{bmatrix}L_1+L_2+L_3\\0\\0\\1\end{bmatrix}\\ &= \begin{bmatrix}L_1+L_3\\0\\L_2\\1\end{bmatrix}.\end{aligned} \tag{2.70}$$

Therefore, $\mathbf{x} = [L_1+L_3\quad 0\quad L_2]^\top$. Figure 2.20 shows the reference configuration and current configuration with $\hat{\theta}_1 = \frac{\pi}{2}$ and $\hat{\theta}_2 = -\frac{\pi}{2}$ of the sheet examined in this example and the coordinates of the considered point in both configurations.

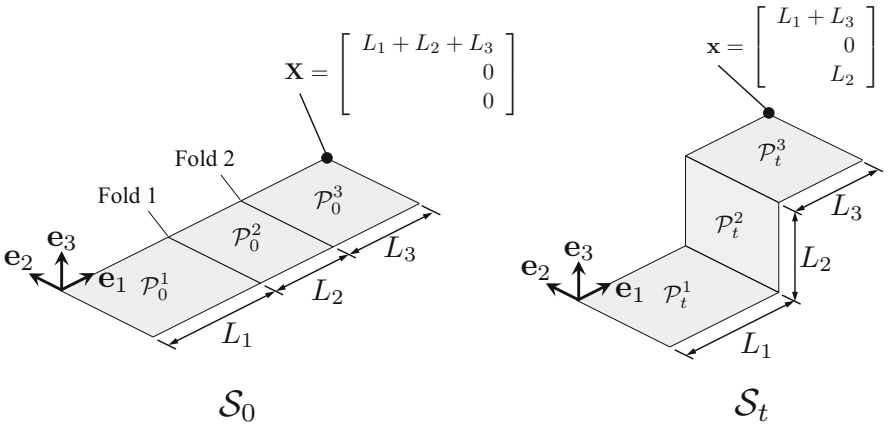


Fig. 2.20 Reference and current configurations of the origami sheet for Example 2.11

2.6 Computational Implementation of the Model

Having introduced the details of the kinematic model for origami structures with creased folds in Sects. 2.2–2.5, this chapter continues by describing a procedure to implement the model in a computational environment. The complete set of MATLAB scripts used for the execution of such a procedure is included in the Supplemental Materials of this chapter. We describe such MATLAB scripts in Appendix B.1.

Our goal is to simulate the folding motion of an origami sheet. We start by providing the fold pattern data listed in Table 2.1. Additionally, we also need to provide information regarding the desired *history of folding motion* for the sheet. For simulation in a numerical environment, the *continuous folding motion of the sheet is partitioned into increments*. Thus, the simulation of the folding motion is performed here by *incrementally* updating the values of the fold angles using input guess increments. As such, we need to provide the following data to simulate the folding motion of an origami sheet: (1) Fold pattern data (Table 2.1) and (2) Guess increments for the fold angles:

$$\Delta\hat{\theta}_1^l, \ \ldots, \ \Delta\hat{\theta}_{N_\mathcal{F}}^l \qquad l = 1, \ \ldots, \ N_{\text{inc}}, \tag{2.71}$$

where N_{inc} is the total number of guess increments.

For notational convenience, let us define $\Delta\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{N_\mathcal{F}}$ as the vector whose components correspond to the lth set of guess increments for the fold angles:

$$\Delta\hat{\boldsymbol{\theta}}^l = \begin{bmatrix} \Delta\hat{\theta}_1^l & \cdots & \Delta\hat{\theta}_{N_\mathcal{F}}^l \end{bmatrix}^\top \qquad l = 1, \ \ldots, \ N_{\text{inc}}. \tag{2.72}$$

Table 2.5 Conventional assignments for $\hat{\theta}_i^L$ and $\hat{\theta}_i^U$

Fold type	$\hat{\theta}_i^L$	$\hat{\theta}_i^U$
Valley	0	π
Mountain	$-\pi$	0
Valley or mountain	$-\pi$	π

Table 2.6 Constraints on the fold angles that must be satisfied at every increment (l) in the simulation of the folding motion

Kinematic constraints (2.39)	$\mathbf{R}^j(\hat{\boldsymbol{\theta}}^l) = \mathbf{I}_3 \quad j = 1, \ldots, N_{\mathcal{I}}$
Lower and upper bounds of the fold angles	$\hat{\boldsymbol{\theta}}^L \leq \hat{\boldsymbol{\theta}}^l \leq \hat{\boldsymbol{\theta}}^U$

Also, let us define $\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{N_{\mathcal{F}}}$ as the vector whose components correspond to the values of the fold angles at the lth increment:

$$\hat{\boldsymbol{\theta}}^l = \begin{bmatrix} \hat{\theta}_1^l & \cdots & \hat{\theta}_{N_{\mathcal{F}}}^l \end{bmatrix}^\top \qquad l = 1, \ldots, N_{\text{inc}}. \tag{2.73}$$

The fold angles $\hat{\boldsymbol{\theta}}^l$ must satisfy the kinematic constraints provided in (2.39) at every increment. Additionally, we also impose lower and upper bounds on the fold angles.[4] The lower bound and upper bound values for the ith fold angle are denoted $\hat{\theta}_i^L$ and $\hat{\theta}_i^U$, respectively. Conventional assignments for the lower and upper bounds of the fold angles are provided in Table 2.5.

We define $\hat{\boldsymbol{\theta}}^L, \hat{\boldsymbol{\theta}}^U \in \mathbb{R}^{N_{\mathcal{F}}}$ as the vectors whose components, respectively, correspond to $\hat{\theta}_1^L, \ldots, \hat{\theta}_{N_{\mathcal{F}}}^L$ and $\hat{\theta}_1^U, \ldots, \hat{\theta}_{N_{\mathcal{F}}}^U$ as follows:

$$\hat{\boldsymbol{\theta}}^L = \begin{bmatrix} \hat{\theta}_1^L & \cdots & \hat{\theta}_{N_{\mathcal{F}}}^L \end{bmatrix}^\top, \qquad \hat{\boldsymbol{\theta}}^U = \begin{bmatrix} \hat{\theta}_1^U & \cdots & \hat{\theta}_{N_{\mathcal{F}}}^U \end{bmatrix}^\top. \tag{2.74}$$

The constraints to be satisfied at each increment are then summarized in Table 2.6.

In general, the configurations determined by the simple addition of the guess increments for the fold angles (i.e., those obtained as $\hat{\boldsymbol{\theta}}^l = \sum_{m=1}^{l} \Delta\hat{\boldsymbol{\theta}}^m$) may not satisfy the constraints listed in Table 2.6. In view of this, we use an *iterative* procedure to apply any necessary corrections such that the resulting configurations approximate the addition of the guess increments and also satisfy the constraints of Table 2.6.

The components of the vector $\hat{\boldsymbol{\theta}}^{l(k)} \in \mathbb{R}^{N_{\mathcal{F}}}$ correspond to the values of the fold angles at the kth *iteration* of the lth increment. The vector $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)}) \in \mathbb{R}^{3N_{\mathcal{I}}+2N_{\mathcal{F}}}$ with components $\mathfrak{R}_j(\hat{\boldsymbol{\theta}}^{l(k)})$ is the *residual vector* from constraints of Table 2.6 ($3N_{\mathcal{I}}$

[4] It is remarked that the value of each fold angle must be contained in the interval $[-\pi, \pi]$ to prevent self-intersection of the pairs of faces connected by each fold.

from the kinematic constraints (2.39) and $2N_{\mathcal{F}}$ from the upper and lower bounds of each fold angle).

The matrix-type constraint (2.39) provides the following three scalar constraints that are included in the components of the residual vector [34][5]:

$$\begin{aligned}
\mathfrak{R}_{3j-2}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right) &= \frac{1}{2}\lambda_R\left(R_{23}^{j}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right)\right)^2,\\
\mathfrak{R}_{3j-1}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right) &= \frac{1}{2}\lambda_R\left(R_{31}^{j}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right)\right)^2,\\
\mathfrak{R}_{3j}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right) &= \frac{1}{2}\lambda_R\left(R_{12}^{j}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right)\right)^2,
\end{aligned} \tag{2.75}$$

where $j = 1, \ldots, N_{\mathcal{I}}$ and λ_R is the scalar weight for residuals from kinematic constraints (2.39).

The additional components of $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})$ required to enforce the lower bounds of the fold angles consist of penalties defined as follows:

$$\mathfrak{R}_{3N_{\mathcal{I}}+2i-1}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right) = \frac{1}{2}\lambda_B \max\left(0,\ -\hat{\theta}_i^{l(k)} + \hat{\theta}_i^{L}\right)^2, \tag{2.76}$$

where $i = 1, \ldots, N_{\mathcal{F}}$ and λ_B is the scalar weight for fold angle bound constraints. Similar penalties are used to enforce the upper bounds of the fold angles and are defined as follows:

$$\mathfrak{R}_{3N_{\mathcal{I}}+2i}\left(\hat{\boldsymbol{\theta}}^{l(k)}\right) = \frac{1}{2}\lambda_B \max\left(0,\ \hat{\theta}_i^{l(k)} - \hat{\theta}_i^{U}\right)^2, \tag{2.77}$$

where $i = 1, \ldots, N_{\mathcal{F}}$.

For each increment, we seek to iteratively correct the fold angles such that the magnitude of the residual vector $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})$ is minimized. At the start of each increment, we first calculate the projection of the guess increments for the fold angles $\Delta\hat{\boldsymbol{\theta}}^{l}$ onto the null space of the residual vector derivatives of the previous configuration (with fold angles $\hat{\boldsymbol{\theta}}^{l-1}$). This is performed to utilize the information of the previous configuration and reduce the number of iterations in the subsequent correction procedure [34]. Refer to Fig. 2.21a for an illustration of this concept. The resulting *projected* fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$ is calculated as follows (refer to (A.56)):

$$\Delta\hat{\boldsymbol{\theta}}^{l*} = \left(\mathbf{I}_{N_{\mathcal{F}}} - \left(\frac{\partial\mathfrak{R}\left(\hat{\boldsymbol{\theta}}^{l-1}\right)}{\partial\hat{\boldsymbol{\theta}}}\right)^{\dagger}\left(\frac{\partial\mathfrak{R}\left(\hat{\boldsymbol{\theta}}^{l-1}\right)}{\partial\hat{\boldsymbol{\theta}}}\right)\right)\Delta\hat{\boldsymbol{\theta}}^{l}. \tag{2.78}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse, see (A.37).

[5] Since $\mathbf{R}^j$ in (2.39) is an *orthogonal* matrix, only three of its nine components are independent.

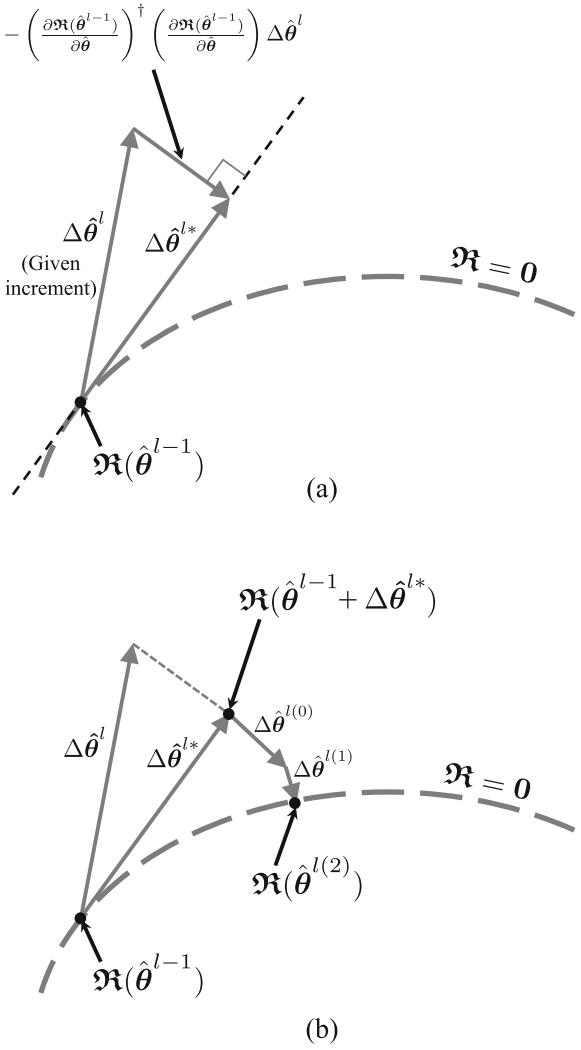


Fig. 2.21 (**a**) Illustration of the hypersurface corresponding to $\mathfrak{R}=\mathbf{0}_{3N_{\mathcal{I}}+2N_{\mathcal{F}}}$. The initial point $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l-1})$ is shown. The projected fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$ obtained via (2.78) is also indicated. (**b**) Subsequent corrections $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ (2.80). Convergence is achieved after two iterations in this illustration

Given the fold angles of the previously determined configuration $\hat{\boldsymbol{\theta}}^{l-1}$ and the projected fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$, a configuration satisfying the kinematic constraints (2.39) and the fold angles bounds can be found via an iterative correction procedure. Such a procedure begins at iteration (0) where the first guess of fold angles is simply the addition of the previous fold angles and the projected fold angle increment:

$$\hat{\boldsymbol{\theta}}^{l(0)}=\hat{\boldsymbol{\theta}}^{l-1}+\Delta\hat{\boldsymbol{\theta}}^{l*}. \tag{2.79}$$

If $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(0)})\|/(3N_{\mathcal{I}}+2N_{\mathcal{F}}) \geq \texttt{tol1}$ (where `tol1` is a numerical tolerance), the fold angles are corrected iteratively as follows (see Fig. 2.21b):

$$\hat{\boldsymbol{\theta}}^{l(k+1)} = \hat{\boldsymbol{\theta}}^{l(k)} + \Delta\hat{\boldsymbol{\theta}}^{l(k)}, \tag{2.80}$$

where $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ is the correction of fold angles at iteration (k). Following the generalized Newton's method [34], the first-order expansion of the residual vector $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})$ is used to determine the correction $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ required to minimize the components of the residual vector:

$$\mathfrak{R}\big(\hat{\boldsymbol{\theta}}^{l(k)} + \Delta\hat{\boldsymbol{\theta}}^{l(k)}\big) = \mathfrak{R}\big(\hat{\boldsymbol{\theta}}^{l(k)}\big) + \frac{\partial\mathfrak{R}\big(\hat{\boldsymbol{\theta}}^{l(k)}\big)}{\partial\hat{\boldsymbol{\theta}}}\Delta\hat{\boldsymbol{\theta}}^{l(k)} \simeq \mathbf{0}_{3N_{\mathcal{I}}+2N_{\mathcal{F}}}. \tag{2.81}$$

The following correction of fold angles $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ is obtained from the previous first-order expansion:

$$\Delta\hat{\boldsymbol{\theta}}^{l(k)} = -\left(\frac{\partial\mathfrak{R}\big(\hat{\boldsymbol{\theta}}^{l(k)}\big)}{\partial\hat{\boldsymbol{\theta}}}\right)^{\dagger}\mathfrak{R}\big(\hat{\boldsymbol{\theta}}^{l(k)}\big). \tag{2.82}$$

The iterative corrector procedure provided in (2.82) and (2.80) is repeated until $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k+1)})\|/(3N_{\mathcal{I}}+2N_{\mathcal{F}}) < \texttt{tol1}$ or $\|\Delta\hat{\boldsymbol{\theta}}^{l(k)}\|/N_{\mathcal{F}} < \texttt{tol2}$, where `tol2` is another numerical tolerance. The numerical procedure used to determine valid configurations at each increment in the folding motion for origami sheets with creased folds is summarized in Table 2.7.

Table 2.7 Numerical procedure used to determine valid configurations at each increment in the folding motion for origami sheets with creased folds (lth fold angle increment)

1:	Determine the projected fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$ from the given $\Delta\hat{\boldsymbol{\theta}}^{l}$ using (2.78)
2:	Calculate guess fold angles $\hat{\boldsymbol{\theta}}^{l(0)}$ using (2.79)

3:	`IF` $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(0)})\|/(3N_{\mathcal{I}}+2N_{\mathcal{F}}) <$ `tol1`
	`THEN` set $\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{l(0)}$ and `EXIT`
	`ELSE CONTINUE`

4:	Determine correction of fold angles $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ using (2.82)
5:	Determine $\hat{\boldsymbol{\theta}}^{l(k+1)}$ using (2.80)

6:	`IF` $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k+1)})\|/(3N_{\mathcal{I}}+2N_{\mathcal{F}}) <$ `tol1` `OR` $\|\Delta\hat{\boldsymbol{\theta}}^{l(k)}\|/N_{\mathcal{F}} <$ `tol2`
	`THEN` set $\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{l(k+1)}$ and `EXIT`
	`ELSE` set $k \leftarrow k+1$ and `GOTO` 4

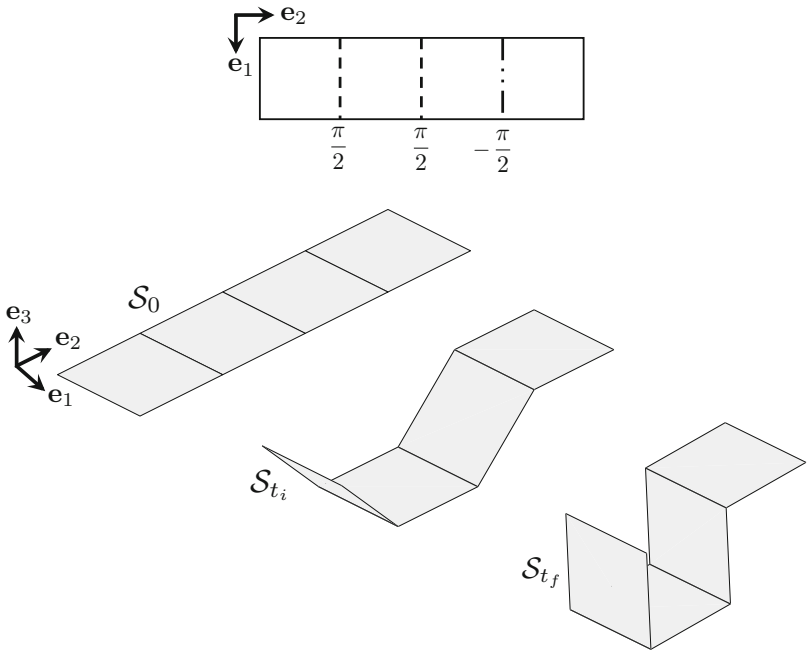


Fig. 2.22 Continuous folding motion of an origami sheet with three parallel folds ($0 < t_i < t_f$). The fold angles at the final configuration $\mathcal{S}_{t_f}$ are shown in the fold pattern schematic

2.7 Simulation Examples of the Kinematic Model

This section provides representative examples of the implemented model for kinematic simulation of origami with creased folds presented in Sects. 2.2–2.6. As mentioned in Sect. 2.6, the procedure for kinematic simulation of origami with creased folds has been implemented in MATLAB. The associated MATLAB scripts are provided in the Supplemental Materials of this chapter and described in Appendix B.1. In MATLAB, we visualize the faces of origami sheets as filled three-dimensional polygons using the command `fill3` [41].

Figures 2.22, 2.23, and 2.24 show examples of origami sheets that do not contain interior fold intersections. Therefore, the kinematic constraints provided in (2.39) do not need to be considered for these examples. A sheet with three parallel folds is shown in Fig. 2.22.

Figure 2.23 shows an "origami helix" whereby an initially planar sheet deforms in a "twisting" manner, although it is remarked that all the faces in the sheet are rigid and folding occurs only at the creases. All final configurations in this example resemble helixes (see Fig. 2.23). The sheets for the origami helix pattern are parameterized by the length L_1 along the direction aligned to $\mathbf{e}_1$, the length L_2 along the direction aligned to $\mathbf{e}_2$, and the number of triangle pairs along the direction aligned to $\mathbf{e}_1$ which is denoted as n.

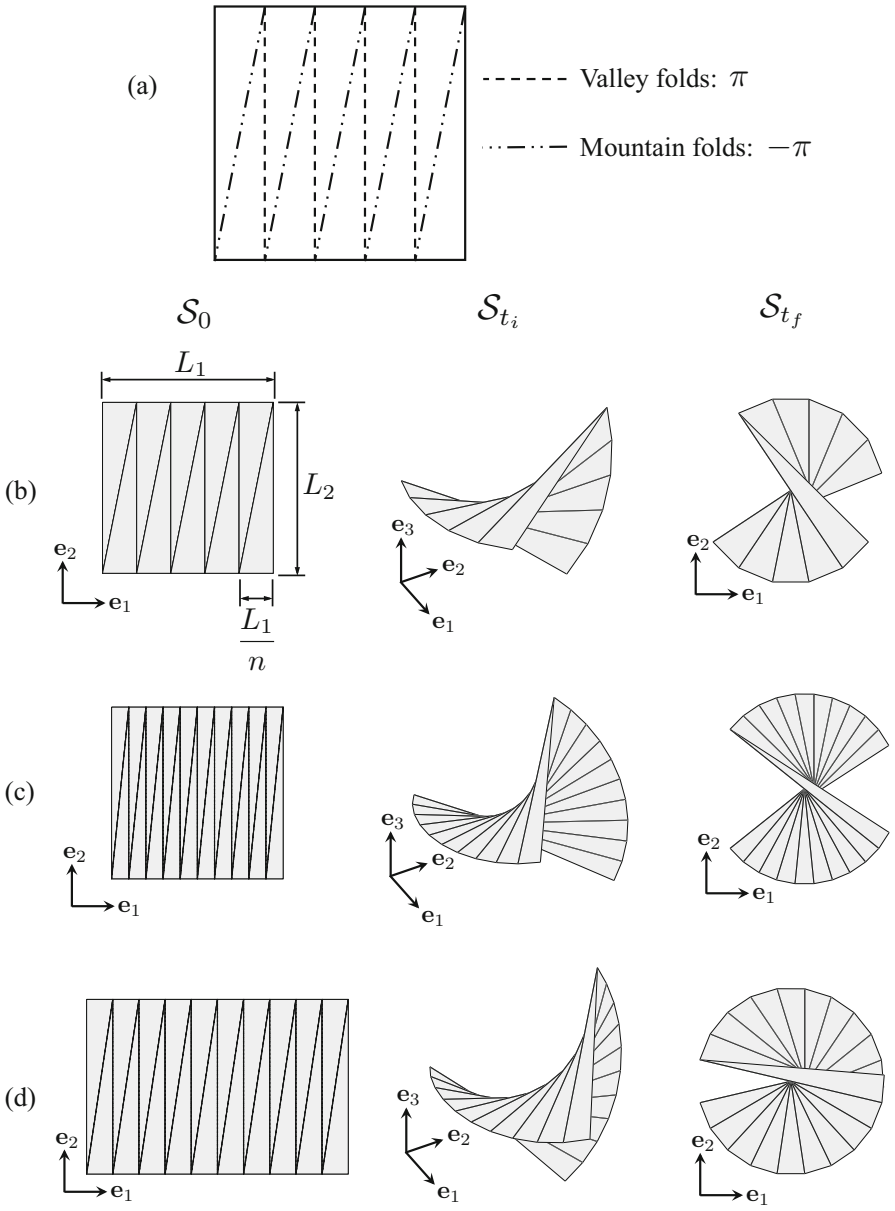


Fig. 2.23 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ for origami helixes. (**a**) $L_1 = L_2$, $n = 5$; (**b**) $L_1 = L_2$, $n = 10$; (**c**) $L_1 = \frac{3}{2}L_2$, $n = 10$

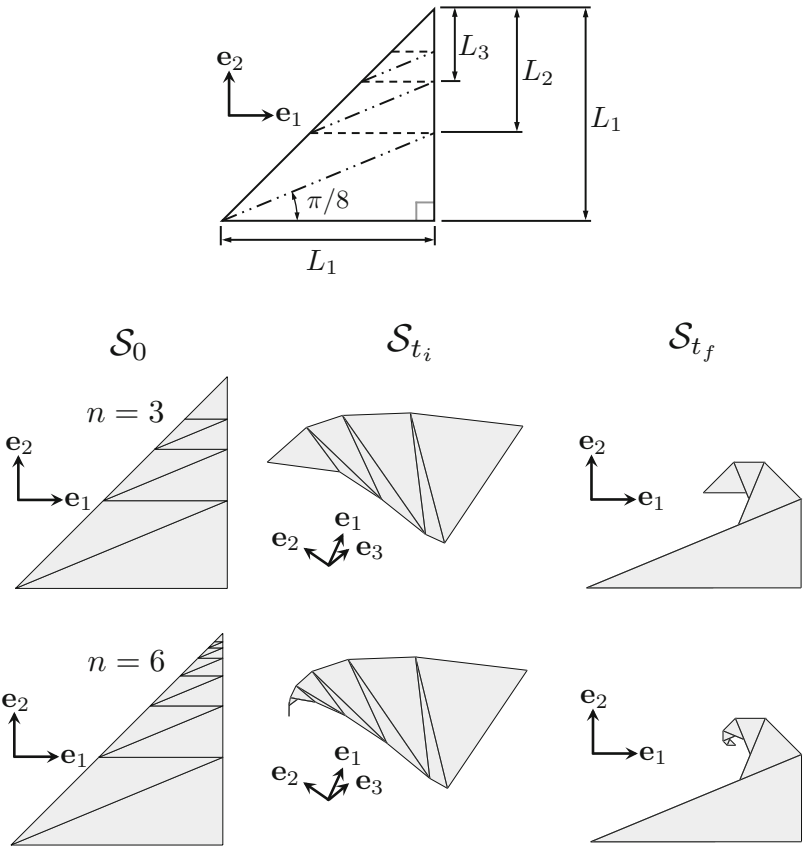


Fig. 2.24 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ for origami self-similar waves. The valley folds shown in the schematic reach a fold angle of π while the mountain folds reach a fold angle of $-\pi$

The fold pattern for the so-called "self-similar wave" [42] is shown in Fig. 2.24. In this fold pattern, the lengths $L_1, \ldots, L_n$ (refer to Fig. 2.24) are recursively determined as follows:

$$L_{i+1} = L_i \left(1 - \tan\left(\tfrac{\pi}{8}\right)\right). \tag{2.83}$$

Two examples of fold patterns for self-similar waves considering $n = 3$ and $n = 6$ are shown in Fig. 2.24. As its name implies, the final configuration for this origami fold pattern resembles a sea wave.

A sheet having eight creased folds meeting at one interior fold intersection is shown in Fig. 2.25. The folds are enumerated in counterclockwise order. Various

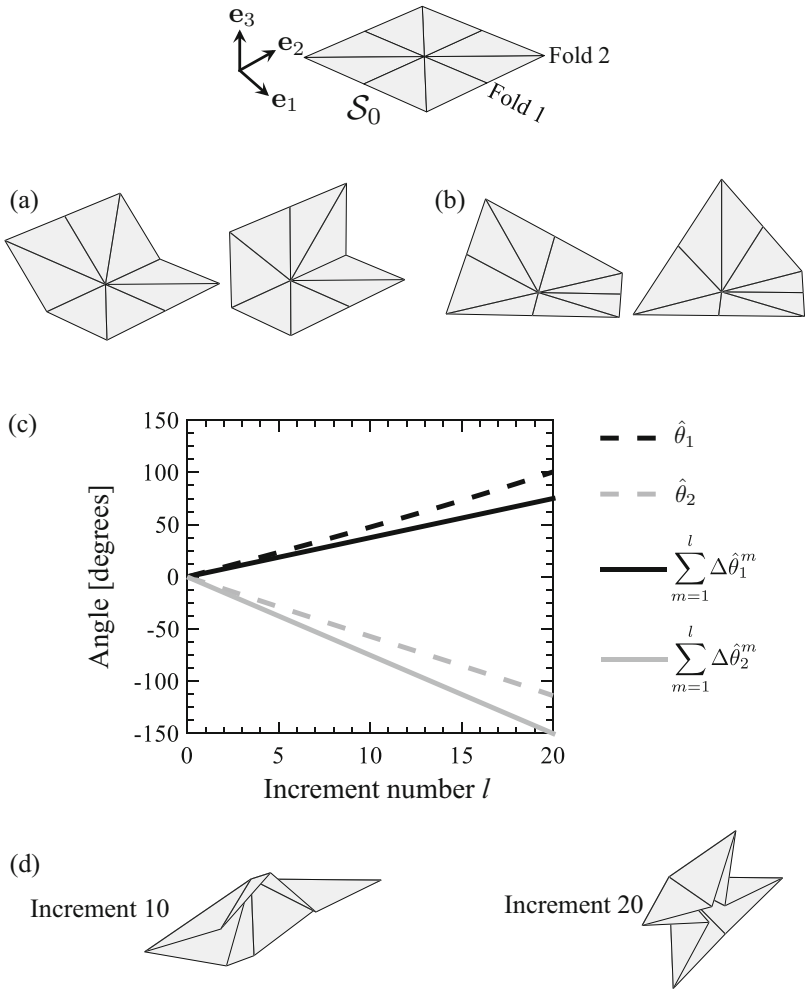


Fig. 2.25 (**a**) and (**b**) Configurations of a sheet having a single interior fold intersection obtained through the guess increments for the fold angles provided in (2.84) and (2.85), respectively. (**c**) Fold angles and sum of guess increments for the fold angles vs. increment number for (2.86). The fold angles obtained from the simulation procedure differ from the simple addition of the guess increments for the fold angles since corrections are required to satisfy the kinematic constraints of (2.39). (**d**) Configurations obtained through the guess increments for the fold angles provided in (2.86)

guess increments for the fold angles are considered ranging from simple to more complex. The folded configurations shown in Fig. 2.25a are obtained through the following guess increments for the fold angles:

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \frac{\pi}{40}\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^{\top} \qquad l = 1, \ldots, 20, \tag{2.84}$$

and the folded configurations shown in Fig. 2.25b are obtained through the following guess increments for the fold angles:

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \frac{\pi}{40}\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}^{\top} \qquad l = 1, \ldots, 20. \tag{2.85}$$

The guess increments for the fold angles in (2.84) and (2.85) represent simple examples where the fold angle correction procedure (refer to Table 2.7) converged prior to performing an initial correction iteration (i.e., $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(0)})\|/(3N_{\mathcal{I}} + 2N_{\mathcal{F}}) <$ `tol1` for $l = 1, \ldots, 20$). An example of a more complex folding motion resulting from guess increments for the fold angles that required iterative corrections is shown in Fig. 2.25c. For this example, the guess increments for the fold angles are as follows:

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \frac{5\pi}{240}\begin{bmatrix} 1 & -2 & -1 & -2 & 1 & -2 & 1 & -2 \end{bmatrix}^{\top} \qquad l = 1, \ldots, 20. \tag{2.86}$$

As shown in the plot in Fig. 2.25c, the fold angles obtained from the simulation procedure differ from the simple addition of the guess increments for the fold angles since corrections were required to satisfy the kinematic constraints of (2.39). It is observed in the configurations shown in Fig. 2.25d that all the faces comprising the sheet remain connected throughout the folding motion. Therefore, the simulation procedure presented in Sect. 2.6 successfully allows for the correction of the fold angles such that they satisfy the kinematic constraints presented in (2.39).

More complex examples of origami sheets having two interior fold intersections are shown in Fig. 2.26. Since the two interior fold intersections for these sheets share a common adjacent fold, their associated kinematic constraint equations are *coupled*. For all three sheets, the guess increments for the fold angles are the same and given by the following expression (refer to the numbering of the folds in Fig. 2.26):

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \frac{2\pi}{150}\begin{bmatrix} 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 \end{bmatrix}^{\top}$$

$$l = 1, \ldots, 50. \tag{2.87}$$

The sheets having fold patterns obtained by modifying the interior vertex coordinates and the boundary vertex coordinates of the baseline fold pattern of Fig. 2.26a are shown in Fig. 2.26b and c, respectively. It is observed that the three sheets undergo dissimilar folding motion as observed from their configurations

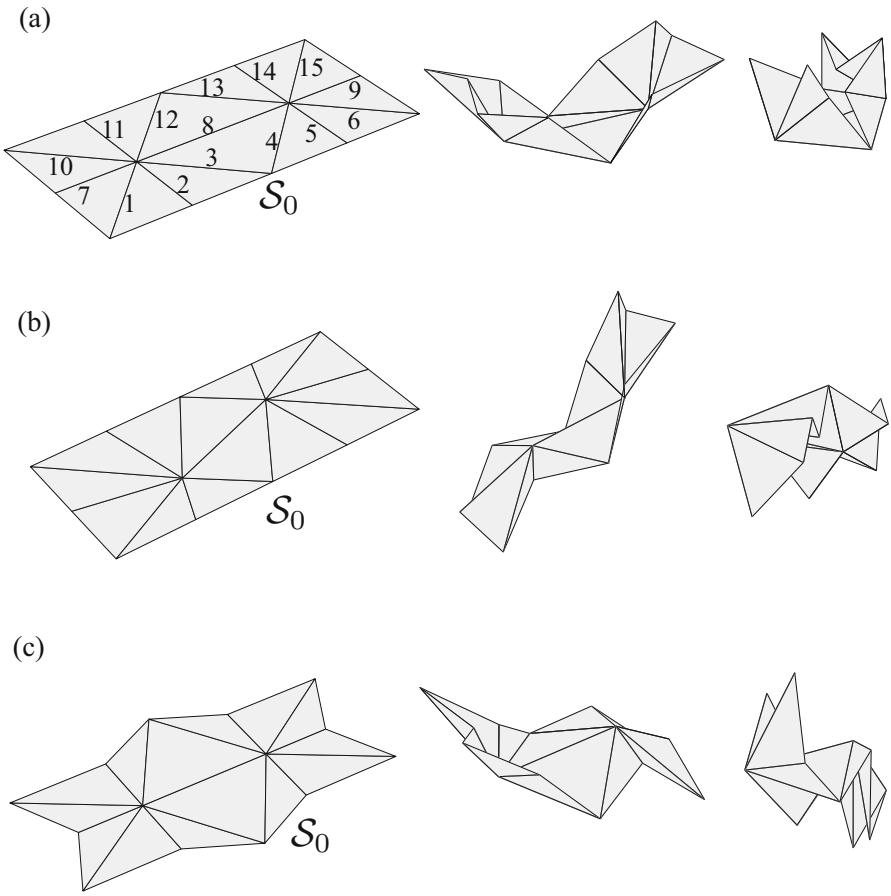


Fig. 2.26 Continuous folding motion of origami sheets having two interior fold intersections: (**a**) Baseline fold pattern; (**b**) Fold pattern obtained by modifying the interior vertex coordinates of the baseline fold pattern; (**c**) Fold pattern obtained by modifying the boundary vertex coordinates of the baseline fold pattern

shown due to the differences in their fold patterns. These examples show the versatility of the model and implementation procedure presented in this chapter to allow for simulation of sheets having arbitrary fold patterns and boundary shapes.

The final example shown in Fig. 2.27 considers a sheet having five interior fold intersections. It is observed that the presented model captures well the behavior of the origami sheet during its full range of motion (fold angles vary from 0 to $\pm\pi$ for various folds in the sheet).

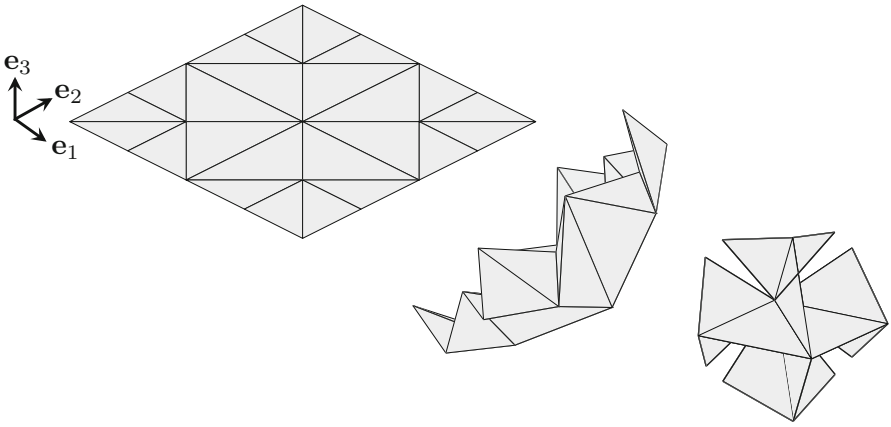


Fig. 2.27 Continuous folding motion of an origami square sheet with five interior fold intersections

Chapter Summary

A general model for the kinematic response of origami structures with creased folds was presented in this chapter. The fold pattern description (Sect. 2.3), developability and kinematic constraints (Sect. 2.4), and the mapping from reference to current configurations (Sect. 2.5) were presented. An implementation procedure for the model allowing for simulation of the folding motion of origami sheets was outlined in Sect. 2.6, and implementation examples generated using the code provided in the Supplemental Material of this chapter were provided in Sect. 2.7. The careful consideration of kinematic constraints ensures that only valid configurations are predicted. The presented model and its implementation readily allow for the simulation of a wide range of fold patterns and folding histories as demonstrated in the implementation examples provided in this chapter.

Problems

2.1 Consider the reference configuration $\mathcal{S}_0$ of the origami airplane shown in Fig. 2.28. (a) Provide the fold pattern data listed in Table 2.1 for this sheet. (b) Calculate the fold pattern parameters listed in Table 2.2.

2.2 Considering the fold pattern presented in Fig. 2.29, determine the mapping between reference and current configurations in the form of (2.28) for each face.

2.3 In Sect. 2.4.2, the kinematic constraints on the fold angles for the folds incident to an interior fold intersection were derived using a path $\boldsymbol{\gamma}^j(\eta)$ oriented counterclockwise. Derive such constraints using the process presented Sect. 2.4 but

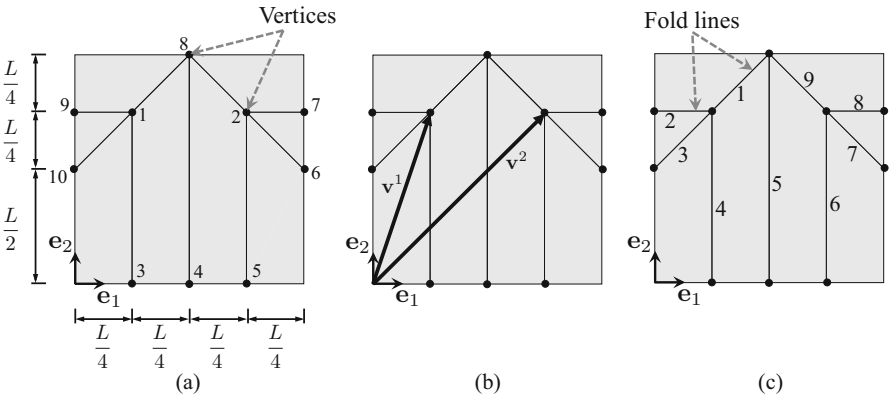


Fig. 2.28 Reference configuration $\mathcal{S}_0$ of the origami airplane illustrated in Fig. 2.1: (**a**) Numbering of the vertices and dimensions of the sheet; (**b**) Position vectors of the first and second vertices; (**c**) Numbering of the fold lines

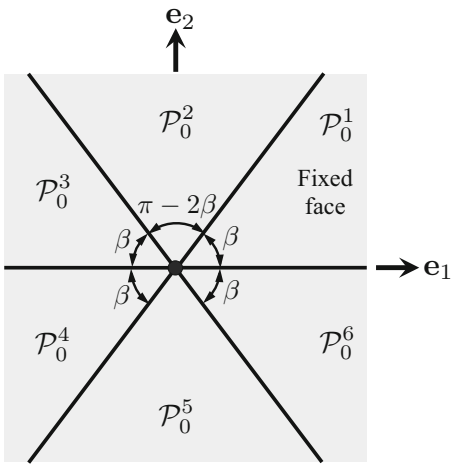


Fig. 2.29 Schematic for Problem 2.2

using a path $\boldsymbol{\gamma}^j(\eta)$ oriented clockwise (i.e., crossing the folds with associated fold vectors in the following order: $\mathbf{m}^{jn_j}, \ldots, \mathbf{m}^{j2}, \mathbf{m}^{j1}$). What would be the folding transformation matrix in the form of (2.27) associated with such folds? Is the resulting constraint equivalent to that provided in (2.39)?

2.4 Verify (2.37) using the definition of α_{jk} provided in (2.14).

2.5 Figure 2.30 shows a fold pattern in a square sheet and fold angles at the final configuration [32, 33]. The reference and final configurations are also shown. Verify that the kinematic constraint for origami with creased folds in (2.39) is satisfied for the fold angles shown. Discuss why the final configuration contains self-

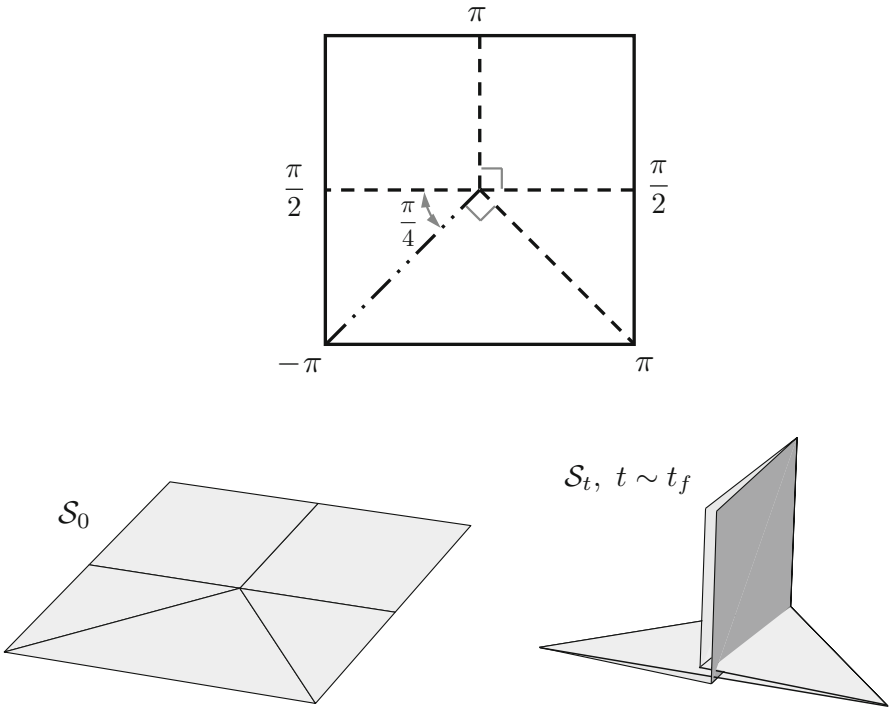


Fig. 2.30 Schematic for Problem 2.5 of a sheet exhibiting self-intersections. Top: Fold pattern and fold angles for each fold at the final configuration [32, 33]. Bottom: Reference configuration $\mathcal{S}_0$ and configuration $\mathcal{S}_t$ (for visualization $t \sim t_f$)

intersections even though the constraint in (2.39) is satisfied. Discuss what could be done to enhance the presented model to account for self-intersection avoidance.

2.6 Verify (2.41) and (2.42) making use of the kinematic constraint provided in (2.39).

2.7 Consider the fold pattern presented in Fig. 2.31. (a) Determine the folding map in the form of (2.63) for each face. (b) Determine the position vector $\mathbf{x}$ in a current configuration with fold angles $\hat{\theta}_1 = \frac{\pi}{2}$ and $\hat{\theta}_2 = -\frac{\pi}{2}$ of the point with reference position vector $\mathbf{X} = [3L \quad 0 \quad 0]^\top$.

2.8 Figure 2.32 shows a fold pattern in a sheet and fold angles at the final configuration [32, 33]. The reference and final configurations are also shown. Verify that the kinematic constraint for origami with creased folds in (2.39) is satisfied for the folds crossed by the paths $\boldsymbol{\gamma}^1(\eta)$ and $\boldsymbol{\gamma}^2(\eta)$. Determine the result of the multiplication of transformation matrices in the form of (2.58) of all the ordered folds crossed by the path $\boldsymbol{\gamma}^{12}(\eta)$. Before calculating it, is it expected to be the identity matrix? Why or why not?

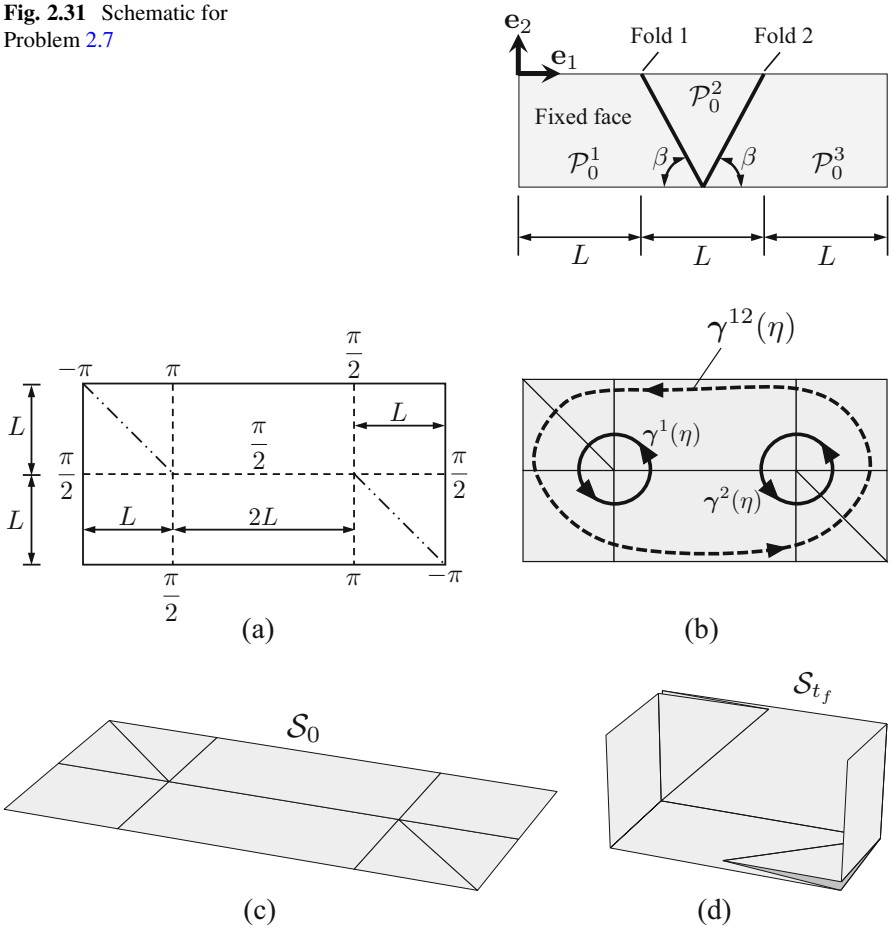


Fig. 2.31 Schematic for Problem 2.7

Fig. 2.32 Schematic for Problem 2.8: (**a**) Fold pattern and fold angles for each fold at the final configuration [32, 33]; (**b**) Various paths enclosing interior fold intersections; (**c**) Reference configuration $\mathcal{S}_0$; (**d**) Final configuration $\mathcal{S}_{t_f}$ (for visualization, the magnitudes of the fold angles at the final configuration are slightly lower than those shown in (**a**))

2.9 Figure 2.33a shows a region of an origami sheet containing three interior fold intersections. Their associated paths $\boldsymbol{\gamma}^j(\eta)$, $\boldsymbol{\gamma}^k(\eta)$, and $\boldsymbol{\gamma}^l(\eta)$ are shown. Assume that the kinematic constraint for origami with creased folds in (2.39) is satisfied for the folds crossed by the paths $\boldsymbol{\gamma}^j(\eta)$, $\boldsymbol{\gamma}^k(\eta)$, and $\boldsymbol{\gamma}^l(\eta)$. Show that the multiplication of transformation matrices in the form of (2.58) of all the ordered folds crossed by the paths $\boldsymbol{\gamma}^{jk}(\eta)$, $\boldsymbol{\gamma}^{jl}(\eta)$, and $\boldsymbol{\gamma}^{jkl}(\eta)$ (shown in Fig. 2.33b, c, and d, respectively) is the identity matrix. Discuss what are the implications of the results obtained in this problem on the resulting folding motion of the sheet. Can these results be generalized to paths in the sheet enclosing any arbitrary number of fold intersections? How?

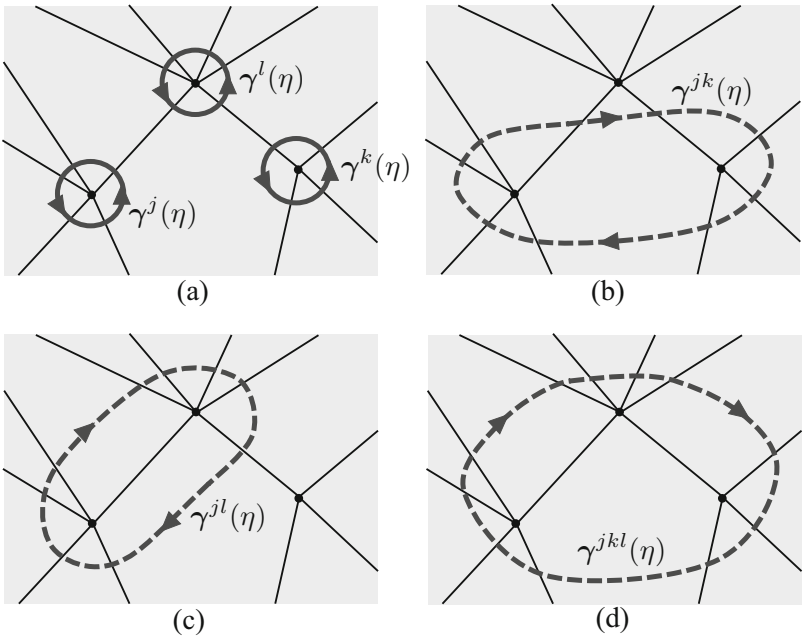


Fig. 2.33 Schematic for Problem 2.9

2.10 Using the results from Problem 2.9, show that the folding map derived in Sect. 2.5 is independent from the path $\boldsymbol{\gamma}^j(\eta)$ chosen to connect the fixed face to a face $\mathcal{P}_0^j$.

2.11 Implement the procedure for kinematic simulation of origami sheets with creased folds presented in this chapter. Then, create the fold patterns and simulate the folding motion of: (a) A box having four corners with the fold pattern used in Fig. 2.14; (b) Origami helixes having the parameterization shown in Fig. 2.23 with parameter values $\{L_1 = L_2, n = 2\}$ and $\{L_1 = L_2, n = 50\}$; (c) An origami self-similar wave having the parameterization shown in Fig. 2.24 and parameter $n = 12$.

References

1. R.J. Lang, The science of origami. Phys. World **20**(2), 30–31 (2007)
2. E.D. Demaine, M.L. Demaine, Recent results in computational origami, in *Proceedings of the 3rd International Meeting of Origami Science, Math, and Education* (Citeseer, 2001), pp. 3–16
3. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Kinematics of origami structures with smooth folds. J. Mech. Robot. **8**(6), 061019 (2016)
4. T. Tachi, Simulation of rigid origami, in *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education* (2009), pp. 175–187

5. J.M. Gattas, W. Wu, Z. You, Miura-base rigid origami: parameterizations of first-level derivative and piecewise geometries. J. Mech. Des. **135**(11), 111011 (2013)
6. S.N. Le, S.-J. Leow, T.-V. Le-Nguyen, C. Ruiz, K.-L. Low, Surface and contour-preserving origamic architecture paper pop-ups. IEEE Trans. Vis. Comput. Graph. **20**(2), 276–288 (2014)
7. X.-Y. Li, T. Ju, Y. Gu, S.-M. Hu, A geometric study of V-style pop-ups: theories and algorithms, in *ACM SIGGRAPH 2011 Papers, SIGGRAPH '11* (ACM, New York, 2011), pp. 98:1–98:10. ISBN 978-1-4503-0943-1
8. P.M. Reis, F. López Jiménez, J. Marthelot, Transforming architectures inspired by origami. Proc. Natl. Acad. Sci. **112**(40), 12234–12235 (2015)
9. T. Tachi, Geometric considerations for the design of rigid origami structures, in *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*, vol. 12 (2010), pp. 458–460
10. T. Tachi, Designing freeform origami tessellations by generalizing Resch's patterns. J. Mech. Des. **135**(11), 111006 (2013)
11. T. Tachi, Freeform origami tessellations by generalizing Resch's patterns, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2013-12326 (American Society of Mechanical Engineers, New York, 2013), pp. V06BT07A025
12. K. Wang, Y. Chen, Folding a patterned cylinder by rigid origami, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (2011), pp. 265–276
13. L.A. Bowen, C.L. Grames, S.P. Magleby, L.L. Howell, R.J. Lang, A classification of action origami as systems of spherical mechanisms. J. Mech. Des. **135**(11), 111008 (2013)
14. L.A. Bowen, C.L. Grames, S.P. Magleby, R.J. Lang, L.L. Howell, An approach for understanding action origami as kinematic mechanisms, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2013-13407, Portland (American Society of Mechanical Engineers, New York, 2013), pp. V06BT07A044
15. L.A. Bowen, W.L. Baxter, S.P. Magleby, L.L. Howell, A position analysis of coupled spherical mechanisms found in action origami. Mech. Mach. Theory **77**, 13–24 (2014)
16. I.L. Delimont, S.P. Magleby, L.L. Howell, Evaluating compliant hinge geometries for origami-inspired mechanisms. J. Mech. Robot. **7**(1), 011009 (2015)
17. T. Evans, Deployable and foldable arrays of spatial mechanisms, Master's thesis, Brigham Young University-Provo, 2015
18. W. Gao, K. Ramani, R.J. Cipra, Reconfigurable foldable spatial mechanisms and robotic forms inspired by kinetogami, in *Proceedings of the ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/-CIE*, No. DETC2012-71403 (American Society of Mechanical Engineers, New York, 2012), pp. 1161–1168
19. B.H. Hanna, S.P. Magleby, R.J. Lang, L.L. Howell, Force–deflection modeling for generalized origami waterbomb-base mechanisms. J. Appl. Mech. **82**(8), 081001 (2015)
20. R.J. Lang, *Origami In Action: Paper Toys That Fly, Flag, Gobble and Inflate!* (St. Martin's Griffin, New York, 1997)
21. E.R. Leal, J.S. Dai, From origami to a new class of centralized 3-DOF parallel mechanisms, in *Proceedings of the ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE* (American Society of Mechanical Engineers, New York, 2007), pp. 1183–1193
22. Y. Qin, J. Dai, Four motion branches of an origami based eight bar spatial mechanism, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2013-12584, Portland (2013), pp. V06BT07A030
23. G. Wei, J.S. Dai, Geometry and kinematic analysis of an origami-evolved mechanism based on artmimetics, in *Proceedings of the 2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots ReMAR* (IEEE, Piscataway, 2009), pp. 450–455

24. G. Wei, J. S. Dai, Origami-inspired integrated planar-spherical overconstrained mechanisms. J. Mech. Des. **136**(5), 051003 (2014)
25. B.G. Winder, S.P. Magleby, L.L. Howell, Kinematic representations of pop-up paper mechanisms. J. Mech. Robot. **1**(2), 021009 (2009)
26. A. Yellowhorse, L.L. Howell, Creating rigid foldability to enable mobility of origami-inspired mechanisms. J. Mech. Robot. **8**(1), 011011 (2016)
27. E.D. Demaine, J. O'Rourke, *Geometric Folding Algorithms* (Cambridge University Press, Cambridge, 2007)
28. E.T. Filipov, T. Tachi, G.H. Paulino, Origami tubes assembled into stiff, yet reconfigurable structures and metamaterials. Proc. Natl. Acad. Sci. **112**(40), 12321–12326 (2015)
29. K. Kuribayashi, K. Tsuchiya, Z. You, D. Tomus, M. Umemoto, T. Ito, M. Sasaki, Self-deployable origami stent grafts as a biomedical application of Ni-rich TiNi shape memory alloy foil. Mater. Sci. Eng. A **419**(1–2), 131–137 (2006)
30. D.-Y. Lee, J.-S. Kim, S.-R. Kim, J.-S. Koh, K.-J. Cho, The deformable wheel robot using magic-ball origami structure, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper DETC2013-13016 (American Society of Mechanical Engineers, New York, 2013), pp. V06BT07A040
31. Y. Chen, W. Lv, J. Li, Z. You, An extended family of rigidly foldable origami tubes. J. Mech. Robot. **9**(2), 021002 (2017)
32. S.-M. Belcastro, T.C. Hull, A mathematical model for non-flat origami, in *Origami 3: Third International Meeting of Origami Mathematics, Science, and Education* (2002), pp. 39–51
33. S.-M. Belcastro, T.C. Hull, Modelling the folding of paper into three dimensions using affine transformations. Linear Algebra Appl. **348**(1–3), 273–282 (2002)
34. T. Tachi, Freeform variations of origami. J. Geom. Graph. **14**(2), 203–215 (2010)
35. A.N. Pressley, *Elementary Differential Geometry* (Springer Science & Business Media, London, 2010)
36. C.R. Calladine, *Theory of Shell Structures* (Cambridge University Press, Cambridge, 1989)
37. E. Akleman, J. Chen, Insight for practical subdivision modeling with discrete gauss-bonnet theorem, in *International Conference on Geometric Modeling and Processing* (Springer, Berlin, 2006), pp. 287–298
38. M. Meyer, M. Desbrun, P. Schröder, A.H. Barr, Discrete differential-geometry operators for triangulated 2-manifolds, in *Visualization and Mathematics III* (Springer, Berlin, 2003), pp. 35–57
39. J.M. Sullivan, Curvatures of smooth and discrete surfaces, in *Discrete Differential Geometry* (Springer, Berlin, 2008), pp. 175–188
40. M.E. Mortenson, *Mathematics for Computer Graphics Applications* (Industrial Press, New York, 1999)
41. MathWorks, Filled 3-D polygons in MATLAB (`fill3`), http://www.mathworks.com/help/matlab/ref/fill3.html
42. T. Hull, *Project Origami: Activities for Exploring Mathematics* (CRC Press, Boca Raton, 2012)

Chapter 3
Unfolding Polyhedra Method for the Design of Origami Structures with Creased Folds

Abstract After addressing the kinematics of origami structures with creased folds in the previous chapter, here we present a method for their design to achieve targeted three-dimensional shapes. This chapter addresses the method of *unfolding polyhedra*. The goal shape is represented in this method as a three-dimensional polygonal mesh, termed as the goal mesh. The objective in unfolding polyhedra is to find the shape and fold pattern of a planar sheet that can be folded towards a configuration that matches the goal mesh. We examine the theory and computational implementation aspects of the unfolding polyhedra method and provide various representative examples.

3.1 Introduction

As reviewed in Chap. 1, the potential engineering advantages of origami structures include storage/deployment capabilities and reduction in manufacturing complexity [1–6]. Such advantages lie in the use of folding to form complex three-dimensional shapes from planar sheets and have allowed origami to be utilized in applications ranging from personal fabrication of customized items [7, 8] to biomedical instruments [9–11] and deployable aerospace structures [12–14].

The process of creating an origami structure with desired characteristics (i.e., a desired shape) is known as *origami design* [15]. Origami design is perhaps the fundamental challenge faced by artists and engineers that apply origami in various fields. Prior to the extensive interest from various research communities seen in the present day, most origami design was performed through trial and error or other heuristic approaches based on the intuition of an experienced artist or designer [16]. With the increase in complexity of origami shapes providing various engineering utilities, computational methods for origami design have become essential [2, 16, 17]. Given the kinematic models for origami on which they are

Electronic Supplementary Material The online version of this article (https://doi.org/10.1007/978-3-319-91866-2_3) contains supplementary material, which is available to authorized users.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_3

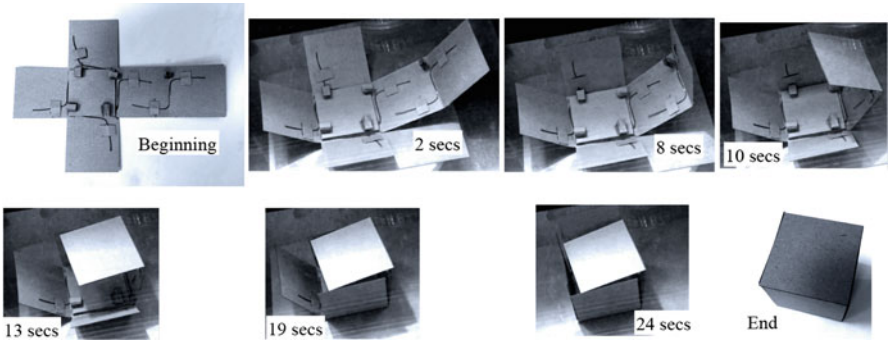


Fig. 3.1 A self-folding cube developed using the unfolding polyhedra method. Upon heating, the folding motion towards the cube shape is driven by the actuation of torsional shape memory alloy (SMA) wires. Credit: Beatriz Borges

based, current methods for origami design generally consider rigid faces and straight creased folds [17–19]. We provide a review of such methods in Sect. 1.3.

A method for determining the shape and fold pattern of a planar sheet that can be folded towards a targeted polyhedral surface is known as *unfolding polyhedra* [17, 20–22], as the objective in this method is to determine an "unfolding" of the polyhedral surface. An *unfolding* is the flattening of the goal polyhedral surface towards a planar shape that has boundary segments corresponding to *cuts* made on the polyhedral surface. Usually, *the unfolding is required to be a single sheet that does not have any overlaps* and *the cuts are required to be made exclusively on edges of the goal polyhedral surface*. An unfolding that satisfies these characteristics is called a *net* [17]. An example of an active origami structure developed using the unfolding polyhedra method is shown in Fig. 3.1. It is noted that the sheet in Fig. 3.1 has no overlaps in its planar configuration and was obtained by making cuts solely on edges of the cube; thus, it is a net.

Unfolding polyhedra is a mathematical problem that has been studied for centuries and several challenges still remain on the topic [17] (see Sect. 3.2.6). In the sixteenth century, polyhedra became geometric objects of special interest to painters as they were developing the rules of perspective. In that century, painter Albrecht Dürer published "The Painter's Manual," which contained the first examples of nets of polyhedra in the literature [17]. During the twentieth century, the interest on solving problems in unfolding polyhedra increased in the mathematics and computer science communities [17]. Even though nets of polyhedra were initially used to make paper origami [23, 24], unfolding polyhedra has found important industrial applications. Such applications include sheet metal folding as an efficient process for manufacturing of three-dimensional objects [25, 26]. In that process, the goal shape of the three-dimensional object is approximated as a polygonal mesh that is then unfolded into a net (or multiple nets for practical purposes such as efficient packing into a rectangle of metal). Afterwards, the nets are cut out of sheets of metal and folded into the goal shape using a bending machine.

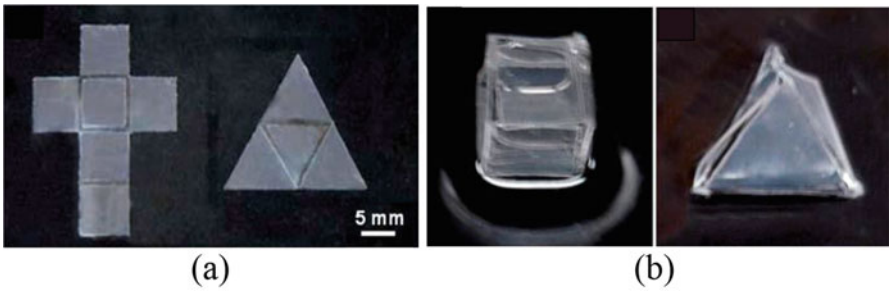


Fig. 3.2 Optical micrographs of self-folding structures with hydrogel-actuated folds: cube and tetrahedron. (**a**) Reference planar configurations determined using the unfolding polyhedra method. (**b**) Final goal configurations. Adapted from [28] with permission of The Royal Society of Chemistry

Unfolding polyhedra has also been employed in the development of active origami structures due to its relative simplicity compared with other origami design methods. An and coworkers developed an approach for the synthesis of three-dimensional surface objects via self-folding of planar sheets activated by uniform heating [27]. Their approach used the unfolding polyhedra method to determine the geometry and fold pattern of the planar sheets, which are then folded through shape memory polymer actuation. Other examples of the use of unfolding polyhedra in the design of active origami structures are provided in Chap. 1. These examples include thermally activated (Fig. 1.2), and electromagnetically activated (Fig. 1.19) self-folding structures. Examples of chemically activated self-folding structures developed using the unfolding polyhedra method are shown in Fig. 3.2.

There is ongoing research on more general unfoldings comprised of multiple disconnected regions or having cuts that are not limited to edges of the goal polyhedral surface [17, 29–32]. In this chapter, however, we study the unfolding polyhedra method considering unfoldings that have the characteristics of a net. Section 3.2 presents the problem description and solution approach of the unfolding polyhedra method. A discussion of the limitations of the unfolding polyhedra method is provided in Sect. 3.2.6 and examples of the implemented method considering various goal shapes are provided in Sect. 3.3.

3.2 Unfolding Polyhedra Method Considering Creased Folds

Here we describe the various aspects of the unfolding polyhedra method for the design of origami structures with creased folds. Section 3.2.1 provides the design problem definition and the solution approach, the goal mesh is described in Sect. 3.2.2, the steps of the design method are addressed in Sects. 3.2.3–3.2.5, and limitations of the method are discussed in Sect. 3.2.6.

3.2.1 Problem Definition

The unfolding polyhedra method aims to solve the following general problem in origami design:

- *Given*: A three-dimensional goal shape represented as a polygonal mesh (termed as the *goal mesh*[1] $\mathcal{M}$),
- *Find*: The shape and fold pattern of a planar sheet that can be folded to match $\mathcal{M}$, and a history of folding motion from the planar configuration of the sheet ($\mathcal{S}_0$) to the goal configuration ($\mathcal{S}_\star$) that matches $\mathcal{M}$

The steps followed in the unfolding polyhedra method to solve this general problem in origami design are illustrated in Fig. 3.3 and enumerated as follows:

1. Determination of a *spanning tree* on the goal mesh $\mathcal{M}$ (Fig. 3.3b) [33, 34]. A spanning tree is a line graph on $\mathcal{M}$ that contains a reference point in each face of $\mathcal{M}$. Such a graph is called a "tree" because it is branched and does not contain any loops. The spanning tree cannot contain any node of $\mathcal{M}$ and cannot cross any edge of $\mathcal{M}$ more than once. The importance of the spanning tree is not its particular shape, but rather the set of edges of $\mathcal{M}$ that it crosses
2. Assignment of every edge that is *not* crossed by the spanning tree as a *boundary edge* (Fig. 3.3c). Since only the edges crossed by spanning tree are not assigned as boundary edges, the resulting mesh corresponds to a tree of faces (i.e., such a union of faces does not form any loop) that has the topology of the spanning tree
3. Mapping the tree of faces obtained in Step 2 onto a plane. We denote this mapping as the *unfolding map*. It produces a *net* of the goal mesh, which corresponds to the reference configuration $\mathcal{S}_0$ of an origami sheet as shown in Fig. 3.3d
4. Determination of a history of folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$ (Fig. 3.3e)

We examine the algorithmic aspects associated with each of the aforementioned steps in the following sections. The data required to define the goal mesh $\mathcal{M}$ is described in Sect. 3.2.2. The procedure used to determine a spanning tree for any $\mathcal{M}$ is presented in Sect. 3.2.3. The formulation used to map the faces of $\mathcal{M}$ towards their position in $\mathcal{S}_0$ (i.e., the unfolding map) is presented in Sect. 3.2.4. The determination of a history of folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ is addressed in Sect. 3.2.5. We provide a set of MATLAB® scripts that perform the aforementioned steps in the Supplemental Material of this chapter. These MATLAB scripts are described in Appendix B.2.

[1] The goal mesh $\mathcal{M}$ is a *connected*, *orientable*, *2-manifold* polygonal mesh.

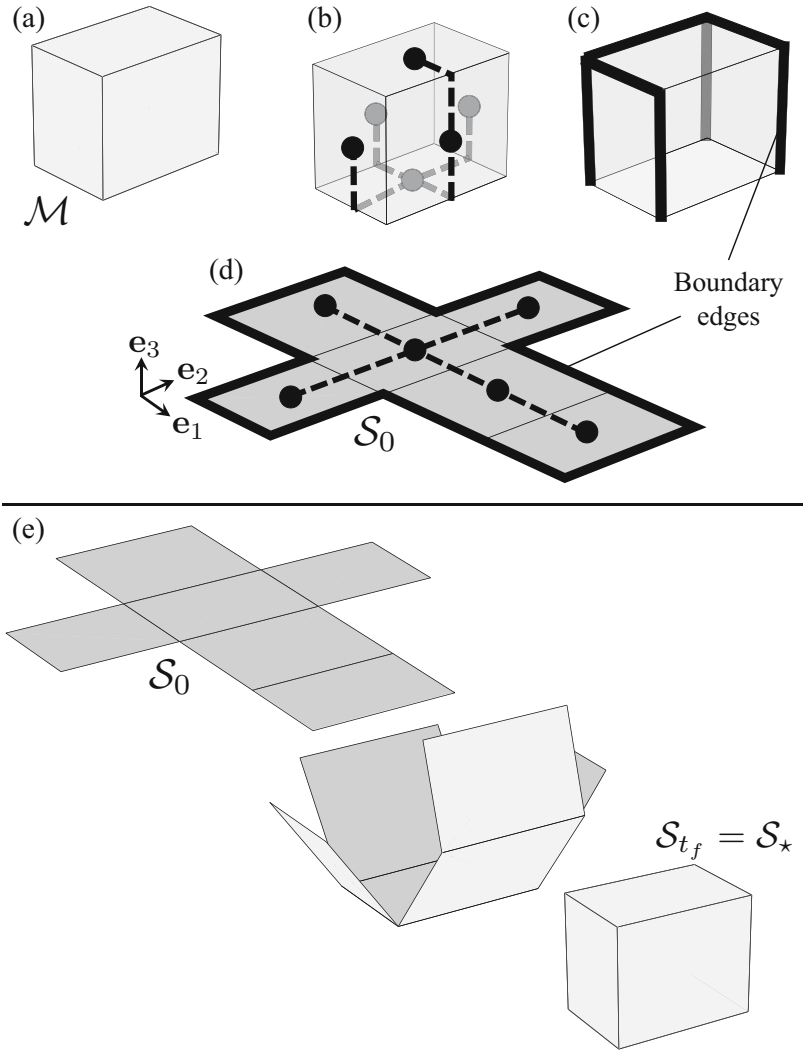


Fig. 3.3 Summary of the unfolding polyhedra method: (**a**) Given goal mesh $\mathcal{M}$; (**b**) Spanning tree including all the faces of $\mathcal{M}$; (**c**) The edges that are not crossed by the spanning tree are assigned as boundary edges; (**d**) Determined net corresponding to the reference configuration $\mathcal{S}_0$ of an origami sheet; (**e**) Continuous folding motion from $t = 0$ to $t = t_f$ such that the final configuration $\mathcal{S}_{t_f}$ corresponds to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

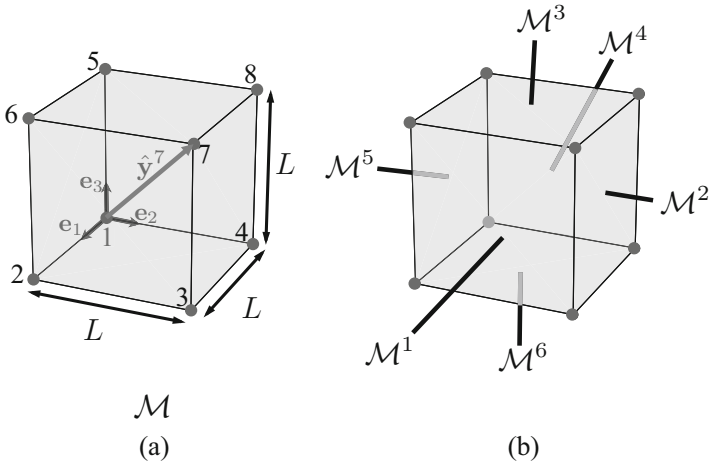


Fig. 3.4 Goal mesh $\mathcal{M}$ of a cube: (**a**) Dimensions, node labels, and the position vector of a node; (**b**) Labels of the mesh faces

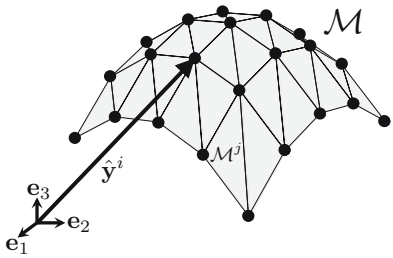


Fig. 3.5 Goal mesh $\mathcal{M}$ of a dome. The position vector of a node $\hat{\mathbf{y}}^i$ and a mesh face $\mathcal{M}^j$ are indicated

3.2.2 Goal Mesh Description

The description of the data required to define a goal mesh $\mathcal{M}$ is provided in this section. This description is used in the subsequent chapters of this book where we study methods for origami design (Chaps. 4, 6, 7, and 8).

We illustrate two different goal meshes in Figs. 3.4 (cube) and 3.5 (dome). Following the coordinate system convention of Sect. 2.2, the orthonormal vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \in \mathbb{R}^3$ form the basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ that defines the fixed global coordinate system.

The goal mesh $\mathcal{M}$ has a total of $N_{\mathcal{M}}$ faces, $N_{\mathcal{E}}$ edges, and $N_{\mathcal{N}}$ nodes. For the cube shown in Fig. 3.4, $N_{\mathcal{M}} = 6$, $N_{\mathcal{E}} = 12$, and $N_{\mathcal{N}} = 8$. For the dome shown in Fig. 3.5, $N_{\mathcal{M}} = 32$, $N_{\mathcal{E}} = 56$, and $N_{\mathcal{N}} = 25$. The *position vectors of the nodes* of $\mathcal{M}$ are denoted $\hat{\mathbf{y}}^1, \ldots, \hat{\mathbf{y}}^{N_{\mathcal{N}}} \in \mathbb{R}^3$. The *mesh faces* of $\mathcal{M}$ are denoted $\mathcal{M}^1, \ldots, \mathcal{M}^{N_{\mathcal{M}}}$ such that $\mathcal{M} = \bigcup_{j=1}^{N_{\mathcal{M}}} \mathcal{M}^j$.

Each mesh face $\mathcal{M}^j$ has $n_j^{\mathcal{M}}$ nodes. For all the faces of the cube shown in Fig. 3.4, $n_j^{\mathcal{M}} = 4$, $j = 1, \ldots, 6$. For all the faces of the dome shown in

Table 3.1 Input parameters required to define a goal mesh $\mathcal{M}$

Parameter	Definition
Position vectors of the nodes $\hat{\mathbf{y}}^1, \ldots, \hat{\mathbf{y}}^{N_\mathcal{N}}$	
Mesh connectivity matrix $\mathbf{C}^\mathcal{M}$	(3.1)

Fig. 3.5, $n_j^\mathcal{M} = 3$, $j = 1, \ldots, 32$. We introduce the *mesh connectivity matrix* $\mathbf{C}^\mathcal{M} \in \mathbb{R}^{N_\mathcal{M} \times \max(n_j^\mathcal{M})}$ to indicate which nodes of $\mathcal{M}$ are associated with each of the faces $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}}$. The components of $\mathbf{C}^\mathcal{M}$ are denoted $C_{jk}^\mathcal{M}$ and are defined as follows:

$$C_{jk}^\mathcal{M} = \text{Index of the node at the } k\text{th corner (in counterclockwise order) of } \mathcal{M}^j$$
$$j = 1, \ldots, N_\mathcal{M}, \quad k = 1, \ldots, n_j^\mathcal{M}. \tag{3.1}$$

The input parameters required to define a goal mesh $\mathcal{M}$ are listed in Table 3.1.

Example 3.1 Goal mesh data for a cube.
Statement: Consider the node and face labels for the cube shown in Fig. 3.4 and provide the following data: (a) The components of the position vector of each node; (b) The components of the mesh connectivity matrix $\mathbf{C}^\mathcal{M}$.
Solution:

(a) For the cube shown in Fig. 3.4, the position vectors of the nodes $\hat{\mathbf{y}}^1, \ldots, \hat{\mathbf{y}}^8$ are given as follows:

$$\hat{\mathbf{y}}^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \hat{\mathbf{y}}^2 = \begin{bmatrix} L \\ 0 \\ 0 \end{bmatrix}, \quad \hat{\mathbf{y}}^3 = \begin{bmatrix} L \\ L \\ 0 \end{bmatrix}, \quad \hat{\mathbf{y}}^4 = \begin{bmatrix} 0 \\ L \\ 0 \end{bmatrix},$$
$$\hat{\mathbf{y}}^5 = \begin{bmatrix} 0 \\ 0 \\ L \end{bmatrix}, \quad \hat{\mathbf{y}}^6 = \begin{bmatrix} L \\ 0 \\ L \end{bmatrix}, \quad \hat{\mathbf{y}}^7 = \begin{bmatrix} L \\ L \\ L \end{bmatrix}, \quad \hat{\mathbf{y}}^8 = \begin{bmatrix} 0 \\ L \\ L \end{bmatrix}. \tag{3.2}$$

(b) The mesh connectivity matrix $\mathbf{C}^\mathcal{M}$ is given as follows:

$$\mathbf{C}^\mathcal{M} = \begin{bmatrix} 2 & 3 & 7 & 6 \\ 3 & 4 & 8 & 7 \\ 5 & 6 & 7 & 8 \\ 4 & 1 & 5 & 8 \\ 1 & 2 & 6 & 5 \\ 4 & 3 & 2 & 1 \end{bmatrix}. \tag{3.3}$$

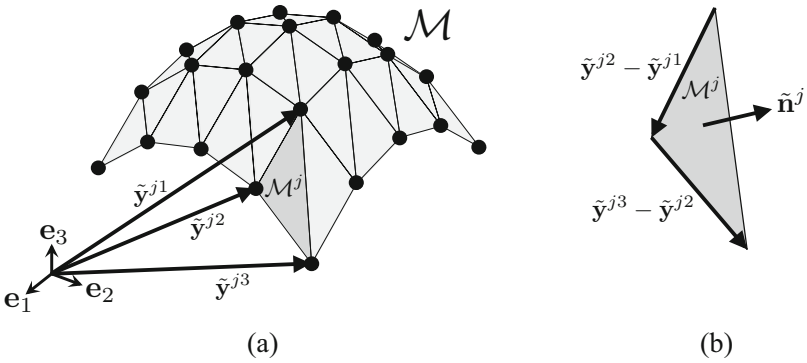


Fig. 3.6 (**a**) Position vectors of the nodes of $\mathcal{M}^j$ ($\tilde{\mathbf{y}}^{j1}$, $\tilde{\mathbf{y}}^{j2}$, and $\tilde{\mathbf{y}}^{j3}$). (**b**) Unit normal vector of $\mathcal{M}^j$ ($\tilde{\mathbf{n}}^j$)

Let $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}} \in \mathbb{R}^3$, $j = 1, \ldots, N_{\mathcal{M}}$, be the position vectors of the nodes of the mesh face $\mathcal{M}^j$ (see Fig. 3.6a). We use the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ to map the position vectors of all the nodes of $\mathcal{M}$ to the position vectors $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}}$, $j = 1, \ldots, N_{\mathcal{M}}$, as follows:

$$\tilde{\mathbf{y}}^{jk} = \hat{\mathbf{y}}^{C_{jk}^{\mathcal{M}}} \qquad j = 1, \ldots, N_{\mathcal{M}}, \quad k = 1, \ldots, n_j^{\mathcal{M}}. \tag{3.4}$$

The *unit normal vector* of $\mathcal{M}^j$ is denoted $\tilde{\mathbf{n}}^j \in \mathbb{R}^3$ and is determined as follows (see Fig. 3.6b):

$$\tilde{\mathbf{n}}^j = \frac{(\tilde{\mathbf{y}}^{j2} - \tilde{\mathbf{y}}^{j1}) \times (\tilde{\mathbf{y}}^{j3} - \tilde{\mathbf{y}}^{j2})}{\|(\tilde{\mathbf{y}}^{j2} - \tilde{\mathbf{y}}^{j1}) \times (\tilde{\mathbf{y}}^{j3} - \tilde{\mathbf{y}}^{j2})\|}. \tag{3.5}$$

Example 3.2 Data for a face in the cube.
Statement: Determine the position vectors of the nodes of $\mathcal{M}^3$ in the cube shown in Fig. 3.4. Also determine the unit normal vector of $\mathcal{M}^3$.
Solution: We use (3.4) to determine the position vectors of the nodes of mesh face $\mathcal{M}^3$:

$$\tilde{\mathbf{y}}^{31} = \hat{\mathbf{y}}^{C_{31}^{\mathcal{M}}}, \qquad \tilde{\mathbf{y}}^{32} = \hat{\mathbf{y}}^{C_{32}^{\mathcal{M}}}, \qquad \tilde{\mathbf{y}}^{33} = \hat{\mathbf{y}}^{C_{33}^{\mathcal{M}}}, \qquad \tilde{\mathbf{y}}^{34} = \hat{\mathbf{y}}^{C_{34}^{\mathcal{M}}}. \tag{3.6}$$

Substituting (3.3) into (3.6):

$$\tilde{\mathbf{y}}^{31} = \hat{\mathbf{y}}^{5}, \qquad \tilde{\mathbf{y}}^{32} = \hat{\mathbf{y}}^{6}, \qquad \tilde{\mathbf{y}}^{33} = \hat{\mathbf{y}}^{7}, \qquad \tilde{\mathbf{y}}^{34} = \hat{\mathbf{y}}^{8}, \tag{3.7}$$

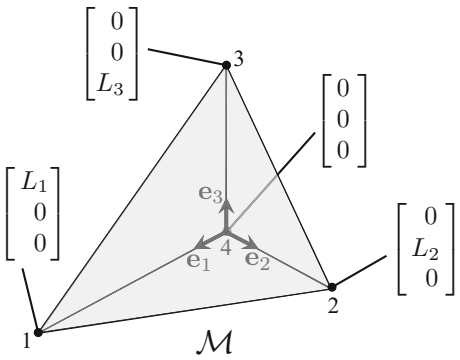


Fig. 3.7 Goal mesh of a tetrahedron and coordinates of the nodes

and substituting (3.2) in (3.7):

$$\tilde{\mathbf{y}}^{31} = \begin{bmatrix}0\\0\\L\end{bmatrix}, \qquad \tilde{\mathbf{y}}^{32} = \begin{bmatrix}L\\0\\L\end{bmatrix}, \qquad \tilde{\mathbf{y}}^{33} = \begin{bmatrix}L\\L\\L\end{bmatrix}, \qquad \tilde{\mathbf{y}}^{34} = \begin{bmatrix}0\\L\\L\end{bmatrix}. \tag{3.8}$$

We use (3.5) to determine the unit normal vector of face $\mathcal{M}^3$ (i.e., $\tilde{\mathbf{n}}^3$) of the cube shown in Fig. 3.4 as follows:

$$\begin{aligned}\tilde{\mathbf{n}}^3 &= \frac{(\tilde{\mathbf{y}}^{32} - \tilde{\mathbf{y}}^{31}) \times (\tilde{\mathbf{y}}^{33} - \tilde{\mathbf{y}}^{32})}{\|(\tilde{\mathbf{y}}^{32} - \tilde{\mathbf{y}}^{31}) \times (\tilde{\mathbf{y}}^{33} - \tilde{\mathbf{y}}^{32})\|} \\ &= \frac{1}{L^2}\begin{bmatrix}0\\0\\L^2\end{bmatrix} \qquad \text{substituting (3.8)} \\ &= \begin{bmatrix}0\\0\\1\end{bmatrix}.\end{aligned} \tag{3.9}$$

Example 3.3 Parameters for the goal mesh of a tetrahedron.
Statement: Consider the goal mesh $\mathcal{M}$ of the tetrahedron illustrated in Fig. 3.7. (a) Determine the position vectors of the nodes $\hat{\mathbf{y}}^i$, $i = 1, \ldots, N_{\mathcal{N}}$, and the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$. Define $\mathbf{C}^{\mathcal{M}}$ such that the unit normal vectors of the mesh faces point towards the exterior of the tetrahedron. (b) Determine the vectors $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}}$ and unit normal vector $\tilde{\mathbf{n}}^j$ of each mesh face.
Solution:

(a) From the schematic in Fig. 3.7, we observe that $N_{\mathcal{N}} = 4$ and $N_{\mathcal{M}} = 4$. The position vectors of the nodes of $\mathcal{M}$ are given as:

$$\hat{\mathbf{y}}^1 = \begin{bmatrix} L_1 \\ 0 \\ 0 \end{bmatrix}, \quad \hat{\mathbf{y}}^2 = \begin{bmatrix} 0 \\ L_2 \\ 0 \end{bmatrix}, \quad \hat{\mathbf{y}}^3 = \begin{bmatrix} 0 \\ 0 \\ L_3 \end{bmatrix}, \quad \hat{\mathbf{y}}^4 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \tag{3.10}$$

Enumerating the faces such that $\mathcal{M}^1$ is the face in the 2-3 plane, $\mathcal{M}^2$ is the face in the 3-1 plane, $\mathcal{M}^3$ is the face in the 1-2 plane, and $\mathcal{M}^4$ is the remainder face, the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ is given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 4 & 3 & 2 \\ 4 & 1 & 3 \\ 4 & 2 & 1 \\ 1 & 2 & 3 \end{bmatrix}. \tag{3.11}$$

(b) After substituting (3.10) and (3.11) into (3.4), the vectors $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{j3}$, $j = 1, \ldots, 4$, are obtained as follows:

$$\begin{aligned}
&\text{Face } \mathcal{M}^1: \quad \tilde{\mathbf{y}}^{11} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{12} = \begin{bmatrix} 0 \\ 0 \\ L_3 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{13} = \begin{bmatrix} 0 \\ L_2 \\ 0 \end{bmatrix}, \\
&\text{Face } \mathcal{M}^2: \quad \tilde{\mathbf{y}}^{21} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{22} = \begin{bmatrix} L_1 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{23} = \begin{bmatrix} 0 \\ 0 \\ L_3 \end{bmatrix}, \\
&\text{Face } \mathcal{M}^3: \quad \tilde{\mathbf{y}}^{31} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{32} = \begin{bmatrix} 0 \\ L_2 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{33} = \begin{bmatrix} L_1 \\ 0 \\ 0 \end{bmatrix}, \\
&\text{Face } \mathcal{M}^4: \quad \tilde{\mathbf{y}}^{41} = \begin{bmatrix} L_1 \\ 0 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{42} = \begin{bmatrix} 0 \\ L_2 \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{y}}^{43} = \begin{bmatrix} 0 \\ 0 \\ L_3 \end{bmatrix}.
\end{aligned} \tag{3.12}$$

The unit normal vectors $\tilde{\mathbf{n}}^1, \ldots, \tilde{\mathbf{n}}^4$ are determined using (3.5):

$$\text{Face } \mathcal{M}^1: \quad \tilde{\mathbf{n}}^1 = \frac{(\tilde{\mathbf{y}}^{12} - \tilde{\mathbf{y}}^{11}) \times (\tilde{\mathbf{y}}^{13} - \tilde{\mathbf{y}}^{12})}{\|(\tilde{\mathbf{y}}^{12} - \tilde{\mathbf{y}}^{11}) \times (\tilde{\mathbf{y}}^{13} - \tilde{\mathbf{y}}^{12})\|} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix},$$

$$\text{Face } \mathcal{M}^2: \quad \tilde{\mathbf{n}}^2 = \frac{(\tilde{\mathbf{y}}^{22} - \tilde{\mathbf{y}}^{21}) \times (\tilde{\mathbf{y}}^{23} - \tilde{\mathbf{y}}^{22})}{\|(\tilde{\mathbf{y}}^{22} - \tilde{\mathbf{y}}^{21}) \times (\tilde{\mathbf{y}}^{23} - \tilde{\mathbf{y}}^{22})\|} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix},$$

$$\text{Face } \mathcal{M}^3: \quad \tilde{\mathbf{n}}^3 = \frac{(\tilde{\mathbf{y}}^{32} - \tilde{\mathbf{y}}^{31}) \times (\tilde{\mathbf{y}}^{33} - \tilde{\mathbf{y}}^{32})}{\|(\tilde{\mathbf{y}}^{32} - \tilde{\mathbf{y}}^{31}) \times (\tilde{\mathbf{y}}^{33} - \tilde{\mathbf{y}}^{32})\|} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix},$$

$$\text{Face } \mathcal{M}^4: \quad \tilde{\mathbf{n}}^4 = \frac{(\tilde{\mathbf{y}}^{42} - \tilde{\mathbf{y}}^{41}) \times (\tilde{\mathbf{y}}^{43} - \tilde{\mathbf{y}}^{42})}{\|(\tilde{\mathbf{y}}^{42} - \tilde{\mathbf{y}}^{41}) \times (\tilde{\mathbf{y}}^{43} - \tilde{\mathbf{y}}^{42})\|}$$

$$= \frac{1}{\left((L_2L_3)^2 + (L_3L_1)^2 + (L_1L_2)^2\right)^{\frac{1}{2}}} \begin{bmatrix} L_2L_3 \\ L_3L_1 \\ L_1L_2 \end{bmatrix}. \tag{3.13}$$

The *number of interior nodes* of $\mathcal{M}$ (i.e., those that are not located at the boundary of the mesh) is denoted $N_{\mathcal{N}}^I$ and the *number of interior edges* is denoted $N_{\mathcal{E}}^I$. The cube shown in Fig. 3.4 is a closed mesh and therefore all of its nodes and edges are, respectively, interior nodes and interior edges (i.e., $N_{\mathcal{N}}^I = N_{\mathcal{N}} = 8$, $N_{\mathcal{E}}^I = N_{\mathcal{E}} = 12$). Conversely, the dome shown in Fig. 3.4 is not a closed mesh and thus it has boundary nodes and edges. From Fig. 3.4, we observe that $N_{\mathcal{N}}^I = 9$ and $N_{\mathcal{E}}^I = 40$ (recall that $N_{\mathcal{N}} = 25$ and $N_{\mathcal{E}} = 56$ for this mesh).

Let $\hat{\mathbf{z}}^i \in \mathbb{R}^3$, $i = 1, \ldots, N_{\mathcal{E}}^I$, be a vector along the length of the ith interior edge of the goal mesh $\mathcal{M}$. To identify which nodes are the start points and end points of each interior edge in $\mathcal{M}$, we introduce the *edge connectivity matrix* $\mathbf{C}^{\mathcal{E}_I} \in \mathbb{R}^{N_{\mathcal{E}}^I \times 2}$ with components $C_{ij}^{\mathcal{E}_I}$ defined as follows:

$$\begin{aligned} C_{i1}^{\mathcal{E}_I} &= \text{Index of the node corresponding to the start point of the } i\text{th interior edge,} \\ C_{i2}^{\mathcal{E}_I} &= \text{Index of the node corresponding to the end point of the } i\text{th interior edge,} \\ i &= 1, \ldots, N_{\mathcal{E}}^I. \end{aligned} \tag{3.14}$$

The components of the edge connectivity matrix $\mathbf{C}^{\mathcal{E}_I}$ can be determined from the given mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ (see Problem 3.3). The mapping from the position vectors of the nodes of $\mathcal{M}$ $(\hat{\mathbf{y}}^1, \ldots, \hat{\mathbf{y}}^{N_{\mathcal{N}}})$ to the vectors along the length of the interior edges of $\mathcal{M}$ $(\hat{\mathbf{z}}^1, \ldots, \hat{\mathbf{z}}^{N_{\mathcal{E}}^I})$ is given as follows:

$$\hat{\mathbf{z}}^i = \hat{\mathbf{y}}^{C_{i2}^{\mathcal{E}_I}} - \hat{\mathbf{y}}^{C_{i1}^{\mathcal{E}_I}} \qquad i = 1, \ldots, N_{\mathcal{E}}^I. \tag{3.15}$$

Each interior node has a total of $n_j^{\mathcal{N}}$ incident edges. We define $\mathbf{z}^{jk} \in \mathbb{R}^3$, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, as the vector along the edge connecting the jth interior node to its kth adjacent node. The vectors $\mathbf{z}^{j1}, \ldots, \mathbf{z}^{jn_j^{\mathcal{N}}}$ are arranged

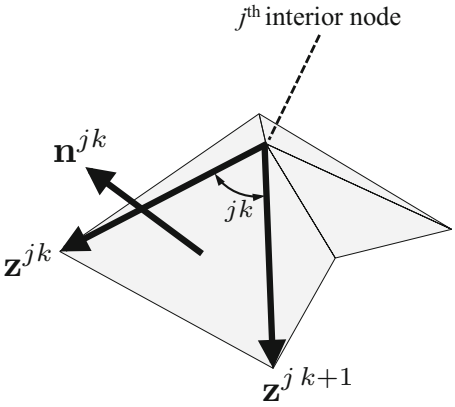


Fig. 3.8 Geometric parameters of the faces in the goal mesh $\mathcal{M}$ connected to a common interior node

in counterclockwise order (see Fig. 3.8). The *node connectivity matrix* $\mathbf{C}^{\mathcal{N}_I}$ with components $C_{jk}^{\mathcal{N}_I}$ is used to identify the edges incident to each interior node:

$$C_{jk}^{\mathcal{N}_I} = \text{Index of the } k\text{th interior edge incident to the } j\text{th interior node (multiplied by } -1 \text{ if the interior edge, as defined in } \mathbf{C}^{\mathcal{E}_I}\text{, ends at the interior node)},$$
$$j = 1, \ldots, N_{\mathcal{N}}^I, \quad k = 1, \ldots n_j^{\mathcal{N}}. \tag{3.16}$$

The components of $\mathbf{C}^{\mathcal{N}_I}$ can also be determined from the given mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ (see Problem 3.4).

The mapping from the vectors along the length of the interior edges of $\mathcal{M}$ ($\hat{\mathbf{z}}^1, \ldots, \hat{\mathbf{z}}^{N_{\mathcal{E}}^I}$) to the vectors $\mathbf{z}^{j1}, \ldots, \mathbf{z}^{jn_j^{\mathcal{N}}}$, $j = 1, \ldots, N_{\mathcal{N}}^I$, is given as follows:

$$\mathbf{z}^{jk} = \begin{cases} \hat{\mathbf{z}}^{C_{jk}^{\mathcal{N}_I}}; & C_{jk}^{\mathcal{N}_I} > 0 \\ -\hat{\mathbf{z}}^{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} < 0, \end{cases} \tag{3.17}$$
$$j = 1, \ldots, N_{\mathcal{N}}^I, \qquad k = 1, \ldots, n_j^{\mathcal{N}}.$$

Note that $\mathbf{z}^{jk}$ has the opposite direction of its associated vector $\hat{\mathbf{z}}^{|C_{jk}^{\mathcal{N}_I}|}$ if $C_{jk}^{\mathcal{N}_I} < 0$. This is applied such that a vector $\mathbf{z}^{jk}$ always emanate from the jth interior node (since $C_{jk}^{\mathcal{N}_I} < 0$ if the considered edge *ends* at the interior node, recall the definition of $\mathbf{C}^{\mathcal{N}_I}$ in (3.16)).

The parameters associated with the faces of $\mathcal{M}$ connected to the jth interior node are the *face corner angles* ϕ_{jk} and the *unit normal vectors* $\mathbf{n}^{jk} \in \mathbb{R}^3$, $j =$

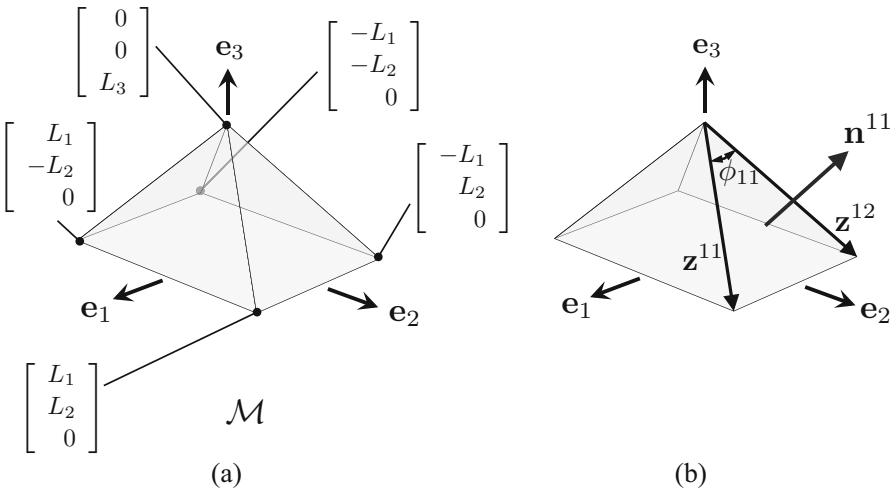


Fig. 3.9 Schematic for Example 3.4. The base of the pyramid is open. (**a**) Goal mesh $\mathcal{M}$ having a single interior node. The coordinates of the nodes are shown. (**b**) Vectors $\mathbf{z}^{11}$ and $\mathbf{z}^{12}$

$1, \ldots, N_{\mathcal{N}}^{I}$, $k = 1, \ldots, n_j^{\mathcal{N}}$. These parameters are determined as follows (see Fig. 3.8):

$$\phi_{jk} = \begin{cases} \cos^{-1}\left(\dfrac{\mathbf{z}^{jk} \cdot \mathbf{z}^{j\,k+1}}{\|\mathbf{z}^{jk}\|\,\|\mathbf{z}^{j\,k+1}\|}\right); & k = 1, \ldots, n_j^{\mathcal{N}} - 1 \\ \cos^{-1}\left(\dfrac{\mathbf{z}^{jk} \cdot \mathbf{z}^{j1}}{\|\mathbf{z}^{jk}\|\,\|\mathbf{z}^{j1}\|}\right); & k = n_j^{\mathcal{N}}, \end{cases} \tag{3.18}$$

$$\mathbf{n}^{jk} = \begin{cases} \dfrac{\mathbf{z}^{jk} \times \mathbf{z}^{j\,k+1}}{\|\mathbf{z}^{jk} \times \mathbf{z}^{j\,k+1}\|}; & k = 1, \ldots, n_j^{\mathcal{N}} - 1 \\ \dfrac{\mathbf{z}^{jk} \times \mathbf{z}^{j1}}{\|\mathbf{z}^{jk} \times \mathbf{z}^{j1}\|}; & k = n_j^{\mathcal{N}}. \end{cases} \tag{3.19}$$

Example 3.4 Determining the face corner angles and the unit normal vectors of the faces connected to an interior node.

Statement: Figure 3.9a illustrates a goal mesh $\mathcal{M}$ having a single interior node. Determine the vectors $\mathbf{z}^{11}, \ldots, \mathbf{z}^{14}$, the face corner angles $\phi_{11}, \ldots, \phi_{14}$, and the unit normal vectors $\mathbf{n}^{11}, \ldots, \mathbf{n}^{14}$.

Solution: From the node coordinates provided in Fig. 3.9a, the vectors $\mathbf{z}^{11}, \ldots, \mathbf{z}^{14}$ are given as follows:

$$\mathbf{z}^{11} = \begin{bmatrix} L_1 \\ L_2 \\ -L_3 \end{bmatrix}, \quad \mathbf{z}^{12} = \begin{bmatrix} -L_1 \\ L_2 \\ -L_3 \end{bmatrix}, \quad \mathbf{z}^{13} = \begin{bmatrix} -L_1 \\ -L_2 \\ -L_3 \end{bmatrix}, \quad \mathbf{z}^{14} = \begin{bmatrix} L_1 \\ -L_2 \\ -L_3 \end{bmatrix}. \tag{3.20}$$

The face corner angles $\phi_{11}, \ldots, \phi_{14}$ are determined via (3.18):

$$\begin{aligned}
\phi_{11} &= \cos^{-1}\left(\frac{\mathbf{z}^{11}\cdot\mathbf{z}^{12}}{\|\mathbf{z}^{11}\|\,\|\mathbf{z}^{12}\|}\right) = \cos^{-1}\left(\frac{-L_1^2+L_2^2+L_3^2}{L_1^2+L_2^2+L_3^2}\right), \\
\phi_{12} &= \cos^{-1}\left(\frac{\mathbf{z}^{12}\cdot\mathbf{z}^{13}}{\|\mathbf{z}^{12}\|\,\|\mathbf{z}^{13}\|}\right) = \cos^{-1}\left(\frac{L_1^2-L_2^2+L_3^2}{L_1^2+L_2^2+L_3^2}\right), \\
\phi_{13} &= \cos^{-1}\left(\frac{\mathbf{z}^{13}\cdot\mathbf{z}^{14}}{\|\mathbf{z}^{13}\|\,\|\mathbf{z}^{14}\|}\right) = \cos^{-1}\left(\frac{-L_1^2+L_2^2+L_3^2}{L_1^2+L_2^2+L_3^2}\right), \\
\phi_{14} &= \cos^{-1}\left(\frac{\mathbf{z}^{14}\cdot\mathbf{z}^{11}}{\|\mathbf{z}^{14}\|\,\|\mathbf{z}^{11}\|}\right) = \cos^{-1}\left(\frac{L_1^2-L_2^2+L_3^2}{L_1^2+L_2^2+L_3^2}\right).
\end{aligned} \tag{3.21}$$

Finally, the unit normal vectors $\mathbf{n}^{11}, \ldots, \mathbf{n}^{14}$ are determined using (3.19) as follows:

$$\begin{aligned}
\mathbf{n}^{11} &= \frac{\mathbf{z}^{11}\times\mathbf{z}^{12}}{\|\mathbf{z}^{11}\times\mathbf{z}^{12}\|} = \frac{1}{(L_2^2+L_3^2)^{\frac{1}{2}}}\begin{bmatrix} 0 \\ L_3 \\ L_2 \end{bmatrix}, \\
\mathbf{n}^{12} &= \frac{\mathbf{z}^{12}\times\mathbf{z}^{13}}{\|\mathbf{z}^{12}\times\mathbf{z}^{13}\|} = \frac{1}{(L_1^2+L_3^2)^{\frac{1}{2}}}\begin{bmatrix} -L_3 \\ 0 \\ L_1 \end{bmatrix}, \\
\mathbf{n}^{13} &= \frac{\mathbf{z}^{13}\times\mathbf{z}^{14}}{\|\mathbf{z}^{13}\times\mathbf{z}^{14}\|} = \frac{1}{(L_2^2+L_3^2)^{\frac{1}{2}}}\begin{bmatrix} 0 \\ -L_3 \\ L_2 \end{bmatrix}, \\
\mathbf{n}^{14} &= \frac{\mathbf{z}^{14}\times\mathbf{z}^{11}}{\|\mathbf{z}^{14}\times\mathbf{z}^{11}\|} = \frac{1}{(L_1^2+L_3^2)^{\frac{1}{2}}}\begin{bmatrix} L_3 \\ 0 \\ L_1 \end{bmatrix}.
\end{aligned} \tag{3.22}$$

Table 3.2 Calculated parameters required to define a goal mesh $\mathcal{M}$

Parameter	Equation
Position vectors of the nodes of each mesh face $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}}$, $j = 1, \ldots, N_{\mathcal{M}}$	(3.4)
Unit normal vector $\tilde{\mathbf{n}}^j$ of each mesh face, $j = 1, \ldots, N_{\mathcal{M}}$	(3.5)
Edge connectivity matrix $\mathbf{C}^{\mathcal{E}_I}$	(3.14)
Vectors along the length of each interior edge $\hat{\mathbf{z}}^1, \ldots, \hat{\mathbf{z}}^{N_{\mathcal{E}}^I}$	(3.15)
Node connectivity matrix $\mathbf{C}^{\mathcal{N}_I}$	(3.16)
Vectors along the edges connecting the jth interior node to its adjacent nodes $\mathbf{z}^{j1}, \ldots, \mathbf{z}^{jn_j^{\mathcal{N}}}$, $j = 1, \ldots, N_{\mathcal{N}}^I$	(3.17)
Corner angles of the faces connected to the jth interior node $\phi_{j1}, \ldots, \phi_{jn_j^{\mathcal{N}}}$, $j = 1, \ldots, N_{\mathcal{N}}^I$	(3.18)
Unit normal vectors of the faces connected to the jth interior node $\mathbf{n}^{j1}, \ldots, \mathbf{n}^{jn_j^{\mathcal{N}}}$, $j = 1, \ldots, N_{\mathcal{N}}^I$	(3.19)

Such parameters are determined from those provided in Table 3.1

The calculated parameters required to define a goal mesh $\mathcal{M}$ are summarized in Table 3.2.

3.2.3 *Determination of Spanning Trees*

This section describes a procedure to determine a *spanning tree* for a given goal mesh $\mathcal{M}$, which is the first step in the implementation of the unfolding polyhedra method (refer to Sect. 3.2.1). A spanning tree is a line graph on $\mathcal{M}$ that contains a reference point in each face of $\mathcal{M}$. Such a graph is called a "tree" because it does not have any loops. We illustrate a spanning tree in Fig. 3.3b. The spanning tree does not cross the nodes of $\mathcal{M}$ and does not cross any edge of $\mathcal{M}$ more than once. The significance of the spanning tree is not its particular geometry but is rather the set of edges of $\mathcal{M}$ that it crosses.

The *original boundary* of a given goal mesh $\mathcal{M}$ is denoted $\partial\mathcal{M}^O$ (Fig. 3.10b). In general, the goal mesh $\mathcal{M}$ with boundary $\partial\mathcal{M}^O$ is not a tree of faces (i.e., the faces of $\mathcal{M}$ with boundary $\partial\mathcal{M}^O$ may form loops; see Fig. 3.10b). In the unfolding polyhedra method, we require the goal mesh to have the topology of a tree of faces in order to flatten it and form a net (see Fig. 3.3). In view of this, we introduce additional boundary edges forming the *cut boundary* $\partial\mathcal{M}^C$ (Fig. 3.10c) such that the goal mesh becomes a tree of faces having boundary $\partial\mathcal{M} = \partial\mathcal{M}^O \cup \partial\mathcal{M}^C$ and can be flattened onto a plane to form a net (Fig. 3.10d).

We use a *spanning tree* to compute the cut boundary $\partial\mathcal{M}^C$. Once a spanning tree is determined for $\mathcal{M}$, the edges that are *not* crossed by the spanning tree are assigned as the edges forming the cut boundary $\partial\mathcal{M}^C$ (see Fig. 3.11). The resulting mesh having boundary $\partial\mathcal{M} = \partial\mathcal{M}^O \cup \partial\mathcal{M}^C$ corresponds to a tree of faces with the same topology of the spanning tree.

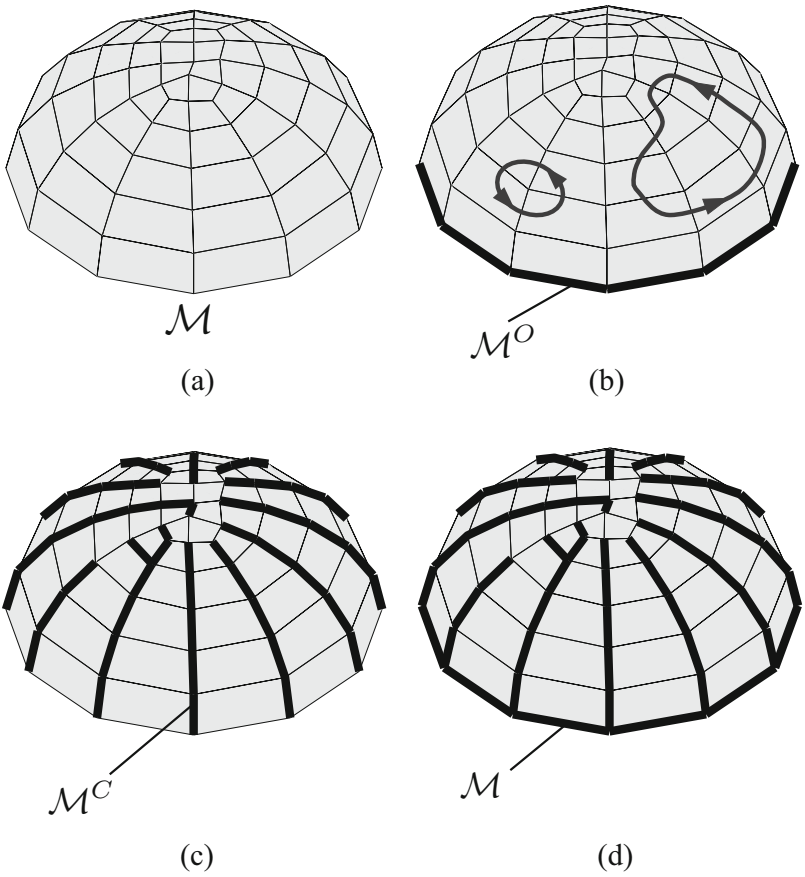


Fig. 3.10 (**a**) Goal mesh $\mathcal{M}$ of a dome. The base of the dome is open. (**b**) Original boundary of the given goal mesh $\partial\mathcal{M}^O$. Two arbitrary paths are shown on the mesh to indicate loops of faces. (**c**) Edges forming the cut boundary $\partial\mathcal{M}^C$ that is introduced to convert $\mathcal{M}$ into a tree of faces. (**d**) Resulting mesh corresponding to a tree of faces having boundary $\partial\mathcal{M} = \partial\mathcal{M}^O \cup \partial\mathcal{M}^C$

In general, the spanning tree of a given goal mesh $\mathcal{M}$ is not unique as illustrated in Fig. 3.11. Schlickenrieder proposed several heuristic procedures to determine spanning trees in polyhedra [21]. Here, we employ the following procedure to determine a spanning tree for a given goal mesh $\mathcal{M}$:

1. An arbitrary face in $\mathcal{M}$ is selected as the *reference face* (Fig. 3.12a). A point in this face becomes the start point of the spanning tree
2. The faces sharing an edge with the reference face are then integrated to the spanning tree (Step 1; see Fig. 3.12b). At Step $i + 1$, the faces sharing an edge with those included in the spanning tree at Step i but have not yet been included to the spanning tree are then integrated to it. The steps continue until all the faces in $\mathcal{M}$ have been incorporated to the spanning tree (Fig. 3.12c–e)

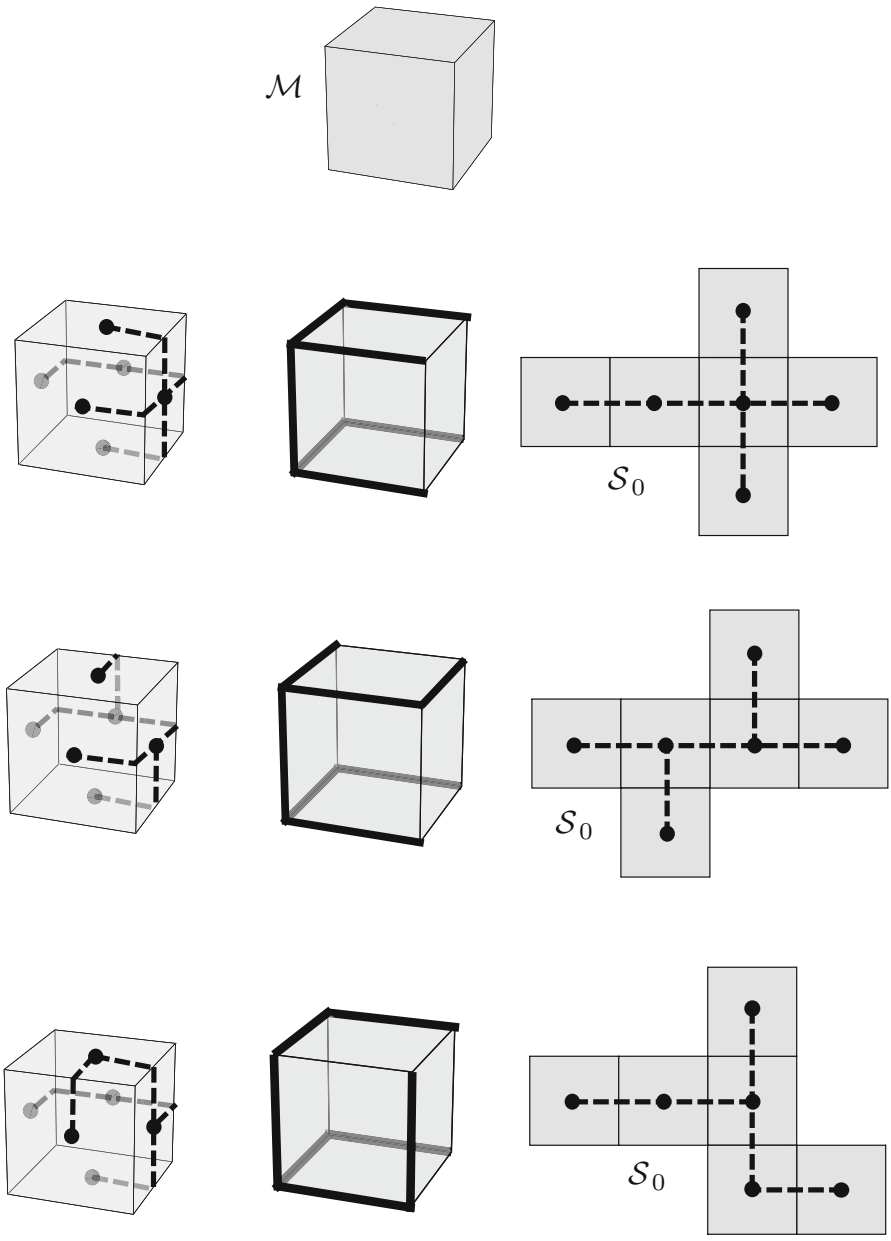


Fig. 3.11 Goal mesh $\mathcal{M}$ of a cube and different boundary edges and nets determined from three different spanning trees

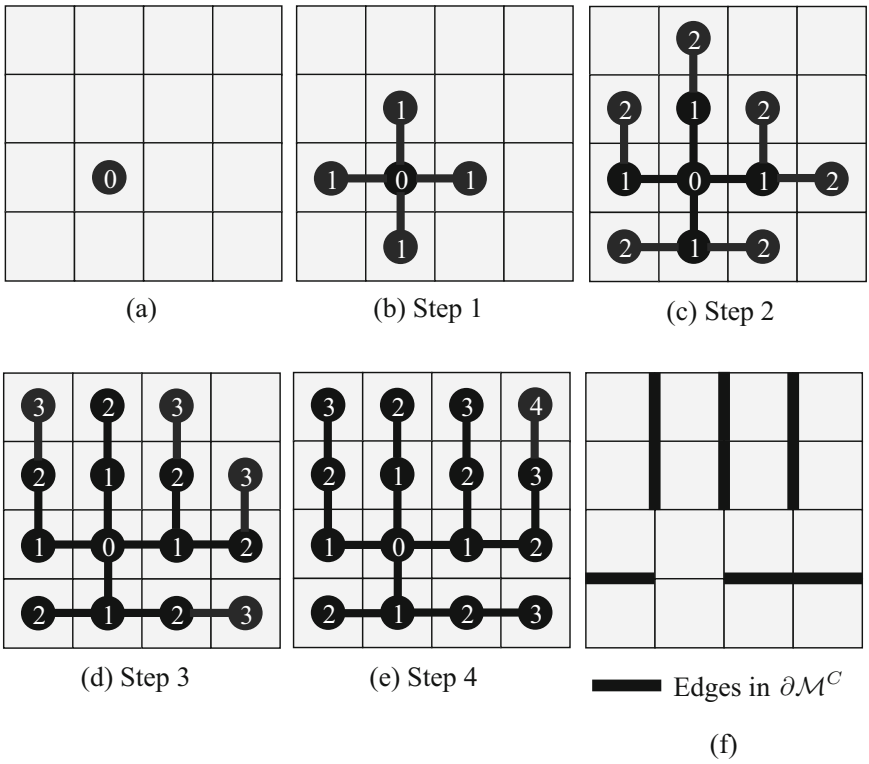


Fig. 3.12 (**a**)–(**e**) Schematic of the procedure to determine a spanning tree for a given goal mesh. The reference face is marked with a circle enclosing a 0. At each step, the neighboring faces that have not been incorporated to the spanning tree are then integrated to it. The numbers enclosed in the circles correspond to the step number at which each face is incorporated to the spanning tree. (**f**) Introduced boundary edges forming the cut boundary $\partial\mathcal{M}^C$ corresponding to the edges that are not crossed by the spanning tree

The spanning tree determined using the described procedure is obviously dependent on the face selected as the reference face. This leads to different possible nets for a given goal mesh $\mathcal{M}$. To illustrate this, Fig. 3.13 shows two different nets for a truncated cube, one determined by selecting a triangle face as the reference face and another by selecting an octagon face.

A net obtained by mapping a determined tree of faces onto a plane has obviously the same number of faces of $\mathcal{M}$. Thus, the number of faces of an origami sheet ($N_{\mathcal{P}}$; refer to Sect. 2.2) obtained via the unfolding polyhedra method is equal to the number of faces of the goal mesh ($N_{\mathcal{M}}$). Furthermore, the number of folds in such an origami sheet ($N_{\mathcal{F}}$) corresponds to the number of edges in the tree of faces which

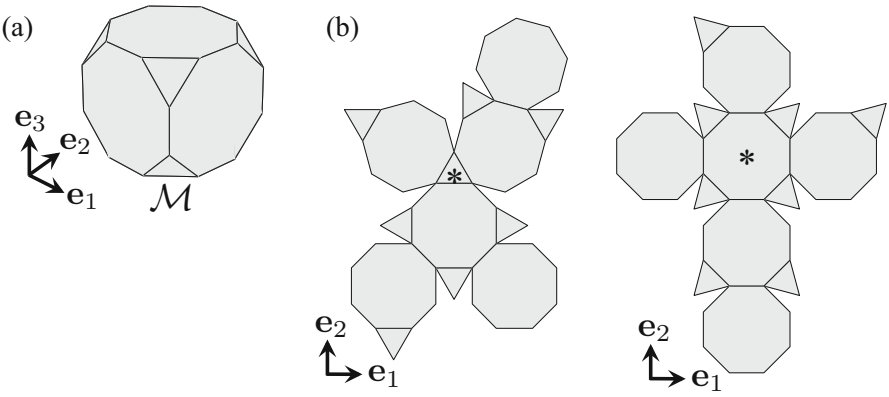


Fig. 3.13 (**a**) Goal mesh $\mathcal{M}$ of a truncated cube. (**b**) Two different nets for the truncated cube. The reference faces for the spanning tree of each net are marked with the symbol $*$. The number of faces of the goal mesh ($N_{\mathcal{M}}$) is 14 (6 octagons and 8 triangles). As indicated in (3.23), the number of folds ($N_{\mathcal{F}}$) for both nets is $N_{\mathcal{M}} - 1 = 13$

is equal to $N_{\mathcal{M}} - 1$. This is because one edge is crossed by the spanning tree to reach each of the $N_{\mathcal{M}}$ faces of $\mathcal{M}$, except for the reference face. In summary:

$$N_{\mathcal{P}} = N_{\mathcal{M}}, \qquad N_{\mathcal{F}} = N_{\mathcal{M}} - 1. \tag{3.23}$$

As stated in Sect. 3.2.1, after determining a spanning tree and a cut boundary $\partial\mathcal{M}^C$ that converts $\mathcal{M}$ to a tree of faces, the next step is to map such a tree of faces onto a plane to form a net. We summarize such a mapping in the following section.

3.2.4 Formulation of the Unfolding Map

As stated in Sect. 3.2.1, after determining a spanning tree and a cut boundary that converts the goal mesh $\mathcal{M}$ to a tree of faces (see Sect. 3.2.3), the next step in the unfolding polyhedra method is to map the tree of faces onto a plane to form a net. We denote this mapping as the *unfolding map* and we summarize it in this section. We formulate the unfolding map by assuming that the complete net is placed on the 1-2 plane. The mapping of each mesh face $\mathcal{M}^j$, $j = 1, \ldots, N_{\mathcal{M}}$, towards its position in the net is performed through the following three steps:

1. The face $\mathcal{M}^j$ is rotated to align its unit normal vector with $\mathbf{e}_3$ (Fig. 3.14b). After this first step is done, the updated configuration of the mesh face is denoted $\mathcal{M}^j_{(1)}$ and the position vectors of its nodes are denoted $\tilde{\mathbf{y}}^{jk}_{(1)}$, $k = 1, \ldots, n^{\mathcal{M}}_j$
2. The face $\mathcal{M}^j_{(1)}$ is translated to place one of its nodes at the origin and rotated to align one of its edges with $\mathbf{e}_1$ (Fig. 3.14c). The face is fully contained in the

1-2 plane after this step is done. The updated configuration of the face is denoted $\mathcal{M}^j_{(2)}$ and the position vectors of its nodes are denoted $\tilde{\mathbf{y}}^{jk}_{(2)}$, $k = 1, \ldots, n^{\mathcal{M}}_j$

3. The face $\mathcal{M}^j_{(2)}$ is rotated and translated within the 1-2 plane towards its corresponding position in the net (Fig. 3.14d). After this final step is done, the updated configuration of the face is denoted $\mathcal{M}^j_{(3)}$, *which corresponds to the configuration of the face in the net*, and the position vectors of its nodes are denoted $\tilde{\mathbf{y}}^{jk}_{(3)}$, $k = 1, \ldots, n^{\mathcal{M}}_j$

To determine the parameters needed to perform these steps, we define the paths $\tilde{\boldsymbol{\gamma}}^1(\eta), \ldots, \tilde{\boldsymbol{\gamma}}^{N_{\mathcal{M}}}(\eta) : [0, 1] \to \mathcal{M}$ that connect the *reference face*[2] to each face $\mathcal{M}^1, \ldots, \mathcal{M}^{N_{\mathcal{M}}}$ in the tree of faces (Fig. 3.15a). Each path $\tilde{\boldsymbol{\gamma}}^j(\eta)$ crosses a total of $\tilde{n}_j$ faces from the reference face until reaching $\mathcal{M}^j$, including the reference face and $\mathcal{M}^j$. For the path $\tilde{\boldsymbol{\gamma}}^j(\eta)$ illustrated in Fig. 3.15a, $\tilde{n}_j = 4$. The unit normal vector of each face in $\mathcal{M}$ crossed by $\tilde{\boldsymbol{\gamma}}^j(\eta)$ is denoted $\tilde{\mathbf{n}}^{jk}_{\gamma} \in \mathbb{R}^3$, $k = 1, \ldots, \tilde{n}_j$.

As shown in Fig. 3.15b, the vector along the edge where $\tilde{\boldsymbol{\gamma}}^j(\eta)$ *enters* the kth face is denoted $\tilde{\mathbf{z}}^{jk}_{\mathrm{I}} \in \mathbb{R}^3$, $k = 1, \ldots, \tilde{n}_j$, and the position vector of the start point of this edge is denoted $\tilde{\mathbf{y}}^{jk}_{\mathrm{I}} \in \mathbb{R}^3$. Likewise, the vector along the edge where $\tilde{\boldsymbol{\gamma}}^j(\eta)$ *exits* the kth face is denoted $\tilde{\mathbf{z}}^{jk}_{\mathrm{O}} \in \mathbb{R}^3$ and the position vector of the start point of this edge is denoted $\tilde{\mathbf{y}}^{jk}_{\mathrm{O}} \in \mathbb{R}^3$. To define $\tilde{\mathbf{z}}^{j1}_{\mathrm{I}}$ and $\tilde{\mathbf{y}}^{j1}_{\mathrm{I}}$, $j = 1, \ldots, N_{\mathcal{M}}$, an arbitrary edge of the reference face (i.e., the *reference edge*; see Fig. 3.15c) is selected and $\tilde{\mathbf{z}}^{j1}_{\mathrm{I}}$ is the vector along that edge while $\tilde{\mathbf{y}}^{j1}_{\mathrm{I}}$ is the position vector of the start point of that edge.

Step 1: Rotating $\mathcal{M}^j$ to align its unit normal vector with $\mathbf{e}_3$

We perform the first step of the unfolding map through a single rotation that aligns the unit normal vector of $\mathcal{M}^j$ with $\mathbf{e}_3$ (see Fig. 3.14b). A rotation of vector $\tilde{\mathbf{y}}^{jk}$ by ϕ radians about an axis aligned with a unit vector $\mathbf{b} \in \mathbb{R}^3$ can be expressed as follows [35, 36]:

$$\tilde{\mathbf{y}}^{jk}_{(1)} = \tilde{\mathbf{y}}^{jk}\cos(\phi) + (\mathbf{b} \times \tilde{\mathbf{y}}^{jk})\sin(\phi) + \mathbf{b}(\mathbf{b} \cdot \tilde{\mathbf{y}}^{jk})(1 - \cos(\phi)). \tag{3.24}$$

We use such a formula to define a rotation that aligns the unit normal vector of $\mathcal{M}^j$ with $\mathbf{e}_3$. As illustrated in Fig. 3.16, this can be achieved through a rotation of $\cos^{-1}(\tilde{\mathbf{n}}^j \cdot \mathbf{e}_3)$ about an axis aligned with the vector $\tilde{\mathbf{n}}^j \times \mathbf{e}_3$. Accordingly, the following expressions are substituted in (3.24) to determine the rotation that aligns the unit normal vector of $\mathcal{M}^j$ with $\mathbf{e}_3$:

$$\cos(\phi) = \frac{\tilde{\mathbf{n}}^j \cdot \mathbf{e}_3}{\|\tilde{\mathbf{n}}^j\| \, \|\mathbf{e}_3\|} = \frac{\tilde{\mathbf{n}}^j \cdot \mathbf{e}_3}{(1)(1)} = \tilde{\mathbf{n}}^j \cdot \mathbf{e}_3, \tag{3.25}$$

[2] Consult Sect. 3.2.3 for the definition of the reference face.

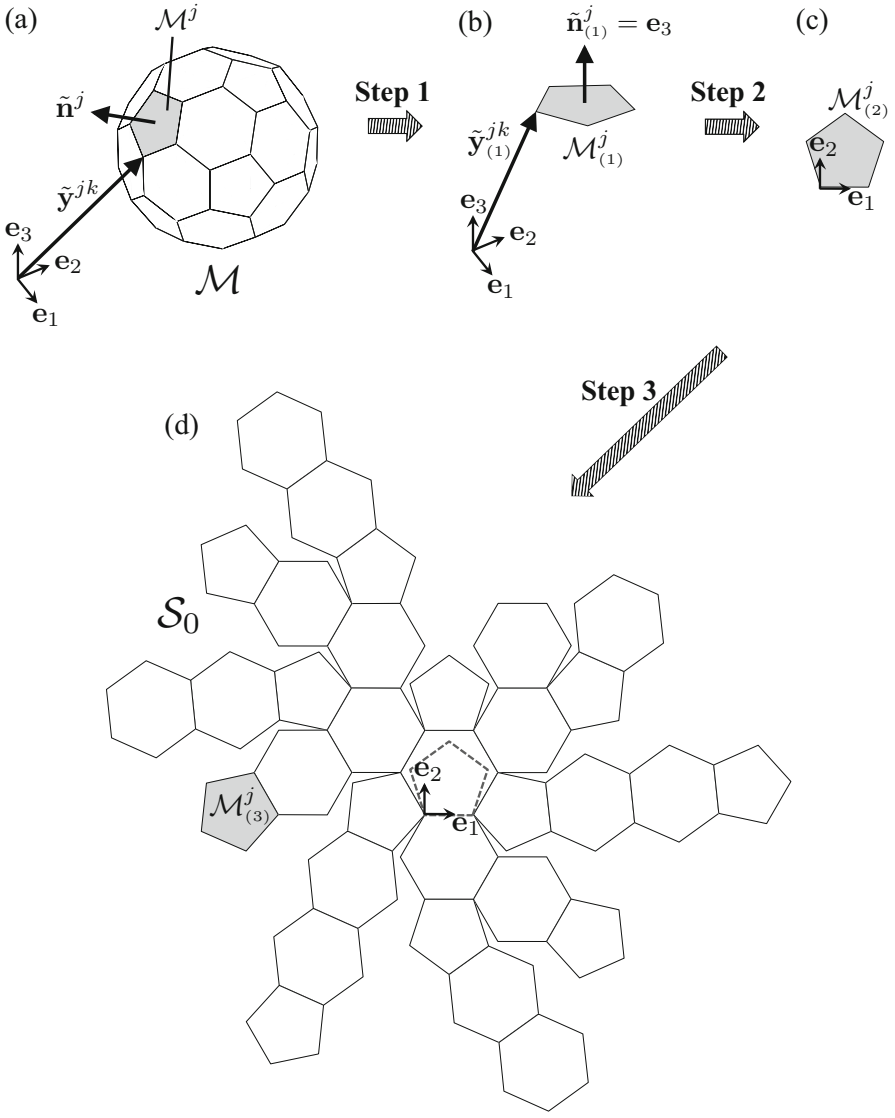


Fig. 3.14 Steps taken in the mapping of mesh face $\mathcal{M}^j$ from its position in the goal mesh $\mathcal{M}$ to its position in the net: (**a**) Configuration of $\mathcal{M}^j$ within $\mathcal{M}$; (**b**) The face is rotated to align its unit normal vector with $\mathbf{e}_3$; (**c**) The face is translated to place one of its nodes at the origin and rotated to align one of its edges with $\mathbf{e}_1$; (**d**) The face is then rotated and translated within the 1-2 plane towards its corresponding position in the net

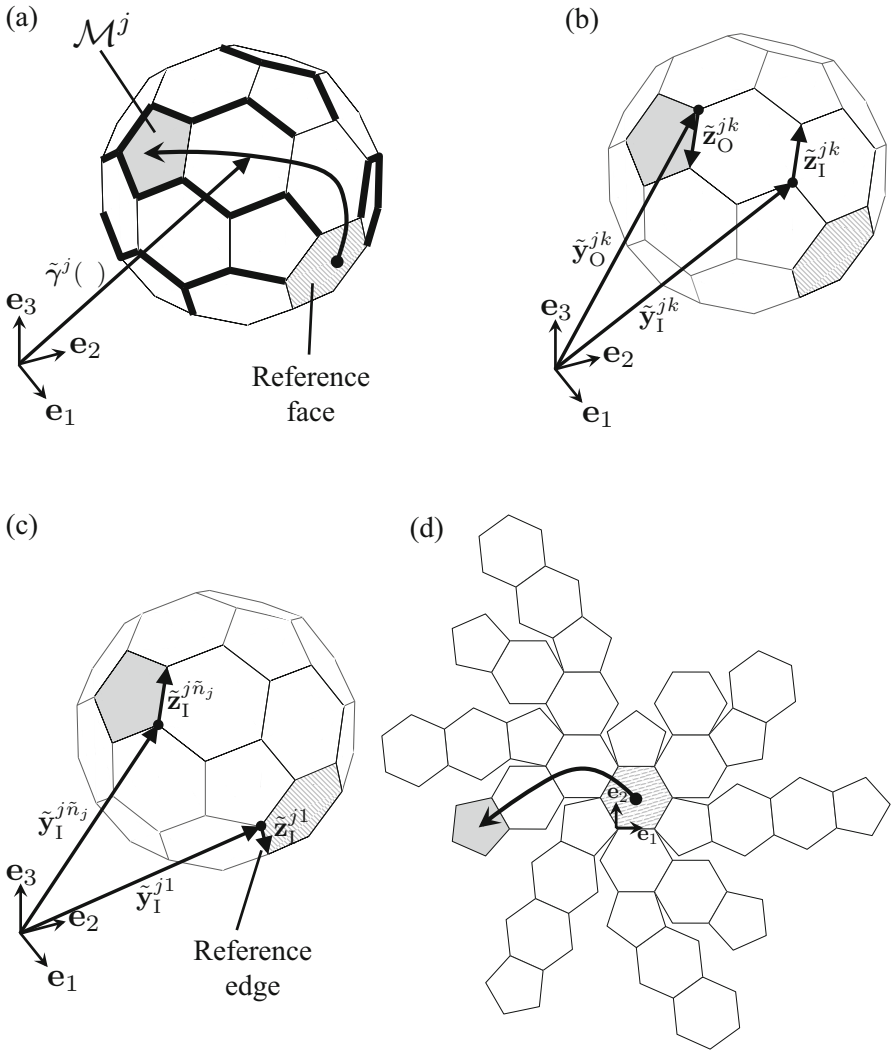


Fig. 3.15 (**a**) Path $\tilde{\boldsymbol{\gamma}}^j(\eta)$ connecting the reference face with $\mathcal{M}^j$. (**b**) Vectors $\tilde{\mathbf{z}}_{\mathrm{I}}^{jk}$ and $\tilde{\mathbf{z}}_{\mathrm{O}}^{jk}$ along the edges where $\tilde{\boldsymbol{\gamma}}^j(\eta)$, respectively, enters and exits the kth face that it crosses. (**c**) Vector $\tilde{\mathbf{z}}_{\mathrm{I}}^{j1}$ along the reference edge and vector $\tilde{\mathbf{z}}_{\mathrm{I}}^{j\tilde{n}_j}$ along the edge where $\tilde{\boldsymbol{\gamma}}^j(\eta)$ enters $\mathcal{M}^j$. (**d**) Face $\mathcal{M}^j$ mapped to its position at the net

$$\sin(\phi) \;=\; \frac{\|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|}{\|\tilde{\mathbf{n}}^j\|\,\|\mathbf{e}_3\|} \;=\; \frac{\|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|}{(1)(1)} \;=\; \|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|, \tag{3.26}$$

$$\mathbf{b} = \frac{\tilde{\mathbf{n}}^j \times \mathbf{e}_3}{\|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|}. \tag{3.27}$$

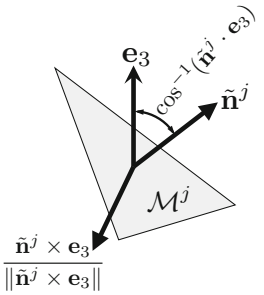


Fig. 3.16 Mesh face $\mathcal{M}^j$ with unit normal vector $\tilde{\mathbf{n}}^j$

The following is obtained by substituting (3.27) into (3.24):

$$\begin{aligned}\tilde{\mathbf{y}}_{(1)}^{jk} = \tilde{\mathbf{y}}^{jk}\cos(\phi) + \left(\frac{\tilde{\mathbf{n}}^j \times \mathbf{e}_3}{\|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|} \times \tilde{\mathbf{y}}^{jk}\right)\sin(\phi) \\ + \frac{\tilde{\mathbf{n}}^j \times \mathbf{e}_3}{\|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|}\left(\frac{\tilde{\mathbf{n}}^j \times \mathbf{e}_3}{\|\tilde{\mathbf{n}}^j \times \mathbf{e}_3\|} \cdot \tilde{\mathbf{y}}^{jk}\right)(1 - \cos(\phi)),\end{aligned} \tag{3.28}$$

which can be further simplified by substitution of (3.26) into (3.28):

$$\begin{aligned}\tilde{\mathbf{y}}_{(1)}^{jk} &= \tilde{\mathbf{y}}^{jk}\cos(\phi) + \left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \times \tilde{\mathbf{y}}^{jk} \\ &\quad + \left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right)\left(\left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \cdot \tilde{\mathbf{y}}^{jk}\right)\frac{1 - \cos(\phi)}{\sin^2(\phi)} \\ &= \tilde{\mathbf{y}}^{jk}\cos(\phi) + \left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \times \tilde{\mathbf{y}}^{jk} \\ &\quad + \left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right)\left(\left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \cdot \tilde{\mathbf{y}}^{jk}\right)\frac{1}{1 + \cos(\phi)}.\end{aligned} \tag{3.29}$$

By substituting (3.25) into (3.29), the following is determined:

$$\tilde{\mathbf{y}}_{(1)}^{jk} = \tilde{\mathbf{y}}^{jk}\left(\tilde{\mathbf{n}}^j \cdot \mathbf{e}_3\right) + \left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \times \tilde{\mathbf{y}}^{jk} + \frac{\tilde{\mathbf{n}}^j \times \mathbf{e}_3}{1 + \tilde{\mathbf{n}}^j \cdot \mathbf{e}_3}\left(\left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \cdot \tilde{\mathbf{y}}^{jk}\right). \tag{3.30}$$

It is noted that the rotation formula above is not applicable when $\tilde{\mathbf{n}}^j = -\mathbf{e}_3$. In that case, a rotation of π radians about an axis aligned to $\mathbf{e}_1$ is applied to align the unit normal vector of $\mathcal{M}^j$ with $\mathbf{e}_3$. Accordingly, the equation used to transform the position vectors of the nodes of the mesh faces $\mathcal{M}^j$, $j = 1, \ldots, N_{\mathcal{M}}$, to align their unit normal vector with $\mathbf{e}_3$ is then given as follows:

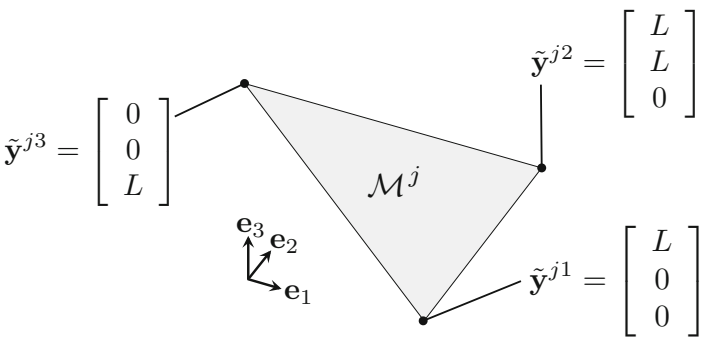


Fig. 3.17 Schematic for Example 3.5: Face $\mathcal{M}^j$ and the position vectors of its nodes

$$\tilde{\mathbf{y}}^{jk}_{(1)} = \tilde{\chi}(\tilde{\mathbf{y}}^{jk})$$

where:

$$\tilde{\chi}(\tilde{\mathbf{y}}^{jk}) = \begin{cases} \mathbf{R}_1(\pi)\,\tilde{\mathbf{y}}^{jk}; & \tilde{\mathbf{n}}^j = -\mathbf{e}_3 \\ \tilde{\mathbf{y}}^{jk}\left(\tilde{\mathbf{n}}^j \cdot \mathbf{e}_3\right) + \left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \times \tilde{\mathbf{y}}^{jk} \\ \quad + \dfrac{\tilde{\mathbf{n}}^j \times \mathbf{e}_3}{1 + \tilde{\mathbf{n}}^j \cdot \mathbf{e}_3}\left(\left(\tilde{\mathbf{n}}^j \times \mathbf{e}_3\right) \cdot \tilde{\mathbf{y}}^{jk}\right); & \tilde{\mathbf{n}}^j \neq -\mathbf{e}_3, \end{cases} \tag{3.31}$$

where $\mathbf{R}_1(\pi)$ is the matrix associated with a rotation of π radians about an axis aligned with $\mathbf{e}_1$; refer to (2.21). The unit normal vectors $\tilde{\mathbf{n}}^j_{(1)}$, $j = 1, \ldots, N_{\mathcal{M}}$, of the faces $\mathcal{M}^j_{(1)}$ having nodes with position vectors $\tilde{\mathbf{y}}^{j1}_{(1)}, \ldots, \tilde{\mathbf{y}}^{jn^{\mathcal{M}}_j}_{(1)}$ are equal to $\mathbf{e}_3$ (see Fig. 3.14b):

$$\tilde{\mathbf{n}}^j_{(1)} = \frac{(\tilde{\mathbf{y}}^{j2}_{(1)} - \tilde{\mathbf{y}}^{j1}_{(1)}) \times (\tilde{\mathbf{y}}^{j3}_{(1)} - \tilde{\mathbf{y}}^{j2}_{(1)})}{\|(\tilde{\mathbf{y}}^{j2}_{(1)} - \tilde{\mathbf{y}}^{j1}_{(1)}) \times (\tilde{\mathbf{y}}^{j3}_{(1)} - \tilde{\mathbf{y}}^{j2}_{(1)})\|} = \mathbf{e}_3. \tag{3.32}$$

Example 3.5 Rotating a mesh face to align its unit normal vector with $\mathbf{e}_3$.

Statement: Consider the mesh face $\mathcal{M}^j$ illustrated in Fig. 3.17. (a) Determine the rotation in the form of (3.31) that is required to align the unit normal vector of $\mathcal{M}^j$ with $\mathbf{e}_3$. (b) Determine the position vectors $\tilde{\mathbf{y}}^{j1}_{(1)}$, $\tilde{\mathbf{y}}^{j2}_{(1)}$, and $\tilde{\mathbf{y}}^{j3}_{(1)}$. (c) Verify that $\tilde{\mathbf{n}}^j_{(1)} = \mathbf{e}_3$.

Solution:

(a) To determine the rotation in the form of (3.31), the unit normal vector of $\mathcal{M}^j$ (i.e., $\tilde{\mathbf{n}}^j$) must be determined first. Using (3.5), $\tilde{\mathbf{n}}^j$ is calculated as follows:

$$\begin{aligned}\tilde{\mathbf{n}}^j &= \frac{(\tilde{\mathbf{y}}^{j2}-\tilde{\mathbf{y}}^{j1})\times(\tilde{\mathbf{y}}^{j3}-\tilde{\mathbf{y}}^{j2})}{\|(\tilde{\mathbf{y}}^{j2}-\tilde{\mathbf{y}}^{j1})\times(\tilde{\mathbf{y}}^{j3}-\tilde{\mathbf{y}}^{j2})\|} \\ &= \frac{2^{-\frac{1}{2}}}{L^2}\begin{bmatrix}L^2\\0\\L^2\end{bmatrix} = \begin{bmatrix}2^{-\frac{1}{2}}\\0\\2^{-\frac{1}{2}}\end{bmatrix},\end{aligned} \tag{3.33}$$

and therefore:

$$\tilde{\mathbf{n}}^j\cdot\mathbf{e}_3 = 2^{-\frac{1}{2}}, \qquad \tilde{\mathbf{n}}^j\times\mathbf{e}_3 = \begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}. \tag{3.34}$$

By substituting (3.33) and (3.34) into (3.31), the following transformation is obtained:

$$\tilde{\mathbf{y}}^{jk}_{(1)} = 2^{-\frac{1}{2}}\,\tilde{\mathbf{y}}^{jk} + \begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\times\tilde{\mathbf{y}}^{jk} + \frac{1}{1+2^{-\frac{1}{2}}}\begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\left(\begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\cdot\tilde{\mathbf{y}}^{jk}\right). \tag{3.35}$$

(b) The position vectors of the nodes of the rotated face ($\tilde{\mathbf{y}}^{j1}_{(1)}$, $\tilde{\mathbf{y}}^{j2}_{(1)}$, and $\tilde{\mathbf{y}}^{j3}_{(1)}$) are determined by, respectively, substituting $\tilde{\mathbf{y}}^{j1}$, $\tilde{\mathbf{y}}^{j2}$, and $\tilde{\mathbf{y}}^{j3}$ into (3.35):

$$\begin{aligned}\tilde{\mathbf{y}}^{j1}_{(1)} &= 2^{-\frac{1}{2}}\begin{bmatrix}L\\0\\0\end{bmatrix} + \begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\times\begin{bmatrix}L\\0\\0\end{bmatrix} \\ &\quad + \frac{1}{1+2^{-\frac{1}{2}}}\begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\left(\begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\cdot\begin{bmatrix}L\\0\\0\end{bmatrix}\right) = \begin{bmatrix}2^{-\frac{1}{2}}L\\0\\2^{-\frac{1}{2}}L\end{bmatrix},\end{aligned} \tag{3.36}$$

$$\begin{aligned}\tilde{\mathbf{y}}^{j2}_{(1)} &= 2^{-\frac{1}{2}}\begin{bmatrix}L\\L\\0\end{bmatrix} + \begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\times\begin{bmatrix}L\\L\\0\end{bmatrix} \\ &\quad + \frac{1}{1+2^{-\frac{1}{2}}}\begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\left(\begin{bmatrix}0\\-2^{-\frac{1}{2}}\\0\end{bmatrix}\cdot\begin{bmatrix}L\\L\\0\end{bmatrix}\right) = \begin{bmatrix}2^{-\frac{1}{2}}L\\L\\2^{-\frac{1}{2}}L\end{bmatrix},\end{aligned} \tag{3.37}$$

$$\tilde{\mathbf{y}}_{(1)}^{j3} = 2^{-\frac{1}{2}} \begin{bmatrix} 0 \\ 0 \\ L \end{bmatrix} + \begin{bmatrix} 0 \\ -2^{-\frac{1}{2}} \\ 0 \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ L \end{bmatrix} + \frac{1}{1+2^{-\frac{1}{2}}} \begin{bmatrix} 0 \\ -2^{-\frac{1}{2}} \\ 0 \end{bmatrix} \left(\left(\begin{bmatrix} 0 \\ -2^{-\frac{1}{2}} \\ 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ L \end{bmatrix} \right) = \begin{bmatrix} -2^{-\frac{1}{2}} L \\ 0 \\ 2^{-\frac{1}{2}} L \end{bmatrix} . \tag{3.38}$$

(c) The unit normal vector of the rotated face ($\tilde{\mathbf{n}}_{(1)}^{j}$) is calculated using (3.32) in order to verify if it is aligned with $\mathbf{e}_3$:

$$\begin{aligned} \tilde{\mathbf{n}}_{(1)}^{j} &= \frac{(\tilde{\mathbf{y}}_{(1)}^{j2} - \tilde{\mathbf{y}}_{(1)}^{j1}) \times (\tilde{\mathbf{y}}_{(1)}^{j3} - \tilde{\mathbf{y}}_{(1)}^{j2})}{\| (\tilde{\mathbf{y}}_{(1)}^{j2} - \tilde{\mathbf{y}}_{(1)}^{j1}) \times (\tilde{\mathbf{y}}_{(1)}^{j3} - \tilde{\mathbf{y}}_{(1)}^{j2}) \|} \\ &= \frac{2^{-\frac{1}{2}}}{L^2} \begin{bmatrix} 0 \\ 0 \\ 2^{\frac{1}{2}} L^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \mathbf{e}_3. \end{aligned} \tag{3.39}$$

Step 2: Translation of $\mathcal{M}_{(1)}^{j}$ to place one of its nodes at the origin and rotation to align one of its edges with $\mathbf{e}_1$

After determining the configurations of the faces $\mathcal{M}_{(1)}^{1}, \ldots, \mathcal{M}_{(1)}^{N_{\mathcal{M}}}$ having their unit normal vectors aligned with $\mathbf{e}_3$, these faces are translated to place one of their nodes at the origin and rotated to align one of their edges with $\mathbf{e}_1$ (Fig. 3.14c). These transformations place the mesh faces fully in the 1-2 plane.

To determine the parameters required to apply these transformations, let the vectors $\tilde{\mathbf{z}}_{\mathrm{I}(1)}^{jk}$, $\tilde{\mathbf{z}}_{\mathrm{O}(1)}^{jk}$, $\tilde{\mathbf{y}}_{\mathrm{I}(1)}^{jk}$, and $\tilde{\mathbf{y}}_{\mathrm{O}(1)}^{jk}$, respectively, correspond to $\tilde{\mathbf{z}}_{\mathrm{I}}^{jk}$, $\tilde{\mathbf{z}}_{\mathrm{O}}^{jk}$, $\tilde{\mathbf{y}}_{\mathrm{I}}^{jk}$, and $\tilde{\mathbf{y}}_{\mathrm{O}}^{jk}$ after undergoing the rotation transformation (3.31):

$$\begin{aligned} \tilde{\mathbf{z}}_{\mathrm{I}(1)}^{jk} &= \tilde{\chi}(\tilde{\mathbf{z}}_{\mathrm{I}}^{jk}), & \tilde{\mathbf{z}}_{\mathrm{O}(1)}^{jk} &= \tilde{\chi}(\tilde{\mathbf{z}}_{\mathrm{O}}^{jk}), \\ \tilde{\mathbf{y}}_{\mathrm{I}(1)}^{jk} &= \tilde{\chi}(\tilde{\mathbf{y}}_{\mathrm{I}}^{jk}), & \tilde{\mathbf{y}}_{\mathrm{O}(1)}^{jk} &= \tilde{\chi}(\tilde{\mathbf{y}}_{\mathrm{O}}^{jk}). \end{aligned} \tag{3.40}$$

The face $\mathcal{M}_{(1)}^{j}$ is translated such that its node with position vector $\tilde{\mathbf{y}}_{\mathrm{I}(1)}^{j\tilde{n}_j}$ is placed at the origin (Fig. 3.18b). This translation is formulated in matrix form as $\mathbf{T}^{-1}(\tilde{\mathbf{y}}_{\mathrm{I}(1)}^{j\tilde{n}_j})$, where $\mathbf{T}(\cdot)$ is defined in (2.53). Next, the face is rotated such that the edge along $\tilde{\mathbf{z}}_{\mathrm{I}(1)}^{j\tilde{n}_j}$ is aligned with $\mathbf{e}_1$ (Fig. 3.18c). This is accomplished through a rotation of $-\varphi(\tilde{\mathbf{z}}_{\mathrm{I}(1)}^{j\tilde{n}_j})$ about an axis aligned with $\mathbf{e}_3$, where $\varphi(\tilde{\mathbf{z}}_{\mathrm{I}(1)}^{j\tilde{n}_j})$ is the angle between $\mathbf{e}_1$ and $\tilde{\mathbf{z}}_{\mathrm{I}(1)}^{j\tilde{n}_j}$ calculated using (2.13). Such a transformation is formulated via the rotation matrix $\mathbf{Q}_3^{-1}(\varphi(\tilde{\mathbf{z}}_{\mathrm{I}(1)}^{j\tilde{n}_j}))$, where $\mathbf{Q}_3(\cdot)$ is defined in (2.57).

In summary, the position vectors of the nodes of $\mathcal{M}^j_{(2)}$, which corresponds to the configuration of face $\mathcal{M}^j$ placed on the 1-2 plane and having one node at the origin and one edge aligned with $\mathbf{e}_1$, are determined as follows:

$$\boxed{\begin{bmatrix} \tilde{\mathbf{y}}^{jk}_{(2)} \\ 1 \end{bmatrix} = \mathbf{Q}_3^{-1}(\varphi(\tilde{\mathbf{z}}^{j\tilde{n}_j}_{\mathrm{I}(1)}))\, \mathbf{T}^{-1}(\tilde{\mathbf{y}}^{j\tilde{n}_j}_{\mathrm{I}(1)}) \begin{bmatrix} \tilde{\mathbf{y}}^{jk}_{(1)} \\ 1 \end{bmatrix}}, \tag{3.41}$$

where the vectors $\tilde{\mathbf{y}}^{jk}_{(1)}$, $j = 1, \ldots, N_{\mathcal{M}}$, $k = 1, \ldots, n^{\mathcal{M}}_j$, are computed using (3.31).

Step 3: Rotating and translating $\mathcal{M}^j_{(2)}$ within the 1-2 plane towards its corresponding position in the net

The transformations required to map a face $\mathcal{M}^j_{(2)}$ with node position vectors $\tilde{\mathbf{y}}^{jk}_{(2)}$, $k = 1, \ldots, n^{\mathcal{M}}_j$, towards its configuration in the net $\mathcal{M}^j_{(3)}$ with node position vectors $\tilde{\mathbf{y}}^{jk}_{(3)}$ are illustrated in Fig. 3.18d–f.

Referring to Fig. 3.19, a rotation of $\tilde{\mu}_{jk}$ in the 1-2 plane and a translation by $\tilde{\mathbf{b}}^{jk} \in \mathrm{span}(\mathbf{e}_1, \mathbf{e}_2)$ must be applied for each of the $\tilde{n}_j - 1$ faces crossed by the path $\tilde{\boldsymbol{\gamma}}^j(\eta)$ before reaching $\mathcal{M}^j$. The parameters $\tilde{\mu}_{jk}$ and $\tilde{\mathbf{b}}^{jk}$ are computed as follows:

$$\tilde{\mu}_{jk} = \varphi(-\tilde{\mathbf{z}}^{jk}_{\mathrm{O}(1)}) - \varphi(\tilde{\mathbf{z}}^{jk}_{\mathrm{I}(1)}), \tag{3.42}$$

$$\tilde{\mathbf{b}}^{jk} = \mathbf{R}_3\left((\textstyle\sum_{l=1}^{k-1} \tilde{\mu}_{jl}) - \varphi(\tilde{\mathbf{z}}^{jk}_{\mathrm{I}(1)})\right)\left(\tilde{\mathbf{z}}^{jk}_{\mathrm{O}(1)} + \tilde{\mathbf{y}}^{jk}_{\mathrm{O}(1)} - \tilde{\mathbf{y}}^{jk}_{\mathrm{I}(1)}\right), \tag{3.43}$$

where $\mathbf{R}_3(\cdot)$ represents a rotation about the 3-axis and is defined in (2.22).

As shown in Fig. 3.18d–f, the face $\mathcal{M}^j_{(2)}$ is mapped to its corresponding configuration in the net $\mathcal{M}^j_{(3)}$ by applying a rotation of $\sum_{k=1}^{\tilde{n}_j-1} \tilde{\mu}_{jk}$ in the 1-2 plane followed by a translation of $\sum_{k=1}^{\tilde{n}_j-1} \tilde{\mathbf{b}}^{jk}$:

$$\boxed{\begin{bmatrix} \tilde{\mathbf{y}}^{jk}_{(3)} \\ 1 \end{bmatrix} = \mathbf{T}\left(\textstyle\sum_{l=1}^{\tilde{n}_j-1} \tilde{\mathbf{b}}^{jl}\right)\, \mathbf{Q}_3\left(\textstyle\sum_{l=1}^{\tilde{n}_j-1} \tilde{\mu}_{jl}\right) \begin{bmatrix} \tilde{\mathbf{y}}^{jk}_{(2)} \\ 1 \end{bmatrix}}, \tag{3.44}$$

where the vectors $\tilde{\mathbf{y}}^{jk}_{(2)}$, $j = 1, \ldots, N_{\mathcal{M}}$, $k = 1, \ldots, n^{\mathcal{M}}_j$, are computed using (3.41).

As previously indicated, the vectors $\tilde{\mathbf{y}}^{jk}_{(3)}$, $j = 1, \ldots, N_{\mathcal{M}}$, $k = 1, \ldots, n^{\mathcal{M}}_j$, provide the position of the nodes of each face of the goal mesh mapped to its corresponding position in the net. Once these vectors have been determined, the data of Table 2.1 required to the define fold pattern of the origami sheet associated with the net can be readily obtained.

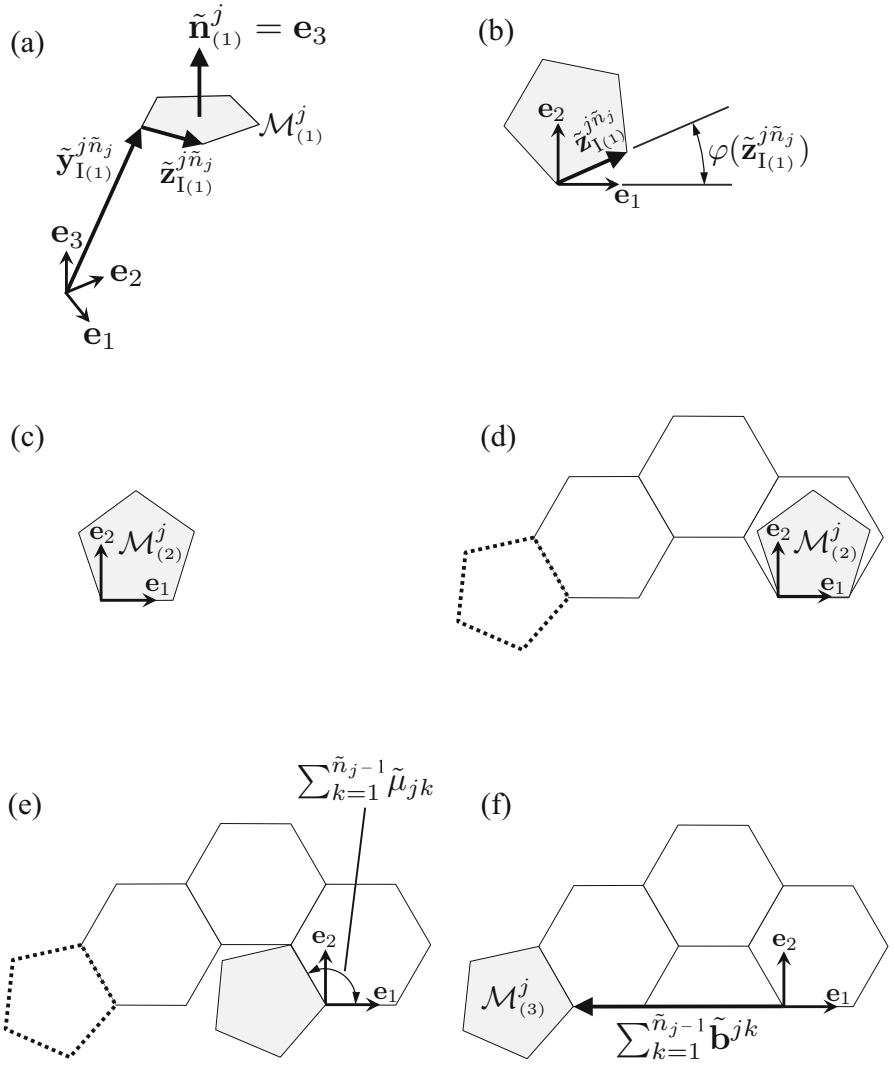


Fig. 3.18 Schematic showing the configurations of a mesh face during the transformations applied in (3.41) and (3.44): (**a**) Face $\mathcal{M}^j_{(1)}$ with unit normal vector $\tilde{\mathbf{n}}^j_{(1)}$ aligned with $\mathbf{e}_3$ (obtained via (3.31)); (**b**) Face after translation via $\mathbf{T}^{-1}(\tilde{\mathbf{y}}^{j\tilde{n}_j}_{\mathrm{I}(1)})$ to place its node with position vector $\tilde{\mathbf{y}}^{j\tilde{n}_j}_{\mathrm{I}(1)}$ at the origin; (**c**) Face after rotation via $\mathbf{Q}_3^{-1}(\varphi(\tilde{\mathbf{z}}^{j\tilde{n}_j}_{\mathrm{I}(1)}))$ to align its edge along $\tilde{\mathbf{z}}^{j\tilde{n}_j}_{\mathrm{I}(1)}$ with $\mathbf{e}_1$. The configuration of the face after this transformation corresponds to $\mathcal{M}^j_{(2)}$; (**d**) Position of the considered face in the net shown with the dotted line; (**e**) Rotation of the face via $\mathbf{Q}_3(\sum_{k=1}^{\tilde{n}_j-1}\tilde{\mu}_{jk})$; (**f**) Translation of the face via $\mathbf{T}(\sum_{k=1}^{\tilde{n}_j-1}\tilde{b}^{jk})$ to finally place it at its configuration in the net $\mathcal{M}^j_{(3)}$

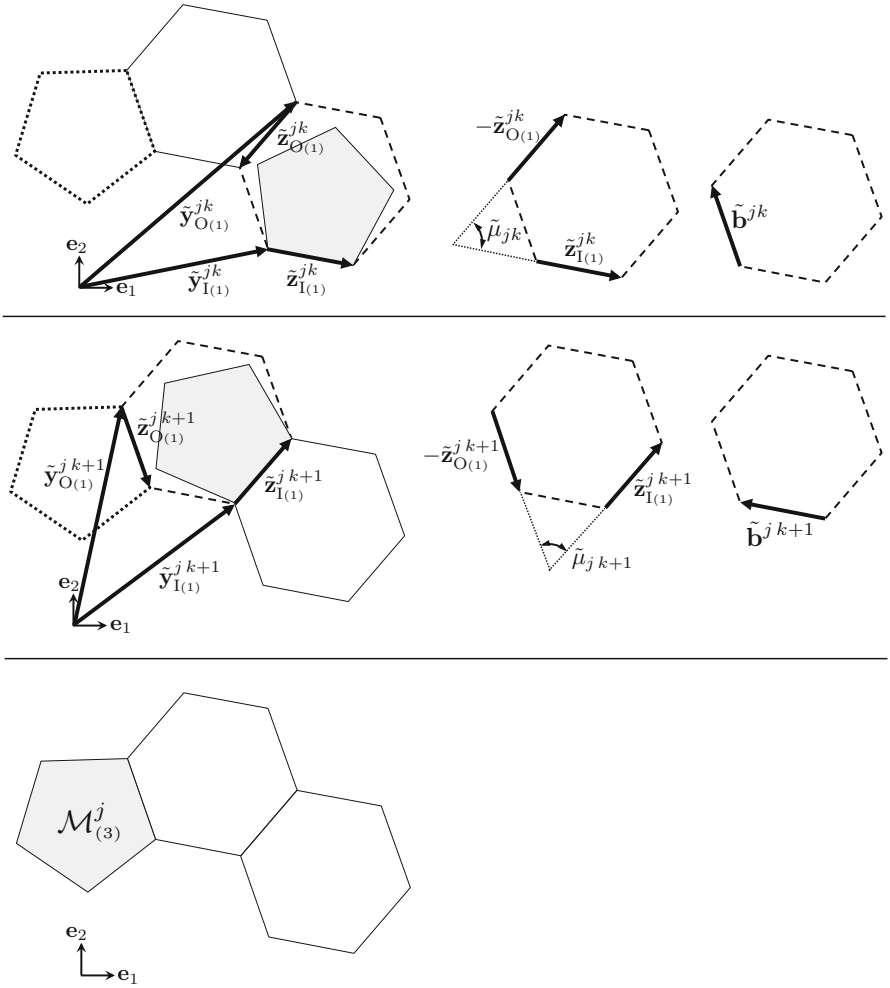


Fig. 3.19 Parameters $\tilde{\mu}_{jk}$ and $\tilde{\mathbf{b}}^{jk}$ used to map a face towards its corresponding position in the net

3.2.5 Determination of Folding Motion

Following the steps of the unfolding polyhedra method listed in Sect. 3.2.1, once a net has been obtained for a given goal mesh $\mathcal{M}$, the next step is to determine a history of folding motion from the reference configuration $\mathcal{S}_0$ (i.e., the net) towards the *goal configuration* $\mathcal{S}_\star$ that matches $\mathcal{M}$. The determination of such a history of folding motion is addressed in this section.

We utilize the model for origami with creased folds presented in Chap. 2 to simulate the folding motion of the sheets determined via the unfolding polyhedra method. As explained in Sect. 2.6, the folding motion is simulated by *incrementally*

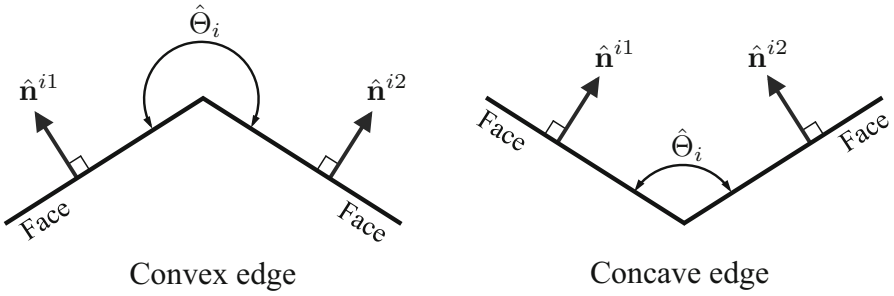


Fig. 3.20 Side views of faces in the goal mesh $\mathcal{M}$ connected by a *convex edge* and a *concave edge*. The edge dihedral angle $\hat{\Theta}_i$ and unit normal vectors $\hat{\mathbf{n}}^{i1}$ and $\hat{\mathbf{n}}^{i2}$ of the connected faces are shown. For convex edges, $\hat{\Theta}_i \geq \pi$; for concave edges, $\hat{\Theta}_i < \pi$

updating the values of the fold angles using guess increments and then applying any required corrections such that every folded configuration is valid. Since an origami sheet obtained via the unfolding polyhedra method *does not contain any interior fold intersections*, the kinematic constraints for origami with creased folds (2.39) do not apply and are not considered. This represents a practical engineering advantage of the unfolding polyhedra method.

We define $\Delta\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{N_\mathcal{F}}$ as the vector whose components correspond to the lth set of guess increments for the fold angles (see (2.72)):

$$\Delta\hat{\boldsymbol{\theta}}^l = [\,\Delta\hat{\theta}^l_1 \;\; \cdots \;\; \Delta\hat{\theta}^l_{N_\mathcal{F}}\,]^\top \qquad l = 1, \ldots, N_{\text{inc}}. \tag{3.45}$$

where N_{inc} is the *total number of increments*. The guess increments for the fold angles $\Delta\hat{\boldsymbol{\theta}}^l$, $l = 1, \ldots, N_{\text{inc}}$, must be determined from the given goal mesh $\mathcal{M}$. For this purpose, first let $\hat{\theta}^\star_1, \ldots, \hat{\theta}^\star_{N_\mathcal{F}}$ be the fold angles of each creased fold in the goal configuration $\mathcal{S}_\star$ (i.e., the *goal fold angles*). To determine $\hat{\theta}^\star_i$, the *dihedral angle* $\hat{\Theta}_i$ between the two faces in $\mathcal{M}$ connected by the edge associated with the ith creased fold is calculated as follows (see Fig. 3.20):

$$\hat{\Theta}_i = \begin{cases} \pi + \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for convex edges} \\ \\ \pi - \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for concave edges,} \end{cases} \tag{3.46}$$

$$i = 1, \ldots, N_\mathcal{F},$$

where $\hat{\mathbf{n}}^{i1}, \hat{\mathbf{n}}^{i2} \in \mathbb{R}^3$ are the unit normal vectors in $\mathcal{M}$ of the two faces connected by the edge associated with the ith creased fold. The reader is asked in Problem 3.6 to propose an algorithm to determine whether an interior edge of $\mathcal{M}$ is convex or concave.

After determining the dihedral angles $\hat{\Theta}_1, \ldots, \hat{\Theta}_{N_\mathcal{F}}$ using (3.46), the goal fold angles $\hat{\theta}_1^\star, \ldots, \hat{\theta}_{N_\mathcal{F}}^\star$ are calculated as follows:

$$\hat{\theta}_i^\star = \pi - \hat{\Theta}_i \qquad i = 1, \ldots, N_\mathcal{F}. \tag{3.47}$$

We define $\hat{\boldsymbol{\theta}}^\star \in \mathbb{R}^{N_\mathcal{F}}$ as the vector whose components are the goal fold angles:

$$\hat{\boldsymbol{\theta}}^\star = [\, \hat{\theta}_1^\star \;\; \cdots \;\; \hat{\theta}_{N_\mathcal{F}}^\star \,]^\top. \tag{3.48}$$

The increments for the fold angles $\Delta\hat{\boldsymbol{\theta}}^l, l = 1, \ldots, N_\text{inc}$, are then simply given as follows:

$$\Delta\hat{\boldsymbol{\theta}}^l = \frac{\hat{\boldsymbol{\theta}}^\star}{N_\text{inc}} \qquad l = 1, \ldots, N_\text{inc}. \tag{3.49}$$

As previously stated, the kinematic constraints for origami with creased folds (2.39) are not considered here because origami sheets corresponding to nets do not contain any interior fold intersections. Thus, the fold angles determined at each increment $\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{N_\mathcal{F}}, l = 1, \ldots, N_\text{inc}$, simply correspond to $\sum_{m=1}^{l} \Delta\hat{\boldsymbol{\theta}}^m$ and hence the fold angles at the last increment (N_inc) *exactly correspond to the goal fold angles*:

$$\hat{\boldsymbol{\theta}}^{N_\text{inc}} = \sum_{m=1}^{N_\text{inc}} \Delta\hat{\boldsymbol{\theta}}^m = \sum_{m=1}^{N_\text{inc}} \frac{\hat{\boldsymbol{\theta}}^\star}{N_\text{inc}} = N_\text{inc} \frac{\hat{\boldsymbol{\theta}}^\star}{N_\text{inc}} = \hat{\boldsymbol{\theta}}^\star. \tag{3.50}$$

3.2.6 *Limitations of the Unfolding Polyhedra Method*

This section provides a brief discussion of the limitations of the unfolding polyhedra method. The question of whether it is always possible to determine a *valid net* (i.e., a net that *does not have any overlaps*) for a given goal mesh has been investigated for centuries [17]. Various counterexamples have been found for *non-convex* polyhedral surfaces [22, 37, 38]; however, the question remains open for *convex* polyhedra [17].

There is evidence supporting the conjecture that every convex goal mesh has at least one valid (non-overlapping) net [39]. For instance, a convex goal mesh that does not have a valid net has not yet been found [17]. However, a rigorous mathematical proof remains elusive. For convex goal meshes, various heuristic procedures have been proposed for the determination of a spanning tree that allows for the computation of a valid net. The most complete compilation to date of such procedures was presented by Schlickenrieder [21].

Examples of non-convex goal meshes that do not have a valid net have already been reported [22, 37, 38]. Figure 3.21 shows three distinct non-convex goal meshes

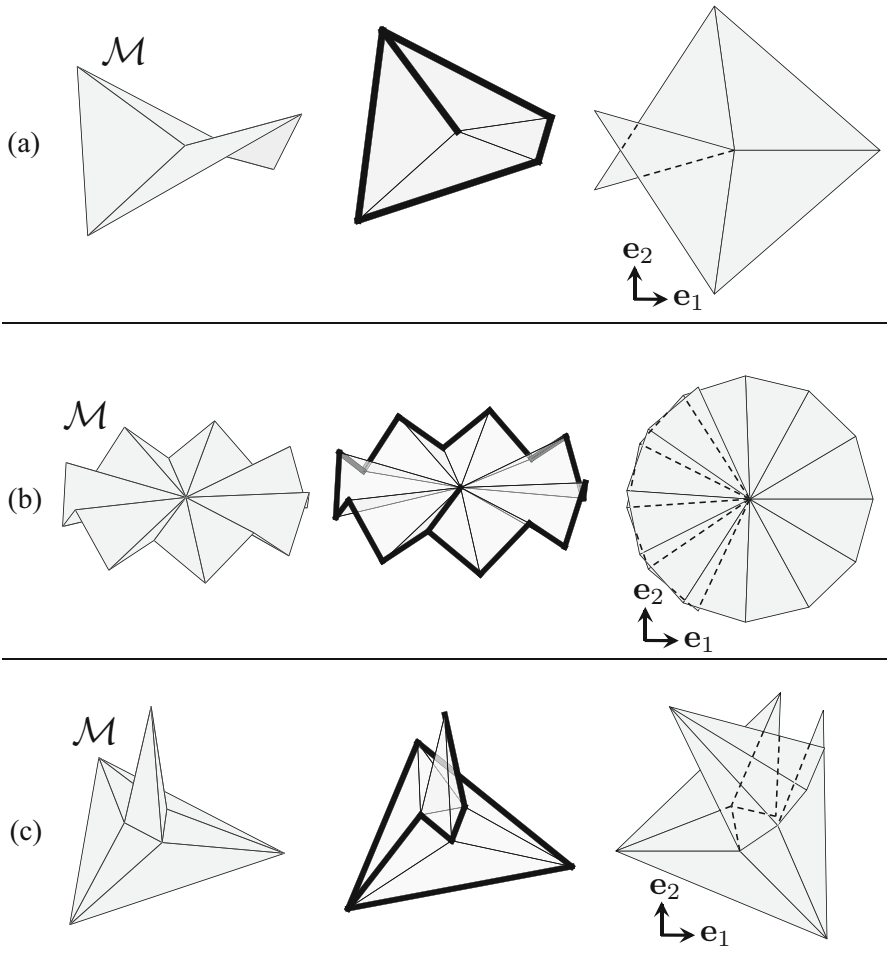


Fig. 3.21 Examples of goal meshes for which a valid (i.e., non-overlapping) net cannot be determined. The left column shows the goal mesh, the center column shows the boundary edges, and the right column shows the determined invalid nets. (**a**) and (**b**) Meshes of saddle surfaces. (**c**) A triangulated hat open at the base (after [22])

that do not have a valid net, regardless of the selected spanning tree. Despite such issues, the unfolding polyhedra method can indeed be applied to determine a valid net for some non-convex goal meshes (see Fig. 3.22 for examples). However, the ability of the unfolding polyhedra method to determine a valid net for a non-convex goal mesh cannot be analytically determined in general. For these cases, one has to rely only on trial and error approaches whereby one determines many possible guess nets and checks if at least one of them is valid.

The heuristic procedures examined by Schlickenrieder in [21] for the computation of a spanning tree are extended for non-convex goal meshes in [29, 40].

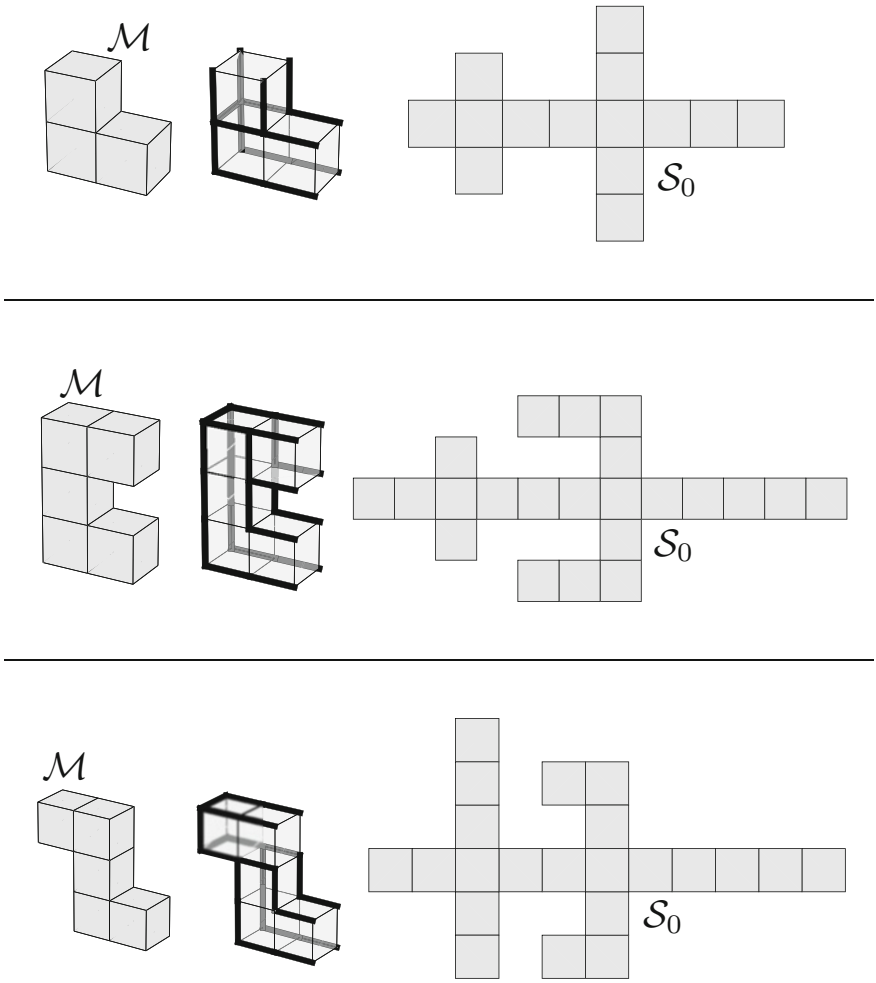


Fig. 3.22 Examples of non-convex goal meshes for which a valid net can be determined. The left column shows the goal mesh, the center column shows the boundary edges, and the right column shows the determined valid nets

However, as previously mentioned, the process of generating a net that does not have overlaps is more involved for non-convex goal meshes. There is ongoing research on more general unfoldings comprised of multiple disconnected regions or having cuts that are not limited to the edges of the goal mesh [17, 29–32, 41–46]. These more advanced variants of the unfolding polyhedra method are beyond the scope of this chapter and the curious reader is referred to the bibliography provided herein. In Chap. 4, we address another method for origami design that is applicable to both convex and non-convex goal meshes.

3.3 Examples of the Unfolding Polyhedra Method

This section provides representative examples of the unfolding polyhedra method described in Sect. 3.2. Such a method has been implemented in MATLAB. The faces of the goal meshes and the nets are visualized as filled three-dimensional polygons using the MATLAB command `fill3`. The complete set of MATLAB scripts used to generate the examples is included in the Supplemental Materials for this chapter and described in Appendix B.2. We use the MATLAB implementation of the kinematic model presented in Chap. 2 to simulate the folding motion of the origami sheets corresponding to the nets.

Figure 3.23 shows goal meshes for the surfaces of the *Platonic solids* and their corresponding nets. The Platonic solids are convex polyhedra constructed by congruent, regular, polygonal faces with the same number of faces meeting at each node. The *regular tetrahedron* (Fig. 3.23a) has four congruent equilateral triangle faces. The cube (Fig. 3.23b) has six congruent square faces. The *regular octahedron* (Fig. 3.23c) has eight congruent equilateral triangle faces. The *regular dodecahedron* (Fig. 3.23d) has 12 congruent regular pentagon faces and the *regular icosahedron* (Fig. 3.23e) has 20 congruent equilateral triangle faces.

Figure 3.24 shows four different nets for a goal mesh of a dome determined by selecting different mesh faces as the reference face of the spanning tree (refer to Sect. 3.2.3). A valid net is determined for all cases.

Often, the goal shape for a particular application of origami may not be initially given as a polygonal mesh but rather as a smooth surface. In those situations, the designer or engineer must first decide a polygonal mesh discretization of such a smooth surface prior to the utilization of the unfolding polyhedra method. To illustrate this, Fig. 3.25 shows different polygonal mesh discretizations for a sphere goal shape and their corresponding nets determined using the unfolding polyhedra method.

Finally, Figs. 3.26, 3.27, 3.28, 3.29 show the determined nets and simulated folding motion associated with goal meshes for a truncated icosahedron, an ellipsoid, a shelter, and a parabolic disk. As observed from these figures, the unfolding polyhedra method is able to successfully generate a net that can be folded to match each of the considered goal meshes.

Chapter Summary

This chapter presented a method known as *unfolding polyhedra* for the design of origami structures with creased folds that achieve targeted three-dimensional shapes. In this method, the goal shape is represented as a three-dimensional polygonal mesh termed as the *goal mesh*. The objective in unfolding polyhedra is to determine a *net* for the goal mesh, which is a connected, non-overlapping sheet corresponding to a flattening of the goal mesh onto a plane and has boundary segments that correspond to cuts made exclusively on edges of the goal mesh.

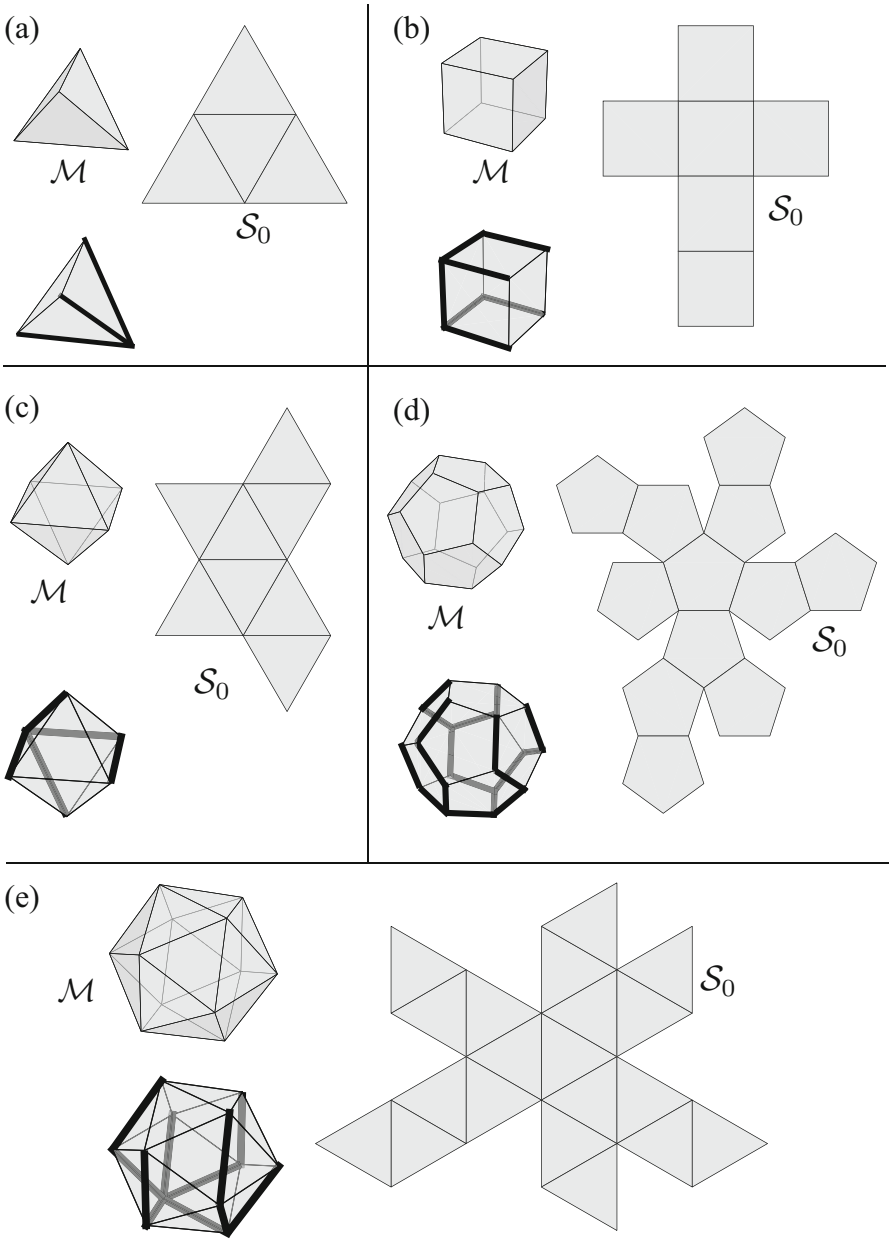


Fig. 3.23 Goal meshes for the surfaces of the Platonic solids, boundary edges generated after the determination of the spanning tree, and corresponding nets. (**a**) Tetrahedron. (**b**) Cube. (**c**) Octahedron. (**d**) Dodecahedron. (**e**) Icosahedron

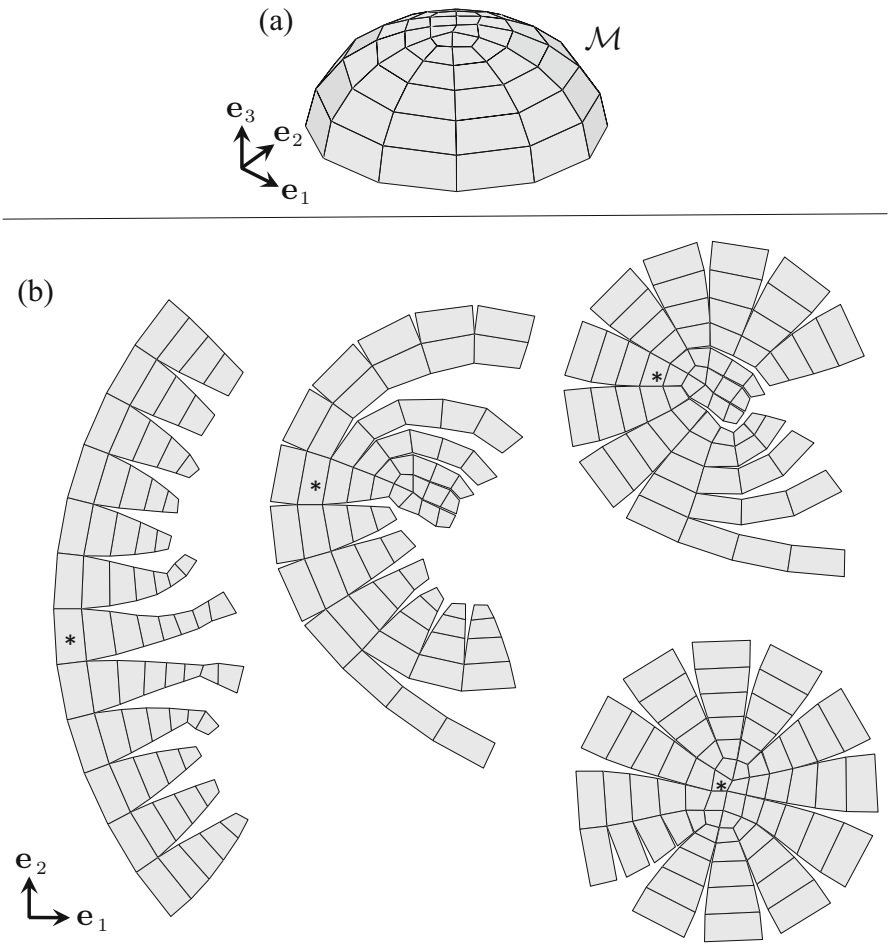


Fig. 3.24 (**a**) Goal mesh $\mathcal{M}$ of a dome. The base of the dome is open. (**b**) Four different nets for the goal mesh of a dome shown in (**a**). The reference faces used to start the generation of the spanning tree are marked with the symbol $*$

We presented the problem description and solution approach of the unfolding polyhedra method in Sect. 3.2.1. The data required to define the goal mesh was described in Sect. 3.2.2. We then addressed the steps towards the design of a net including the determination of a spanning tree (Sect. 3.2.3), mapping of the faces in the goal mesh towards their position in the net (Sect. 3.2.4), and simulation of the folding motion from the planar net configuration to the configuration that matches the goal mesh (Sect. 3.2.5). A brief discussion of the limitations of the unfolding polyhedra method was provided in Sect. 3.2.6. Representative examples of the implemented method considering various goal meshes were provided in Sect. 3.3.

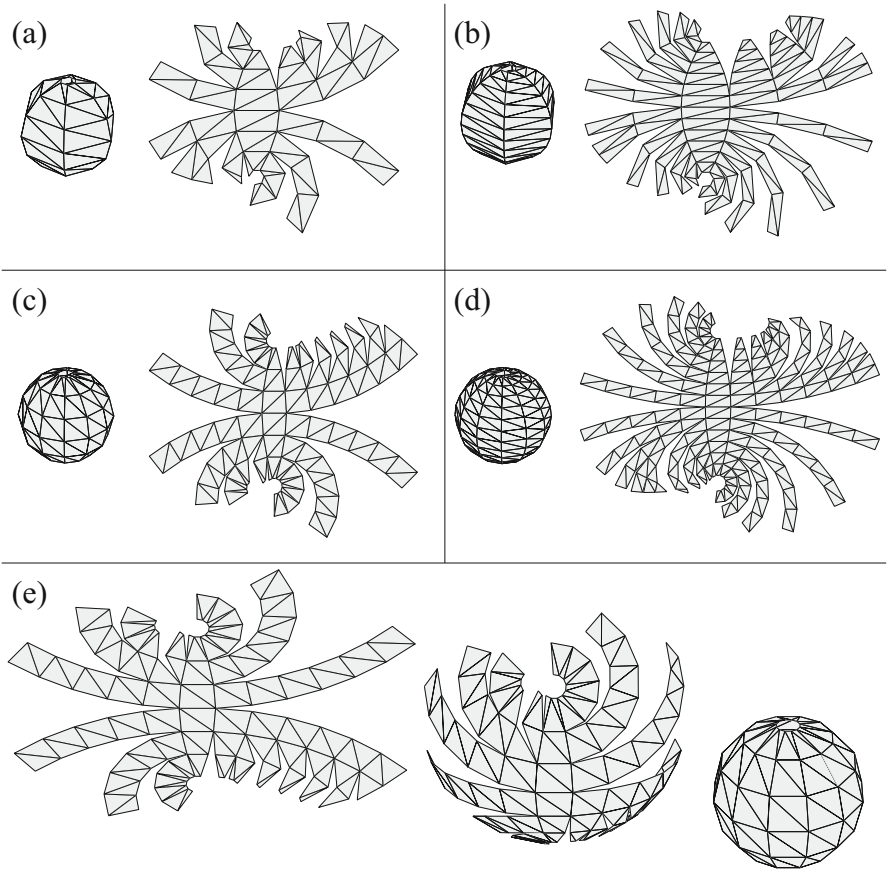


Fig. 3.25 (**a**)–(**d**) Four different mesh discretizations of a sphere and example associated nets. The mesh shown in (**a**) has 6 triangle pairs along the azimuthal direction and 6 triangle pairs along the inclination direction (i.e., 6×6 discretization). (**b**) 6×12 discretization. (**c**) 12×6 discretization. (**d**) 12×12 discretization. (**e**) Folding motion of the net in (**c**) associated with the 12×6 discretization

Problems

3.1 Consider the goal mesh $\mathcal{M}$ of an oblique pyramid illustrated in Fig. 3.30. (a) Determine the position vectors of the nodes $\hat{\mathbf{y}}^i$, $i = 1, \ldots, N_{\mathcal{N}}$, and the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$. Define $\mathbf{C}^{\mathcal{M}}$ such that the unit normal vectors of the mesh faces point towards the exterior of the pyramid. (b) Determine the vectors $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}}$ and unit normal vector $\tilde{\mathbf{n}}^j$ of each face in $\mathcal{M}$.

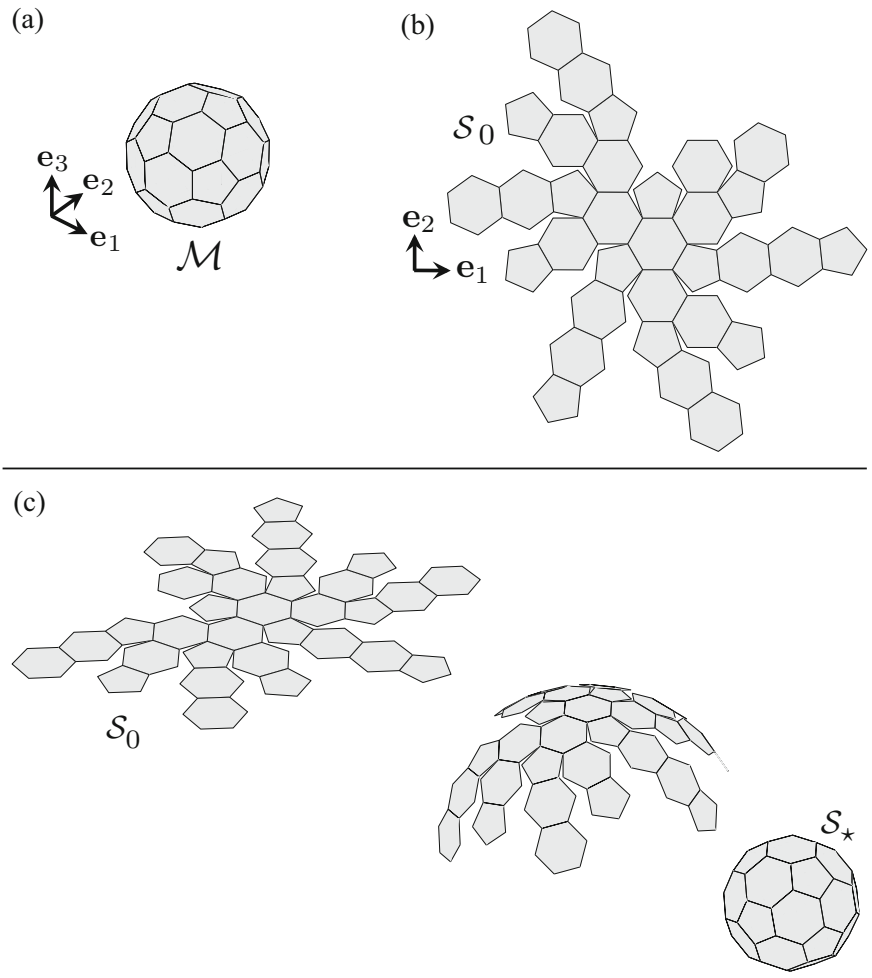


Fig. 3.26 (**a**) Goal mesh of a truncated icosahedron. (**b**) A net of the truncated icosahedron. (**c**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

3.2 Consider the goal mesh $\mathcal{M}$ having a single interior node shown in Fig. 3.31a. Determine the vectors $\mathbf{z}^{11}, \ldots, \mathbf{z}^{14}$, the face corner angles $\phi_{11}, \ldots, \phi_{14}$, and the unit normal vectors $\mathbf{n}^{11}, \ldots, \mathbf{n}^{14}$.

3.3 Propose an algorithm to determine the components of the edge connectivity matrix $\mathbf{C}^{\mathcal{E}_I}$ (defined in (3.14)) from the given mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$.

3.4 Propose an algorithm to determine the components of the node connectivity matrix $\mathbf{C}^{\mathcal{N}_I}$ (defined in (3.16)) from the given mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$.

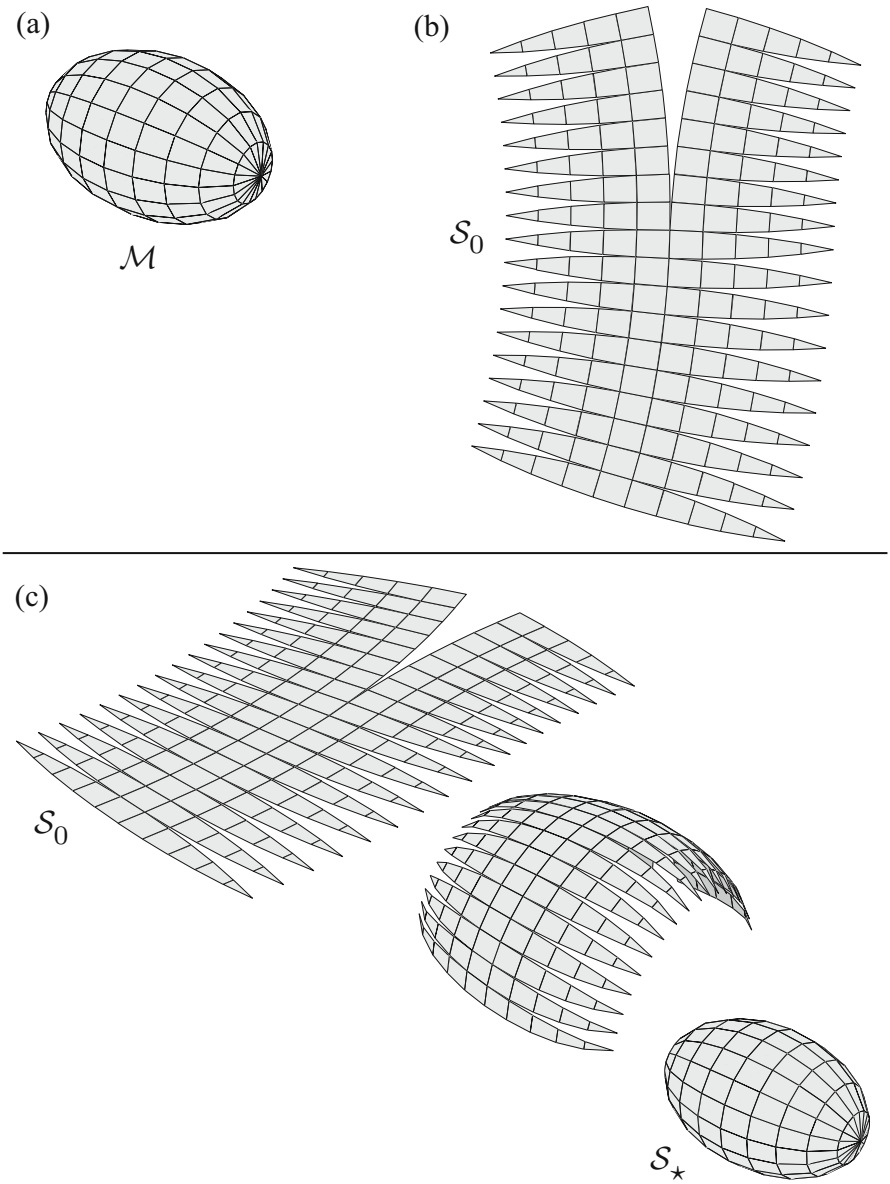


Fig. 3.27 (**a**) Goal mesh $\mathcal{M}$ for an ellipsoidal shape comprised of quadrilaterals and triangles. (**b**) Net of the goal mesh. (**c**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

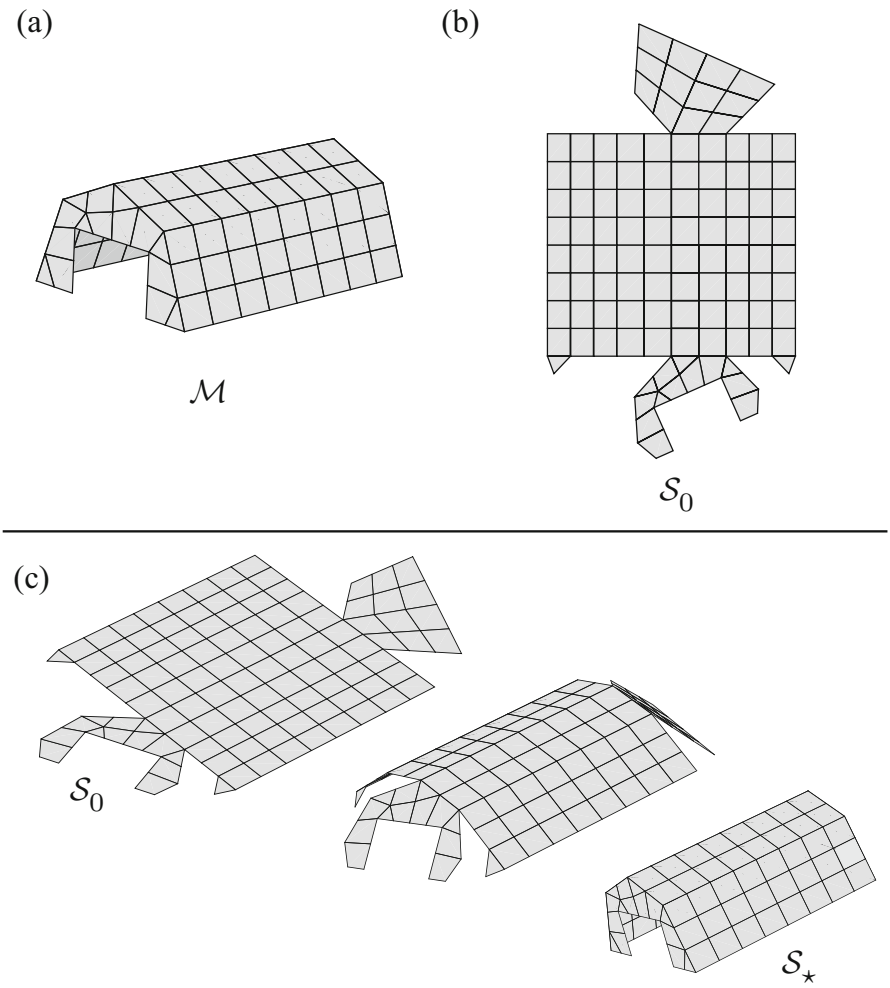


Fig. 3.28 (**a**) Goal mesh $\mathcal{M}$ for a shelter comprised of quadrilaterals and triangles. (**b**) Net of the goal mesh. (**c**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

3.5 Propose and implement two procedures to determine a spanning tree different than the one described in Sect. 3.2.3. Discuss the differences among the procedures and compare them using representative examples.

3.6 Consider the faces $\mathcal{M}^j$ and $\mathcal{M}^l$ of a goal mesh having nodes with position vectors $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}}$ and $\tilde{\mathbf{y}}^{l1}, \ldots, \tilde{\mathbf{y}}^{ln_l^{\mathcal{M}}}$, respectively. Suppose that these faces are connected at one edge such that $\tilde{\mathbf{y}}^{j1} = \tilde{\mathbf{y}}^{l2}$ and $\tilde{\mathbf{y}}^{j2} = \tilde{\mathbf{y}}^{l1}$. Propose an algorithm to determine whether such an edge is convex or concave. Refer to Fig. 3.20 for these definitions.

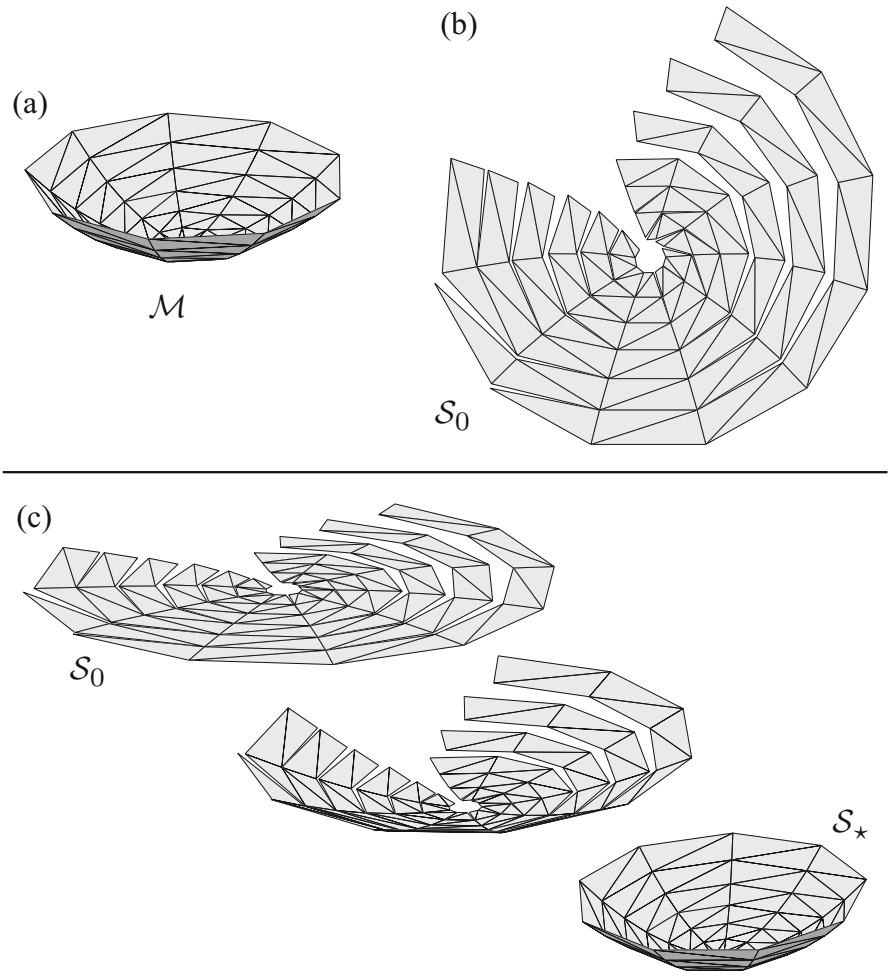


Fig. 3.29 (**a**) Goal mesh $\mathcal{M}$ for a parabolic dish. (**b**) Net of the goal mesh. (**c**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

3.7 Implement the unfolding polyhedra method in a computational environment and determine nets for the 13 Archimedean polyhedra.

3.8 Consider the triangle mesh face $\mathcal{M}^j$ shown in Fig. 3.32. (a) Determine the rotation in the form of (3.31) that is required to align the unit normal vector of $\mathcal{M}^j$ with $\mathbf{e}_3$. (b) Determine the position vectors $\tilde{\mathbf{y}}^{j1}_{(1)}$, $\tilde{\mathbf{y}}^{j2}_{(1)}$, and $\tilde{\mathbf{y}}^{j3}_{(1)}$ of the rotated face $\mathcal{M}^j_{(1)}$. (c) Verify that $\tilde{\mathbf{n}}^j_{(1)} = \mathbf{e}_3$.

3.9 Consider the quadrilateral mesh face $\mathcal{M}^j$ shown in Fig. 3.33. (a) Determine the rotation in the form of (3.31) that is required to align the unit normal vector of $\mathcal{M}^j$

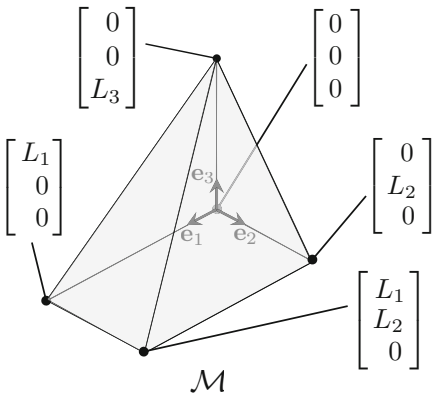


Fig. 3.30 Schematic for Problem 3.1: Goal mesh of an oblique pyramid and coordinates of its nodes

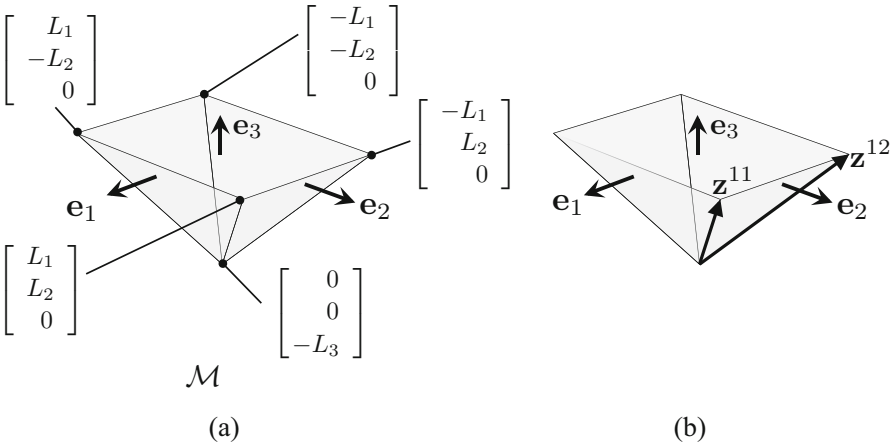


Fig. 3.31 Schematic for Problem 3.2. The top of the pyramid is open. (**a**) Goal mesh $\mathcal{M}$ having a single interior node. The coordinates of the nodes are shown. (**b**) Vectors $\mathbf{z}^{11}$ and $\mathbf{z}^{12}$

with $\mathbf{e}_3$. (b) Determine the position vectors $\tilde{\mathbf{y}}_{(1)}^{j1}$, $\tilde{\mathbf{y}}_{(1)}^{j2}$, $\tilde{\mathbf{y}}_{(1)}^{j3}$, and $\tilde{\mathbf{y}}_{(1)}^{j4}$ of the rotated face $\mathcal{M}_{(1)}^{j}$. (c) Verify that $\tilde{\mathbf{n}}_{(1)}^{j} = \mathbf{e}_3$.

3.10 Show that the position vectors $\tilde{\mathbf{y}}_{(3)}^{jk}$ determined using (3.44) are contained in $\text{span}(\mathbf{e}_1, \mathbf{e}_2)$.

3.11 Derive an unfolding map from a goal mesh $\mathcal{M}$ to its associated net assuming that the net is placed in the 2-3 plane.

3.12 Derive an unfolding map from a goal mesh $\mathcal{M}$ to its associated net assuming that the net is placed in an arbitrary plane in $\mathbb{R}^3$.

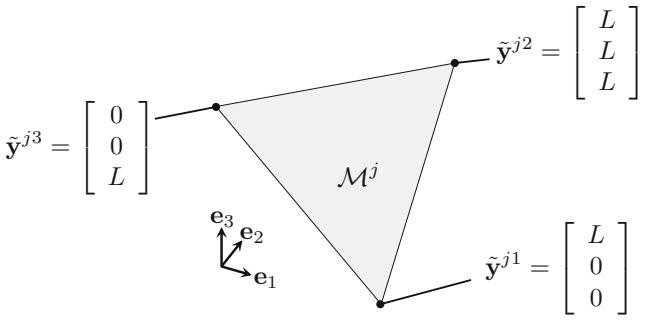


Fig. 3.32 Schematic for Problem 3.8: Face $\mathcal{M}^j$ and position vectors of its nodes

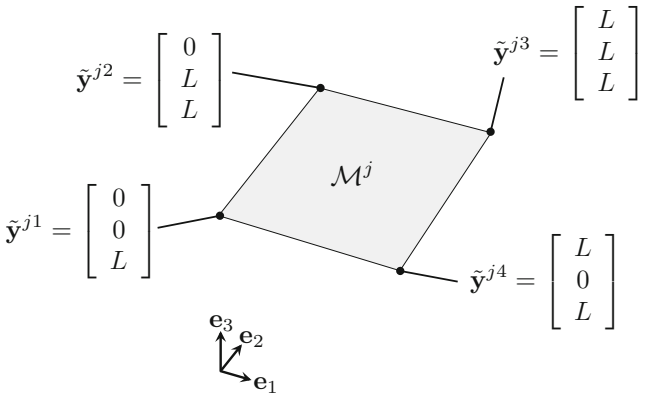


Fig. 3.33 Schematic for Problem 3.9: Face $\mathcal{M}^j$ and position vectors of its nodes

References

1. N. Turner, B. Goodwine, M. Sen, A review of origami applications in mechanical engineering. Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci. (2015). https://doi.org/10.1177/0954406215597713
2. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr, D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
3. N. Bassik, G.M. Stern, D.H. Gracias, Microassembly based on hands free origami with bidirectional curvature. Appl. Phys. Lett. **95**(9), 091901 (2009)
4. B. Shin, S.M. Felton, M.T. Tolley, R.J. Wood, Self-assembling sensors for printable machines, in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2014), pp. 4417–4422
5. S. Felton, M. Tolley, E. Demaine, D. Rus, R. Wood, A method for building self-folding machines. Science **345**(6197), 644–646 (2014)

6. A. Firouzeh, Y. Sun, H. Lee, J. Paik, Sensor and actuator integrated low profile robotic origami, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 4937–4944
7. E. Morris, D.A. McAdams, R. Malak, The state of the art of origami-inspired products: a review, in *Proceedings of the ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, Paper No. DETC2016-59629 (American Society of Mechanical Engineers, New York, 2016), p. V05BT07A014
8. D. Ding, F. Zhang, S. Lan, Y.Y. Liu, S.X. Li, X.H. Xie, Y.C. Chen, Review of materials for origami in fashion design. Adv. Mater. Res. **821**, 790–793 (2013)
9. M. Johnson, Y. Chen, S. Hovet, S. Xu, B. Wood, H. Ren, J. Tokuda, Z. Tse. Fabricating biomedical origami: a state-of-the-art review. Int. J. Comput. Assist. Radiol. Surg. **12**(11), 2023–2032 (2017)
10. K. Kuribayashi, K. Tsuchiya, Z. You, D. Tomus, M. Umemoto, T. Ito, and M. Sasaki, Self-deployable origami stent grafts as a biomedical application of Ni-rich TiNi shape memory alloy foil. Mater. Sci. Eng. A **419**(1–2), 131–137 (2006)
11. B.J. Edmondson, L.A. Bowen, C.L. Grames, S.P. Magleby, L.L. Howell, T.C. Bateman, Oriceps: origami-inspired forceps, in *Proceedings of the ASME 2013 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, Paper No. SMASIS2013-3299 (American Society of Mechanical Engineers, New York, 2013), p. V001T01A027
12. S.A. Zirbel, R.J. Lang, M.W. Thomson, D.A. Sigel, P.E. Walkemeyer, B.P. Trease, S.P. Magleby, L.L. Howell, Accommodating thickness in origami-based deployable arrays. J. Mech. Des. **135**(11), 111005 (2013)
13. J. Morgan, S.P. Magleby, L.L. Howell, An approach to designing origami-adapted aerospace mechanisms. J. Mech. Des. **138**(5), 052301 (2016)
14. R. Salazar, S. Murthy, C. Pellazar, A. Stoica, TransFormers for lunar extreme environments: large origami deployable solar reflectors, in *2017 IEEE Aerospace Conference* (IEEE, Piscataway, 2017), pp. 1–7
15. T. Tachi, Origamizing polyhedral surfaces. IEEE Trans. Vis. Comput. Graph. **16**(2), 298–311 (2010)
16. R.J. Lang, A computational algorithm for origami design, in *Proceedings of the Twelfth Annual Symposium on Computational Geometry* (ACM, New York, 1996), pp. 98–105
17. E.D. Demaine, J. O'Rourke, *Geometric Folding Algorithms* (Cambridge University Press, Cambridge, 2007)
18. T.A. Evans, R.J. Lang, S.P. Magleby, L.L. Howell, Rigidly foldable origami gadgets and tessellations. R. Soc. Open Sci. **2**(9), 150067 (2015)
19. X. Zhou, H. Wang, Z. You, Design of three-dimensional origami structures based on a vertex approach. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. **471**(2181), 20150407 (2015)
20. W. Liu, K. Tai, Optimal design of flat patterns for 3D folded structures by unfolding with topological validation. Comput. Aided Des. **39**(10), 898–913 (2007)
21. W. Schlickenrieder, Nets of polyhedra. Master's thesis, Technische Universität Berlin, 1997
22. M. Bern, E.D. Demaine, D. Eppstein, E. Kuo, A. Mantler, J. Snoeyink, Ununfoldable polyhedra with convex faces. Comput. Geom. **24**(2), 51–62 (2003)
23. H.M. Cundy, A.P. Rollett, Polyhedra, *Mathematical Models* (Oxford University Press, Oxford, 1961), pp. 76–160
24. M.J. Wenninger, *Polyhedron Models* (Cambridge University Press, Cambridge, 1974)
25. S.K. Gupta, D.A. Bourne, K.H. Kim, S.S. Krishnan, Automated process planning for sheet metal bending operations. J. Manuf. Syst. **17**(5), 338 (1998)
26. C.-H. Wang, *Manufacturability-driven decomposition of sheet metal products*. PhD thesis, Carnegie Mellon University, Pittsburgh, 1997
27. B. An, S. Miyashita, M.T. Tolley, D.M. Aukes, L. Meeker, E.D. Demaine, M.L. Demaine, R.J. Wood, D. Rus, An end-to-end approach to making self-folded 3D surface shapes by uniform heating, in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, Piscataway, 2014), pp. 1466–1473

28. K.-U. Jeong, J.-H. Jang, D.-Y. Kim, C. Nah, J.H. Lee, M.-H. Lee, H.-J. Sun, C.-L. Wang, S.Z.D. Cheng, E.L. Thomas, Three-dimensional actuators transformed from the programmed two-dimensional structures via bending, twisting and folding mechanisms. J. Mater. Chem. **21**(19), 6824–6830 (2011)
29. S. Takahashi, H.-Y. Wu, S.H. Saw, C.-C. Lin, H.-C. Yen, Optimized topological surgery for unfolding 3D meshes. Comput. Graph. Forum **30**(7), 2077–2086 (2011)
30. Z. Xi, Y. Kim, Y.J. Kim, J.-M. Lien, Learning to segment and unfold polyhedral mesh from failures. Comput. Graph. **58**, 139–149 (2016)
31. M.A. Alam, I. Streinu, Star-unfolding polygons, in *International Workshop on Automated Deduction in Geometry* (Springer, Berlin, 2014), pp. 1–20
32. E. Akleman, S. Ke, Y. Wu, N. Kalantar, A. Borhani, J. Chen, Construction with physical version of quad-edge data structures. Comput. Graph. **58**, 172–183 (2016)
33. Z. Xi, J.-M. Lien, Continuous unfolding of polyhedra – a motion planning approach, in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, Piscataway, 2015), pp. 3249–3254
34. E.A. Peraza Hernandez, S. Hu, H.W. Kung, D. Hartl, E. Akleman, Towards building smart self-folding structures. Comput. Graph. **37**(6), 730–742 (2013)
35. J.J. Craig, *Introduction to Robotics*, 2nd edn. (Addison-Wesley, New York, 1989)
36. J.S. Dai, Euler–Rodrigues formula variations, quaternion conjugation and intrinsic connections. Mech. Mach. Theory **92**, 144–152 (2015)
37. B. Grünbaum, No-net polyhedra. Geombinatorics **11**(4), 111–114 (2002)
38. B. Grünbaum, A starshaped polyhedron with no net. Geombinatorics **11**, 43–48 (2001)
39. B. Grünbaum, Nets of polyhedra II. Geombinatorics **1**(3), 5–10 (1991)
40. R. Straub, H. Prautzsch, *Creating Optimized Cut-Out Sheets for Paper Models from Meshes* (Citeseer, 2011)
41. B. Aronov, J. O'Rourke, Nonoverlap of the star unfolding. Discrete Comput. Geom. **8**(3), 219–250 (1992)
42. E.D. Demaine, A. Lubiw, A generalization of the source unfolding of convex polyhedra, in *Spanish Meeting on Computational Geometry* (Springer, Berlin, 2011), pp. 185–199
43. V. Chandru, R. Hariharan, N.M. Krishnakumar, Short-cuts on star, source and planar unfoldings, in *International Conference on Foundations of Software Technology and Theoretical Computer Science* (Springer, Berlin, 2004), pp. 174–185
44. P.K. Agarwal, B. Aronov, J. O'Rourke, C.A. Schevon, Star unfolding of a polytope with applications. SIAM J. Comput. **26**(6), 1689–1713 (1997)
45. S. Kiazyk, A. Lubiw, Star unfolding from a geodesic curve. Discrete Comput. Geom. **56**(4), 1018–1036 (2016)
46. J. Itoh, J. O'Rourke, C. Vîlcu, Star unfolding convex polyhedra via quasigeodesic loops. Discrete Comput. Geom. **44**(1), 35–54 (2010)

Chapter 4
Tuck-Folding Method for the Design of Origami Structures with Creased Folds

Abstract Following the description of the unfolding polyhedra method for origami design in the previous chapter, here the focus switches to an origami design method applicable to a much wider spectrum of three-dimensional goal shapes. This chapter presents the *tuck-folding method* to solve the following origami design problem: given a goal shape represented as a polygonal mesh (termed as the goal mesh), find the shape and fold pattern of a planar sheet that can be folded to match the goal mesh, and a history of folding motion from the planar configuration of the sheet to the configuration that matches the goal mesh. The method generates a sheet comprised of the faces of the goal mesh in addition to introduced regions having two rigid faces and three creased folds. The creased folds are used to *tuck-fold* the added regions to form the shape of the goal mesh. We also address the implementation of the tuck-folding method in a computational environment.

4.1 Introduction

As reviewed in Sect. 3.1, origami design is the process of creating an origami structure having desired characteristics such as a targeted shape [1, 2]. Computational methods for origami design have become crucial in the development of origami applications [1, 3–5]. We provide a review of the most well-known methods for origami design in Sect. 1.3. Chapter 3 addresses the method of *unfolding polyhedra* [3, 6] that allows for the determination of the shape and fold pattern of a planar sheet that can be folded towards a three-dimensional polyhedral goal surface. A drawback of the unfolding polyhedra method is that it is not guaranteed to provide a valid (i.e., non-overlapping) sheet design for non-convex goal shapes [3]. There are examples in the literature of non-convex polyhedra for which the method cannot provide a valid sheet design [7]. We discuss this limitation in Sect. 3.2.6. It

Electronic Supplementary Material The online version of this article (https://doi.org/10.1007/978-3-319-91866-2_4) contains supplementary material, which is available to authorized users.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_4

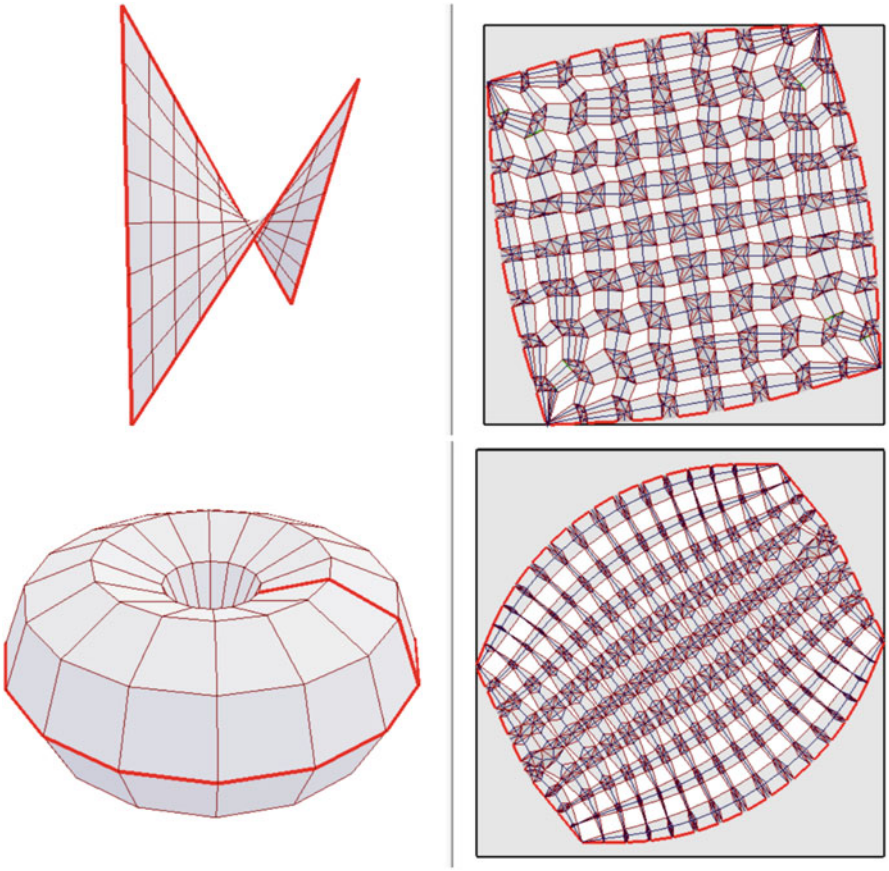


Fig. 4.1 Screenshots of the graphical user interface of Origamizer showing origami sheet designs generated for different goal meshes (left: goal mesh, right: planar sheet with fold pattern). The Origamizer software by Tomohiro Tachi can be obtained from http://www.tsg.ne.jp/TT/software/

also remains an open question as to whether the unfolding polyhedra method can provide a valid sheet design for any convex goal shape [3].

The currently available computational method for origami design that is applicable to the largest spectrum of goal shapes was introduced by Tachi in [2, 8] and was more recently addressed by Tachi and Demaine in [9]. The method determines the shape and fold pattern of a planar sheet that can be folded towards an arbitrary three-dimensional polygonal mesh [2, 8, 10]. This method has been implemented in a software package named *Origamizer*, which is illustrated in Fig. 4.1. In its simplest form, the method generates an origami sheet comprised of all faces of the goal mesh to which are added regions not existing in the goal mesh and having two rigid faces and three creased folds. These added regions are placed between *every* pair of faces of the goal mesh originally connected by an interior edge. Creased folds are used to

tuck-fold the added regions to form the shape of the goal mesh from an initial planar configuration [8]. Thus, throughout the rest of this work, this method for origami design is referred to as the *tuck-folding method*. This method is demonstrated to successfully work on goal polygonal meshes (convex and non-convex) of various complexities in [2].

In this chapter, we address the *tuck-folding method* for origami design presented in [2]. We study this method using notation consistent with the other chapters of the book and with the aim to extend it to account for smooth folds later in Chap. 7. In [2], the folding motion of the sheets designed using the tuck-folding method was not simulated due to the highly complex fold patterns obtained using the full implementation of the method. In view of this, we simplify the method of [2] in order to make the simulation of the folding motion of the designed sheets more straightforward. These simplifications are explained in the following sections. This chapter is organized as follows: Section 4.2 presents the problem description and solution approach of the tuck-folding method and examples of the computationally implemented method considering various goal shapes are provided in Sect. 4.3.

4.2 Tuck-Folding Method Considering Creased Folds

In this section, we describe the various aspects of the tuck-folding method for the design of origami structures with creased folds. Section 4.2.1 provides the design problem definition and the solution approach, Sect. 4.2.2 provides the mathematical description of the goal mesh, the steps in the implementation of the tuck-folding method are addressed in Sects. 4.2.3–4.2.6, and a discussion on the design requirements of the tuck-folding method is provided in Sect. 4.2.7.

4.2.1 Problem Definition

The tuck-folding method aims to solve the following problem in origami design:

- *Given*: A three-dimensional goal shape represented as a polygonal mesh (i.e., the *goal mesh*[1] $\mathcal{M}$)
- *Find*: The shape and fold pattern of a planar sheet that can be folded to match $\mathcal{M}$, and a history of folding motion from the planar configuration of the sheet ($\mathcal{S}_0$) to the goal configuration ($\mathcal{S}_\star$) that matches $\mathcal{M}$

The steps followed in the tuck-folding method to solve this origami design problem are listed as follows:

[1]The goal mesh $\mathcal{M}$ is a *connected*, *orientable*, *2-manifold* polygonal mesh.

1. If the given goal mesh is not topologically equivalent to a disk [2],[2] interior edges of the mesh are assigned as boundary edges to introduce an additional boundary $\partial\mathcal{M}^C$ such that the obtained mesh satisfies the aforementioned topological property. A goal mesh $\mathcal{M}$ with boundary $\partial\mathcal{M} = \partial\mathcal{M}^C \cup \partial\mathcal{M}^O$ and mesh faces $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}} \subset \mathcal{M}$ (where $N_\mathcal{M}$ is the total number of faces in $\mathcal{M}$) is then obtained (the boundary $\partial\mathcal{M}^O$ consists of the boundary edges of the originally given mesh)
2. An origami sheet with reference configuration $\mathcal{S}_0$ comprised of the mesh faces $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}}$ mapped onto a plane and $N_\mathcal{E}^I$ introduced *edge modules* (where $N_\mathcal{E}^I$ is identically the *number of interior edges* of $\mathcal{M}$) is then determined as shown in Fig. 4.2. The edge modules consist of *two rigid faces and three creased folds* and are placed between every pair of faces of $\mathcal{M}$ connected by an interior edge, hence their name. Accordingly, $\mathcal{S}_0$ has $3N_\mathcal{E}^I$ creased folds and $2N_\mathcal{E}^I + N_\mathcal{M}$ rigid faces:

$$N_\mathcal{F} = 3N_\mathcal{E}^I, \qquad N_\mathcal{P} = 2N_\mathcal{E}^I + N_\mathcal{M}, \tag{4.1}$$

   where $N_\mathcal{F}$ and $N_\mathcal{P}$ are, respectively, the number of folds and faces of the designed origami sheet. The challenge in this step is to determine the geometry of the edge modules such that a valid reference configuration $\mathcal{S}_0$ is obtained (refer to Sect. 2.2) and such a sheet matches $\mathcal{M}$ in a known *goal configuration* $\mathcal{S}_\star$ (Fig. 4.3)
3. The final step entails the determination of a history of folding motion from the reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$. Figure 4.3 shows configurations attained by the sheet of Fig. 4.2 during a determined history of folding motion from $\mathcal{S}_0$ to $\mathcal{S}_\star$

We address the previous three steps in the subsequent sections.

An issue that may affect the applicability of the tuck-folding method described in this chapter is that the designed origami sheets contain interior holes as illustrated in Fig. 4.2. This represents a simplification from the implementation of the tuck-folding method presented in [2]. There, the holes are covered with sheet regions having specialized fold patterns that can be folded together with the edge modules to form the shape of the goal mesh; see Fig. 4.1.[3] However, it is noted in [2] that the folding motion of sheets designed using such a full implementation of the tuck-

[2]Two surfaces are said to be *topologically equivalent* (or *homeomorphic*) if there exists a continuous map between the surfaces and such a map has a continuous inverse [11]. For example, a hemispherical surface and a circle are topologically equivalent, as we can continuously map one surface into the other. On the other hand, a closed spherical surface and a circle are *not* topologically equivalent, as we need to introduce additional boundaries to the closed spherical surface in order to continuously map it into a circle.

[3]Another strategy is to cover the holes using compliant membranes such that the deformation of these membranes does not significantly affect the folding motion of the designed origami sheet. Nevertheless, we do not consider such a strategy in this chapter for the sake of simplicity.

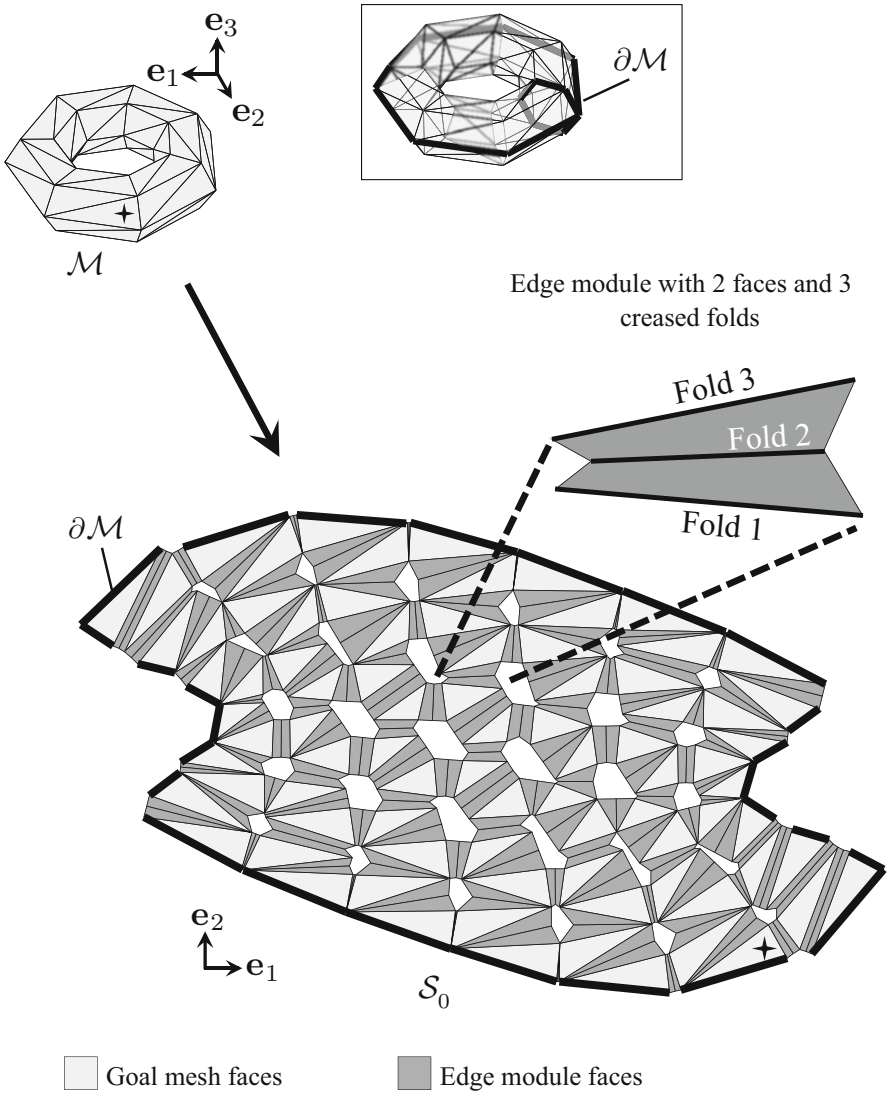


Fig. 4.2 Schematic illustrating Step 2 of the tuck-folding method presented in this chapter: Given a goal mesh $\mathcal{M}$ with boundary $\partial\mathcal{M}$, find the reference configuration $\mathcal{S}_0$ of an origami sheet that matches $\mathcal{M}$ in a known folded configuration. The reference configuration $\mathcal{S}_0$ is comprised of the faces of $\mathcal{M}$ mapped onto a plane and *edge modules* having two rigid faces and three creased folds introduced at all the interior edges of $\mathcal{M}$

folding method was not simulated due to the high complexity of the obtained fold patterns. Here we retain holes in the designed origami sheets in order to make the simulation of the folding motion more straightforward and to ultimately extend the tuck-folding method for the consideration of smooth folds in Chap. 7.

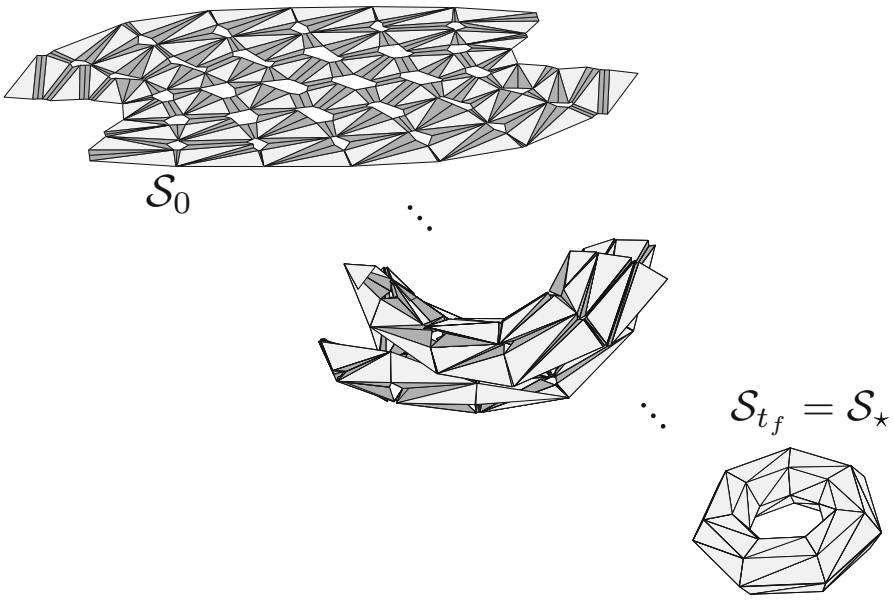


Fig. 4.3 Schematic illustrating Step 3 of the tuck-folding method presented in this chapter: Folding motion of an origami sheet determined in Step 2 from the reference configuration $\mathcal{S}_0$ towards the goal configuration $\mathcal{S}_\star$ that matches the goal mesh $\mathcal{M}$ (refer to Fig. 4.2)

It is remarked that the interior fold intersections of the sheets designed using the present method are not single interior vertices as studied in Chap. 2 but are rather interior holes. Therefore, the kinematic constraints for origami with creased folds derived in Sect. 2.4.2 must be extended to account for this type of interior fold intersections. In Problem 4.1, the reader is asked to use the approach of Sect. 2.4.2 to derive kinematic constraints for origami sheets in which the folds associated with an interior fold intersection are incident to a hole rather than to a single vertex.

4.2.2 Goal Mesh Description

As mentioned in the previous section, the first step of the tuck-folding method is to ensure that the considered goal mesh $\mathcal{M}$ is topologically equivalent to a disk [2]. The boundary of the originally given mesh is denoted $\partial\mathcal{M}^O$. For a given mesh that is *not* topologically equivalent to a disk, interior edges of such a mesh are assigned as boundary edges forming a boundary $\partial\mathcal{M}^C$ to obtain a goal mesh $\mathcal{M}$ with boundary $\partial\mathcal{M} = \partial\mathcal{M}^C \cup \partial\mathcal{M}^O$ that meets the stated topological property. This is illustrated in Fig. 4.4. Given the non-unique nature of assigning boundary edges for $\partial\mathcal{M}^C$, such a process may not only be used to obtain a valid goal mesh $\mathcal{M}$ but also to generate different origami sheet designs for a given mesh. Criteria for determining an optimal

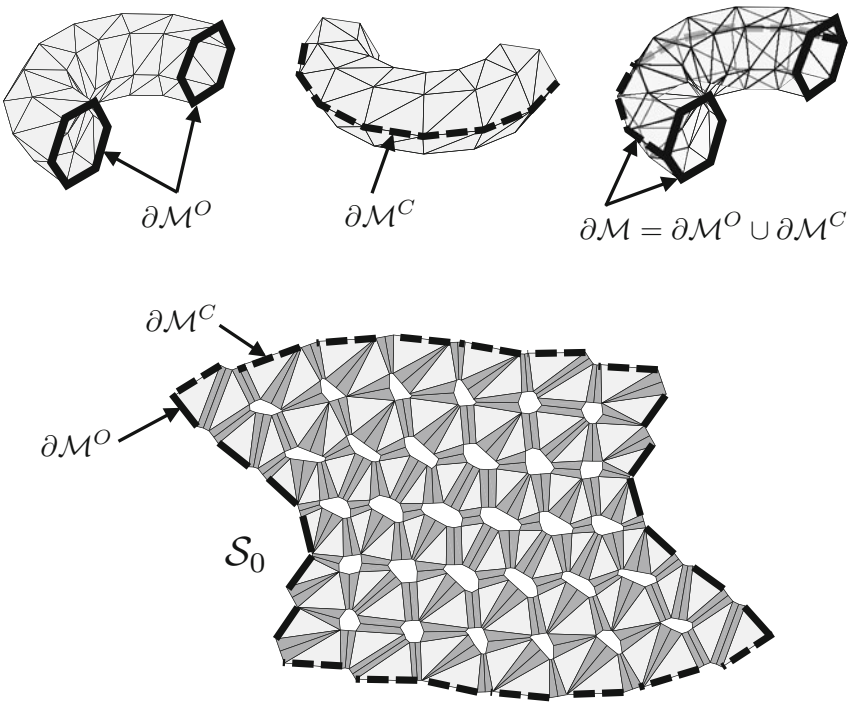


Fig. 4.4 Outer edges of the given polygonal mesh forming the boundary $\partial\mathcal{M}^O$, introduced boundary $\partial\mathcal{M}^C$, and boundary of the goal mesh $\mathcal{M}$ denoted $\partial\mathcal{M} = \partial\mathcal{M}^O \cup \partial\mathcal{M}^C$. The mapped boundary edges in the reference configuration of the designed sheet $\mathcal{S}_0$ are highlighted

boundary $\partial\mathcal{M}^C$ is not addressed here but can be found in the literature [2, 12–15]. In Problem 4.2, the reader is asked to investigate and implement two algorithms for the determination of a boundary $\partial\mathcal{M}^C$ in an arbitrary 2-manifold surface mesh such that the mesh becomes topologically equivalent to a disk.

Here, we use the same data employed in Chap. 3 to define the goal mesh $\mathcal{M}$, which is described in Sect. 3.2.2. As stated therein, the goal mesh $\mathcal{M}$ has a total of $N_\mathcal{N}$ nodes and $N_\mathcal{E}$ edges. Also, the number of interior nodes and interior edges of $\mathcal{M}$ are denoted $N_\mathcal{N}^I$ and $N_\mathcal{E}^I$, respectively. Each node of $\mathcal{M}$ has an associated *position vector* denoted $\hat{\mathbf{y}}^i \in \mathbb{R}^3$, $i = 1, \ldots, N_\mathcal{N}$. As listed in Table 3.1, the input data required to define $\mathcal{M}$ are the node position vectors $\hat{\mathbf{y}}^1, \ldots, \hat{\mathbf{y}}^{N_\mathcal{N}}$ and the *mesh connectivity matrix* $\mathbf{C}^\mathcal{M}$ defined in (3.1) that indicates the nodes associated with each mesh face.

Once a goal mesh $\mathcal{M}$ topologically equivalent to a disk has been determined in Step 1, Step 2 of the tuck-folding method entails the design of a planar origami sheet that can be folded to match $\mathcal{M}$ (refer to Sect. 4.2.1). The designed origami sheet is comprised of the mesh faces $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}}$ mapped onto a plane and $N_\mathcal{E}^I$ introduced *edge modules*, each consisting of *two rigid faces and three creased folds*

(Fig. 4.2). The edge modules are placed between every pair of faces of $\mathcal{M}$ connected by an interior edge. The challenge in this step is to determine the geometry of the edge modules such that a valid reference configuration $\mathcal{S}_0$ is obtained (refer to Sect. 2.2) and the designed sheet matches $\mathcal{M}$ in a known *goal configuration* $\mathcal{S}_\star$ (Fig. 4.3). To address this challenge, we study the parameterization of the edge modules and derive the associated constraints on the design variables in Sect. 4.2.3. In Sect. 4.2.5, we outline the numerical procedure used to solve for the design variables that satisfy the derived constraints.

4.2.3 Edge Module Parameterization and Constraints

The steps towards the determination of an origami sheet design and its subsequent kinematic analysis are illustrated in Fig. 4.5. A simple example of a goal mesh $\mathcal{M}$ with a single interior node is shown in Fig. 4.5a. We proceed by determining the geometry of the *edge modules* that are added between every pair of mesh faces joined by an interior edge[4] (Fig. 4.5b). The edge modules are designed such that the mesh faces $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}}$ together with these edge modules can be placed as a connected surface on the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$. This resulting planar surface corresponds to the sheet reference configuration $\mathcal{S}_0$ (Fig. 4.5b).

As shown in Fig. 4.5c, during folding motion of the origami sheet design, each edge module is *tuck-folded* to morph the sheet towards its goal configuration $\mathcal{S}_\star$. Note that $\mathcal{M} \subset \mathcal{S}_\star$ under appropriate rigid transformations. Different views of the tucked edge modules, which are designed such that they do not intersect in the goal configuration $\mathcal{S}_\star$, are shown in Fig. 4.5d.

We parameterize each edge module by a *length variable* denoted $\hat{W}_i$ and an *angular variable* denoted $\hat{\psi}_i$, $i = 1, \ldots, N_\mathcal{E}^I$. These variables are illustrated in Fig. 4.6. The length variable $\hat{W}_i$ is the distance between the midpoints of the outer folds of the edge module (i.e., Fold 1 and Fold 3 in Fig. 4.6) and $\hat{\psi}_i$ is the angle between the outer folds. We now continue to the derivation of constraints on the edge module variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_\mathcal{E}^I$, allowing for a valid reference configuration $\mathcal{S}_0$ and matching of $\mathcal{M}$ in a known goal configuration $\mathcal{S}_\star$.

[4] A possible extension of the proposed design method could consider the replacement of certain edge modules by single creased folds and thus allow for simplification of the designed origami sheets and reduction of the total number of folds. However, here the approach of [2, 8] is taken where an edge module is applied for each interior edge of $\mathcal{M}$ due to its wide applicability to a range of origami design problems. The aforementioned extension is strongly recommended for future studies.

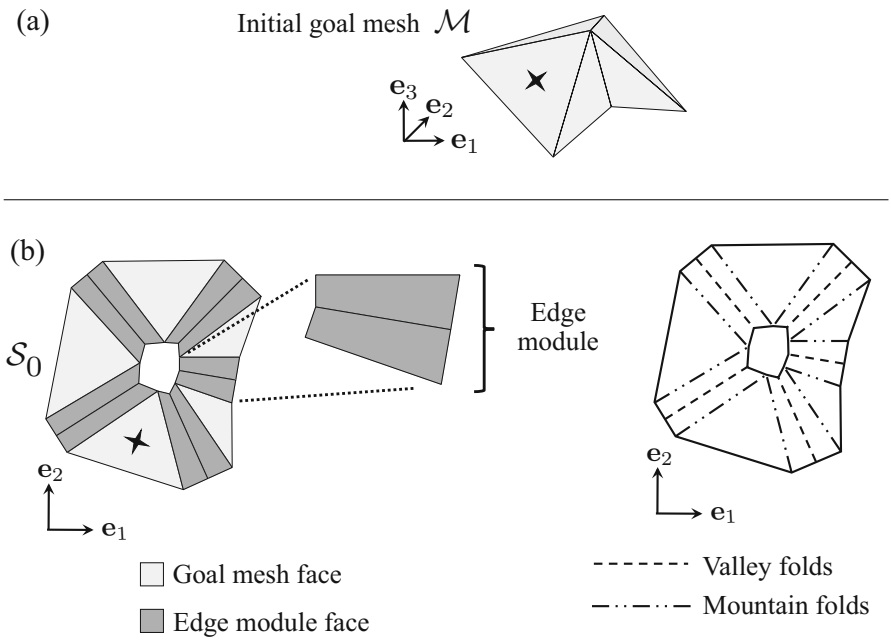


Fig. 4.5 Steps towards the determination of an origami sheet design and its subsequent kinematic analysis: (**a**) Step 1: Example goal mesh $\mathcal{M}$ topologically equivalent to disk. (**b**) Step 2: Determined sheet geometry and fold pattern associated with the goal mesh shown in (**a**). (**c**) Step 3: Folding motion from the reference planar configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$. Note that $\mathcal{M} \subset \mathcal{S}_\star$. (**d**) Different views of the goal configuration $\mathcal{S}_\star$ showing the non-intersecting tucked edge modules

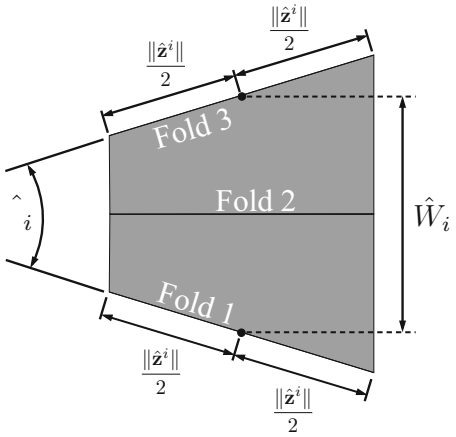


Fig. 4.6 Schematic of the ith edge module of an origami sheet design and its associated length variable $\hat{W}_i$ and angular variable $\hat{\psi}_i$. The vector along the length of the ith interior edge of $\mathcal{M}$ is denoted $\hat{\mathbf{z}}^i \in \mathbb{R}^3$ (refer to Sect. 3.2.2). Therefore, the length of such an edge is determined as $\|\hat{\mathbf{z}}^i\|$

4.2.3.1 Loop Closure Constraints

The variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, must be determined such that the faces of $\mathcal{M}$ and the edge modules associated with each interior node of $\mathcal{M}$ *form a closed loop in* $\mathcal{S}_0$ as shown in Fig. 4.5a, b. As indicated in Figs. 4.2, 4.4, 4.5, there is an interior hole in $\mathcal{S}_0$ for each interior node in $\mathcal{M}$.

Let W_{jk} and ψ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, be the variables $\hat{W}_i$ and $\hat{\psi}_i$, respectively, of the edge module associated with the kth interior edge incident to the jth interior node of $\mathcal{M}$. It is recalled from Sect. 3.2.2 that $n_j^{\mathcal{N}}$ is the number of edges incident to the jth interior node of $\mathcal{M}$. The mapping from $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, to W_{jk} and ψ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, is given as follows:

$$W_{jk} = \hat{W}_{|C_{jk}^{\mathcal{N}_I}|}, \qquad \psi_{jk} = \begin{cases} \hat{\psi}_{C_{jk}^{\mathcal{N}_I}}; & C_{jk}^{\mathcal{N}_I} > 0 \\ -\hat{\psi}_{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} < 0, \end{cases} \tag{4.2}$$

$$j = 1, \ldots, N_{\mathcal{N}}^I, \qquad k = 1, \ldots, n_j^{\mathcal{N}}.$$

where $\mathbf{C}^{\mathcal{N}_I}$ is the *node connectivity matrix* defined in (3.16).

To derive constraints on W_{jk} and ψ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, allowing the faces of $\mathcal{M}$ and the edge modules associated with the jth interior node of $\mathcal{M}$ to form a closed loop in $\mathcal{S}_0$, first let $\mathbf{b}^{j0}, \ldots, \mathbf{b}^{jn_j^{\mathcal{N}}} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the position vectors of the inner corners of the edge modules as shown in Fig. 4.7a (numbered in counterclockwise order).

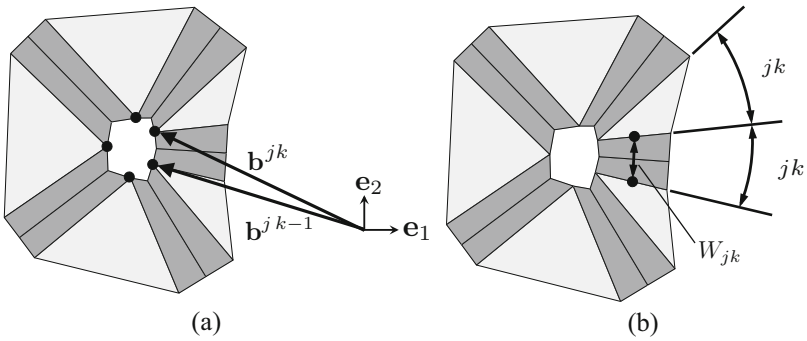


Fig. 4.7 Mesh faces and edge modules in $\mathcal{S}_0$ associated with the jth interior node of $\mathcal{M}$ (refer to Fig. 4.5b): (**a**) Position vectors of the inner corners of the edge modules; (**b**) Face corner angles ϕ_{jk} and edge module variables ψ_{jk} and W_{jk}

The position vectors $\mathbf{b}^{j0}, \ldots, \mathbf{b}^{jn_j^{\mathcal{N}}}$ can be determined recursively as follows (refer to Fig. 4.8):

$$\begin{aligned}\mathbf{b}^{jk} &= \mathbf{b}^{j\,k-1} + \mathbf{R}_3\left(\sum_{l=1}^{k-1}\left(\tfrac{\psi_{jl}}{2} + \phi_{jl} + \tfrac{\psi_{j\,l+1}}{2}\right)\right)\begin{bmatrix} W_{jk} - \|\mathbf{z}^{jk}\|\sin\left(\tfrac{\psi_{jk}}{2}\right) \\ 0 \end{bmatrix} \\ &= \sum_{l=1}^{k}\left(\mathbf{R}_3\left(\sum_{m=1}^{l-1}\left(\tfrac{\psi_{jm}}{2} + \phi_{jm} + \tfrac{\psi_{j\,m+1}}{2}\right)\right)\begin{bmatrix} W_{jl} - \|\mathbf{z}^{jl}\|\sin\left(\tfrac{\psi_{jl}}{2}\right) \\ 0 \end{bmatrix}\right) \\ &\quad + \mathbf{b}^{j0},\end{aligned} \tag{4.3}$$

where $\mathbf{R}_3(\cdot) \in \mathbb{R}^{3\times 3}$ is the matrix representing a rotation about an axis aligned with $\mathbf{e}_3$ and is defined in (2.22). The face corner angles ϕ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, are determined via (3.18).

Consider the mapping of the position vector of a point from the face with corner angle $\phi_{j\,k-1}$ to the face with corner angle ϕ_{jk}. The transformation associated with "crossing" the edge module located between these two faces can be decomposed as the following ordered transformations:

1. Translation by $\mathbf{b}^{jk} - \mathbf{b}^{j\,k-1}$. This translation is formulated in matrix form as $\mathbf{T}(\mathbf{b}^{jk} - \mathbf{b}^{j\,k-1})$, where $\mathbf{T}(\cdot)$ is defined in (2.53)
2. Rotation of $\phi_{jk} + \psi_{jk}$ about an axis aligned to $\mathbf{e}_3$ and intersecting the point with position vector $\mathbf{b}^{jk}$. This rotation is formulated in matrix form as $\mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk} + \psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{jk})$, where $\mathbf{Q}_3(\cdot)$ is defined in (2.57)

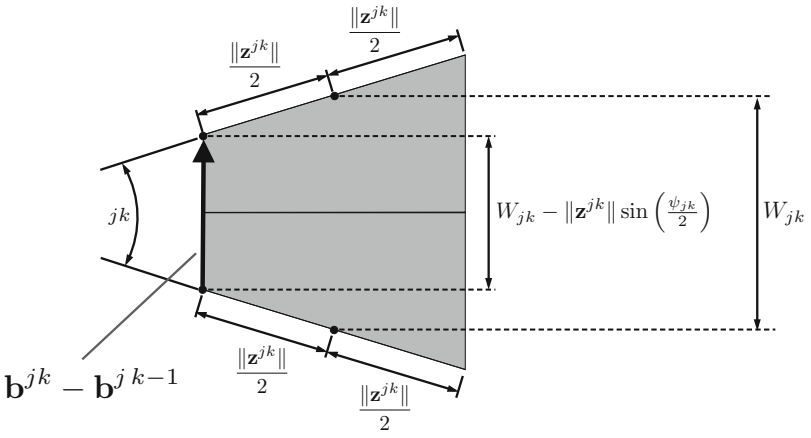


Fig. 4.8 Edge module associated with the kth edge of adjacent to the jth interior node of $\mathcal{M}$

The product of the former transformation matrices is given as follows:

$$\begin{aligned}&\mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk}+\psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{jk})\,\mathbf{T}(\mathbf{b}^{jk}-\mathbf{b}^{j\,k-1})\\&\quad=\mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk}+\psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{j\,k-1}).\end{aligned}\tag{4.4}$$

The composition of the transformations provided in (4.4) associated with crossing the edge modules with variables W_{jk}, ψ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, must be the identity transformation such that these edge modules, along with the mesh faces with corner angles ϕ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, form a closed loop:

$$\begin{aligned}\mathbf{I}_4&=\prod_{k=1}^{n_j^{\mathcal{N}}}\mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk}+\psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{j\,k-1})\\&=\mathbf{T}(\mathbf{b}^{jn_j^{\mathcal{N}}})\,\mathbf{Q}_3\left(\sum_{k=1}^{n_j^{\mathcal{N}}}(\phi_{jk}+\psi_{jk})\right)\mathbf{T}^{-1}(\mathbf{b}^{j0}).\end{aligned}\tag{4.5}$$

The condition (4.5) holds if $\sum_{k=1}^{n_j^{\mathcal{N}}}(\phi_{jk}+\psi_{jk}) = 2\pi n$ with $n \in \mathbb{Z}$.[5] Specifically, n must be equal to 1 for the closed strip to be developable [16, 17], yielding the following constraint:

[5]The set of the integer numbers is denoted as $\mathbb{Z}$.

$$2\pi = \sum_{k=1}^{n_j^{\mathcal{N}}} \phi_{jk} + \sum_{k=1}^{n_j^{\mathcal{N}}} \psi_{jk}. \tag{4.6}$$

In addition to (4.6), it is required that $\mathbf{b}^{jn_j^{\mathcal{N}}} = \mathbf{b}^{j0}$ for (4.5) to hold. Considering (4.3), $\mathbf{b}^{jn_j^{\mathcal{N}}} = \mathbf{b}^{j0}$ implies the following constraint:

$$\mathbf{0}_3 = \sum_{k=1}^{n_j^{\mathcal{N}}} \left(\mathbf{R}_3 \left(\sum_{l=1}^{k-1} \left(\frac{\psi_{jl}}{2} + \phi_{jl} + \frac{\psi_{jl+1}}{2} \right) \right) \begin{bmatrix} 0 \\ W_{jk} - \|\mathbf{z}^{jk}\| \sin\left(\frac{\psi_{jk}}{2}\right) \\ 0 \end{bmatrix} \right). \tag{4.7}$$

The vector constraint (4.7) provides two scalar constraints because the third component of such an equation is always equal to 0. Therefore, the *loop closure constraints* (4.6) and (4.7) provide a total of $3N_{\mathcal{N}}^I$ equality constraints.

Example 4.1 Formulating the loop closure constraints for the goal mesh of a pyramid shape.
Statement: Consider the goal mesh $\mathcal{M}$ of an open pyramid shown in Fig. 3.9. Formulate the constraints (4.6) and (4.7) associated with the single interior node of such a goal mesh.
Solution: The number of edges adjacent to the single interior node of the goal mesh $\mathcal{M}$ shown in Fig. 3.9 is 4 (i.e., $n_1^{\mathcal{N}} = 4$). The face corner angles $\phi_{11}, \ldots, \phi_{14}$ are determined in (3.21). Thus, the constraint (4.6) is formulated as follows:

$$\begin{aligned} 2\pi &= \sum_{k=1}^{4} \phi_{1k} + \sum_{k=1}^{4} \psi_{1k}, \\ &= 2\cos^{-1}\left(\frac{L_1^2 - L_2^2 + L_3^2}{L_1^2 + L_2^2 + L_3^2} \right) + 2\cos^{-1}\left(\frac{-L_1^2 + L_2^2 + L_3^2}{L_1^2 + L_2^2 + L_3^2} \right) + \sum_{k=1}^{4} \psi_{1k}. \end{aligned} \tag{4.8}$$

The vectors $\mathbf{z}^{11}, \ldots, \mathbf{z}^{14}$ along the length of the edges incident to the interior node of $\mathcal{M}$ are determined in (3.20). The magnitude of these vectors is given as follows:

$$\|\mathbf{z}^{11}\| = \|\mathbf{z}^{12}\| = \|\mathbf{z}^{13}\| = \|\mathbf{z}^{14}\| = \left(L_1^2 + L_2^2 + L_3^2 \right)^{\frac{1}{2}}. \tag{4.9}$$

Using such information, the constraint (4.7) is formulated as follows:

$$\mathbf{0}_3 = \sum_{k=1}^{4} \left(\mathbf{R}_3 \left(\sum_{l=1}^{k-1} \left(\frac{\psi_{1l}}{2} + \phi_{1l} + \frac{\psi_{1\,l+1}}{2} \right) \right) \begin{bmatrix} W_{1k} - \|\mathbf{z}^{1k}\| \sin\left(\frac{\psi_{1k}}{2}\right) \\ 0 \\ 0 \end{bmatrix} \right).$$

$$= \begin{bmatrix} W_{11} - \left(L_1^2 + L_2^2 + L_3^2\right)^{\frac{1}{2}} \sin\left(\frac{\psi_{11}}{2}\right) \\ 0 \\ 0 \end{bmatrix}$$

$$+ \mathbf{R}_3 \left(\frac{\psi_{11}}{2} + \phi_{11} + \frac{\psi_{12}}{2} \right) \begin{bmatrix} W_{12} - \left(L_1^2 + L_2^2 + L_3^2\right)^{\frac{1}{2}} \sin(\frac{\psi_{12}}{2}) \\ 0 \\ 0 \end{bmatrix}$$

$$+ \mathbf{R}_3 \left(\sum_{l=1}^{2} \left(\frac{\psi_{1l}}{2} + \phi_{1l} + \frac{\psi_{1\,l+1}}{2} \right) \right) \begin{bmatrix} W_{13} - \left(L_1^2 + L_2^2 + L_3^2\right)^{\frac{1}{2}} \sin(\frac{\psi_{13}}{2}) \\ 0 \\ 0 \end{bmatrix}$$

$$+ \mathbf{R}_3 \left(\sum_{l=1}^{3} \left(\frac{\psi_{1l}}{2} + \phi_{1l} + \frac{\psi_{1\,l+1}}{2} \right) \right) \begin{bmatrix} W_{14} - \left(L_1^2 + L_2^2 + L_3^2\right)^{\frac{1}{2}} \sin(\frac{\psi_{14}}{2}) \\ 0 \\ 0 \end{bmatrix}. \tag{4.10}$$

where the expressions for the face corner angles $\phi_{11}, \ldots, \phi_{14}$ are provided in (3.21).

4.2.3.2 Constraints for Valid Edge Module Geometry

In addition to the loop closure constraints provided in (4.6) and (4.7), other constraints must be imposed to ensure that the geometry of each individual edge module is valid. The faces of an edge module degenerate to straight line segments if the angular variable $\hat{\psi}_i$ reaches $\pm\pi$ (refer to Fig. 4.6). Therefore, the following bounds must be imposed for this variable:

$$\boxed{-\pi < \hat{\psi}_i < \pi,} \tag{4.11}$$

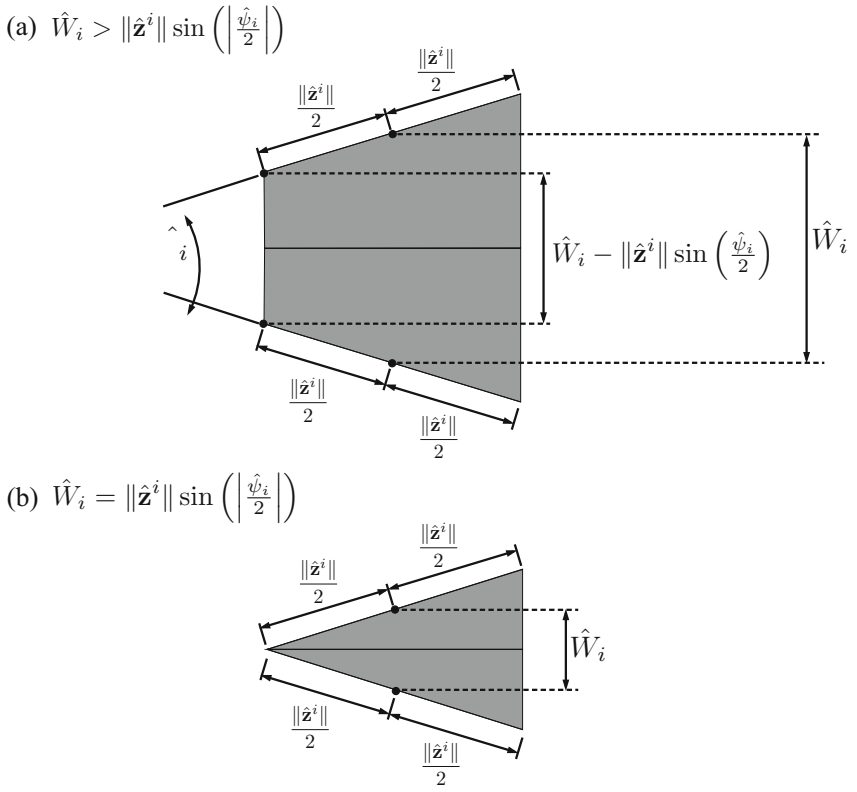


Fig. 4.9 Two valid edge module geometries for which the folds at the boundary of the edge module do not overlap its interior fold: (**a**) $\hat{W}_i > \|\hat{\mathbf{z}}^i\| \sin(|\frac{\hat{\psi}_i}{2}|)$; (**b**) $\hat{W}_i = \|\hat{\mathbf{z}}^i\| \sin(|\frac{\hat{\psi}_i}{2}|)$

where $i = 1, \ldots, N_{\mathcal{E}}^I$. Furthermore, the folds at the boundary of each edge module must not overlap with the interior fold. This requirement yields the following constraint (refer to Fig. 4.9):

$$\boxed{\hat{W}_i \geq \left\|\hat{\mathbf{z}}^i\right\| \sin\left(\left|\frac{\hat{\psi}_i}{2}\right|\right).} \tag{4.12}$$

where $i = 1, \ldots, N_{\mathcal{E}}^I$. The constraints (4.11) and (4.12) provide $3N_{\mathcal{E}}^I$ inequality constraints.

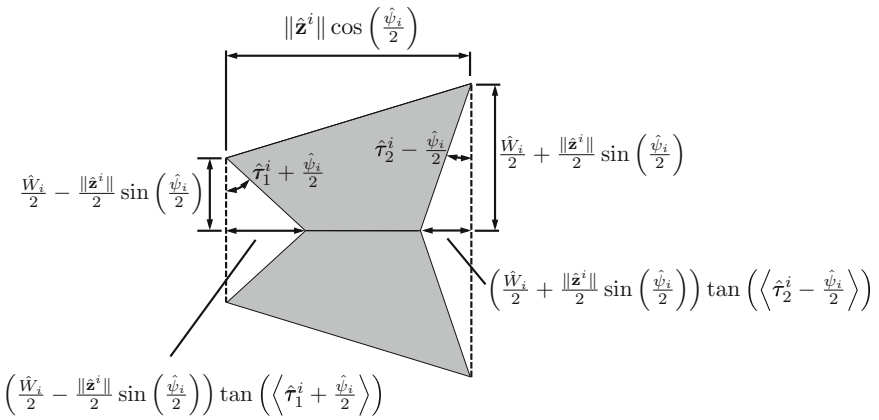


Fig. 4.10 Edge module trimmed to account for self-intersection avoidance in the goal configuration $\mathcal{S}_\star$

4.2.3.3 Constraints to Prevent Intersections Among Tuck-Folded Edge Modules

Intersections between regions of the sheet are not allowed in valid configurations (see Sect. 2.2) and therefore must be avoided when adjacent edge modules are tucked in the goal configuration $\mathcal{S}_\star$ (see Fig. 4.11d). For this purpose, certain regions of the edge modules are *trimmed* such that each edge module does not intersect any of its neighboring edge modules in $\mathcal{S}_\star$. This procedure is called *edge module trimming* and is addressed in Sect. 4.2.4. The *trim angles* $\hat{\tau}_1^i$ and $\hat{\tau}_2^i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, are used to determine the trimmed regions of the ith edge module as shown in Fig. 4.10. If the edge module trimming procedure is not considered and the edge module geometry is as illustrated in Fig. 4.6, adjacent edge modules intersect at $\mathcal{S}_\star$ as shown in Fig. 4.11d. On the other hand, if the edge module trimming procedure is considered and the geometry of the edge modules is as illustrated in Fig. 4.10, intersections between adjacent edge modules are not present as shown in Fig. 4.11g.

As previously mentioned, the angles $\hat{\tau}_1^i$ and $\hat{\tau}_2^i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, are introduced to determine the trimmed regions of the ith edge module. These angles are computed using the procedure described in Sect. 4.2.4. Each edge module *must remain connected after edge module trimming*. This requirement is satisfied if the following inequality holds for each edge module (refer to Fig. 4.10)[6]:

[6]The Macaulay brackets are denoted as $\langle \cdot \rangle$ and defined as: $\langle y \rangle = \begin{cases} y; & y \geq 0 \\ 0; & y < 0 \end{cases}$.

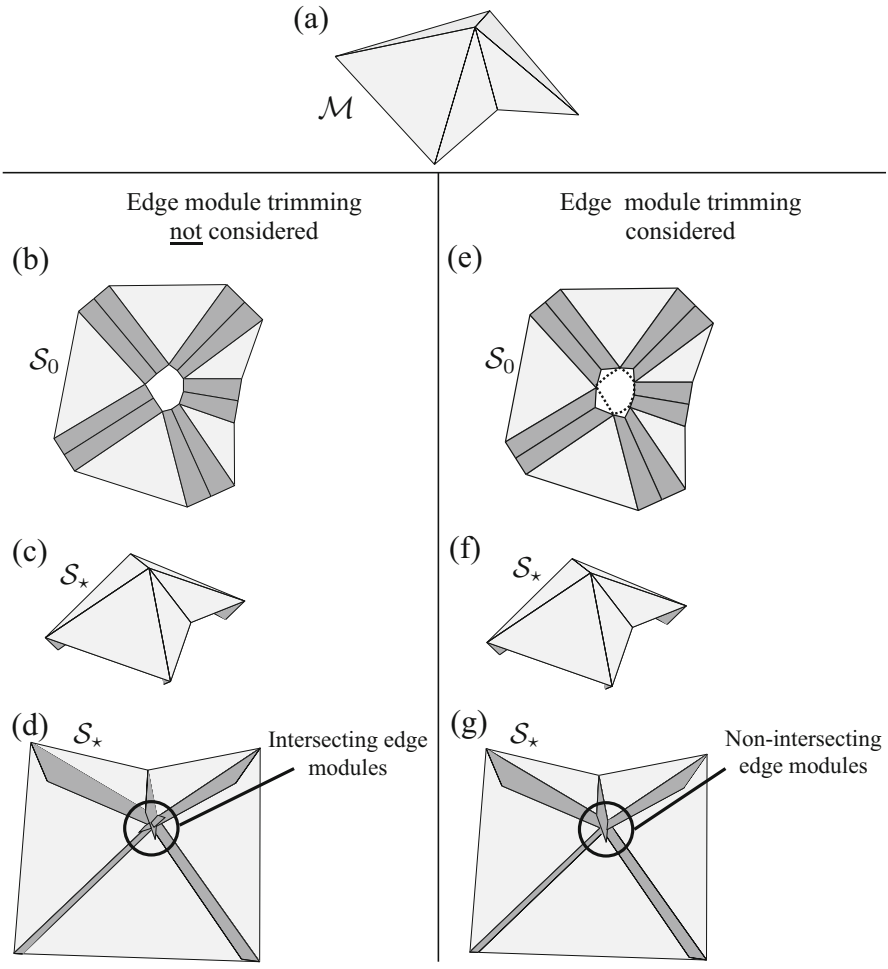


Fig. 4.11 Example showing the need for edge module trimming. (**a**) A simple goal mesh $\mathcal{M}$. (**b**) and (**e**) Determined origami sheet designs. Edge module trimming is *not* considered for the origami sheet design shown in (**b**) while such a procedure is considered in (**e**). (**c**)–(**d**), (**f**)–(**g**) Views of the goal configuration $\mathcal{S}_\star$ for both cases. It is observed in (**d**) that the tuck-folded edge modules intersect in $\mathcal{S}_\star$ if the edge module trimming procedure is not considered. The edge modules do not intersect in $\mathcal{S}_\star$ if such a procedure is considered as shown in (**g**)

$$\begin{aligned}\left\|\hat{\mathbf{z}}^i\right\| \cos\left(\frac{\hat{\psi}_i}{2}\right) > &\left(\frac{\hat{W}_i}{2} - \frac{\|\hat{\mathbf{z}}^i\|}{2}\sin\left(\frac{\hat{\psi}_i}{2}\right)\right)\tan\left(\left\langle\hat{\tau}_1^i + \frac{\hat{\psi}_i}{2}\right\rangle\right) \\ &+ \left(\frac{\hat{W}_i}{2} + \frac{\|\hat{\mathbf{z}}^i\|}{2}\sin\left(\frac{\hat{\psi}_i}{2}\right)\right)\tan\left(\left\langle\hat{\tau}_2^i - \frac{\hat{\psi}_i}{2}\right\rangle\right),\end{aligned} \tag{4.13}$$

where $i = 1, \ldots, N_{\mathcal{E}}^I$. It is noted that if $(\hat{\tau}_1^i + \hat{\psi}_i/2) < 0$ or $(\hat{\tau}_2^i - \hat{\psi}_i/2) < 0$, these angles are set to 0 to prevent any increase in area of the edge modules during this process. The preceding equation provides $N_{\mathcal{E}}^I$ additional inequality constraints.

4.2.3.4 Summary of Design Constraints

The proposed design method introduces $2N_{\mathcal{E}}^I$ *design variables* corresponding to $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$. The loop closure constraints (4.6) and (4.7) provide $3N_{\mathcal{N}}^I$ *equality constraints*. The constraints (4.11), (4.12), and (4.13) allowing for valid edge module geometries and self-intersection avoidance in $\mathcal{S}_\star$ provide $4N_{\mathcal{E}}^I$ *inequality constraints*. In Sect. 4.2.5, we describe a numerical procedure used to determine a set of design parameters that satisfies the aforementioned equality and inequality constraints.

4.2.4 Edge Module Trimming

As stated in Sect. 4.2.3, intersections among regions of the sheet are not allowed in valid configurations and therefore must be prevented when adjacent edge modules are tucked in the goal configuration $\mathcal{S}_\star$. Thus, certain regions of the edge modules are *trimmed* such that each edge module does not intersect any of its neighboring edge modules as shown in Fig. 4.11. The angles $\hat{\tau}_1^i$ and $\hat{\tau}_2^i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, are used to define the trimmed regions of the ith edge module as shown in Fig. 4.10. The formulation used to determine these angles is summarized in this section.

Let τ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, be the trim angle associated with the edge module placed at the kth interior edge adjacent to the jth interior node of $\mathcal{M}$. The trim angles $\hat{\tau}_1^i$ and $\hat{\tau}_2^i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, are determined from their corresponding values in the set of angles τ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$ by inverting the following map:

$$\tau_{jk} = \begin{cases} \hat{\tau}_1^{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} > 0 \\ \hat{\tau}_2^{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} < 0, \end{cases} \tag{4.14}$$

$$j = 1, \ldots, N_{\mathcal{N}}^I, \qquad k = 1, \ldots, n_j^{\mathcal{N}},$$

where $\mathbf{C}^{\mathcal{N}_I}$ is the *node connectivity matrix* defined in (3.16).

Configurations of an edge module in the reference configuration $\mathcal{S}_0$ and the goal configuration $\mathcal{S}_\star$ are illustrated in Fig. 4.12a. To determine the trim angle τ_{jk} shown in this figure, we first compute the unit normal vector associated with the kth edge incident to the jth interior node of $\mathcal{M}$ denoted $\bar{\mathbf{n}}^{jk} \in \mathbb{R}^3$:

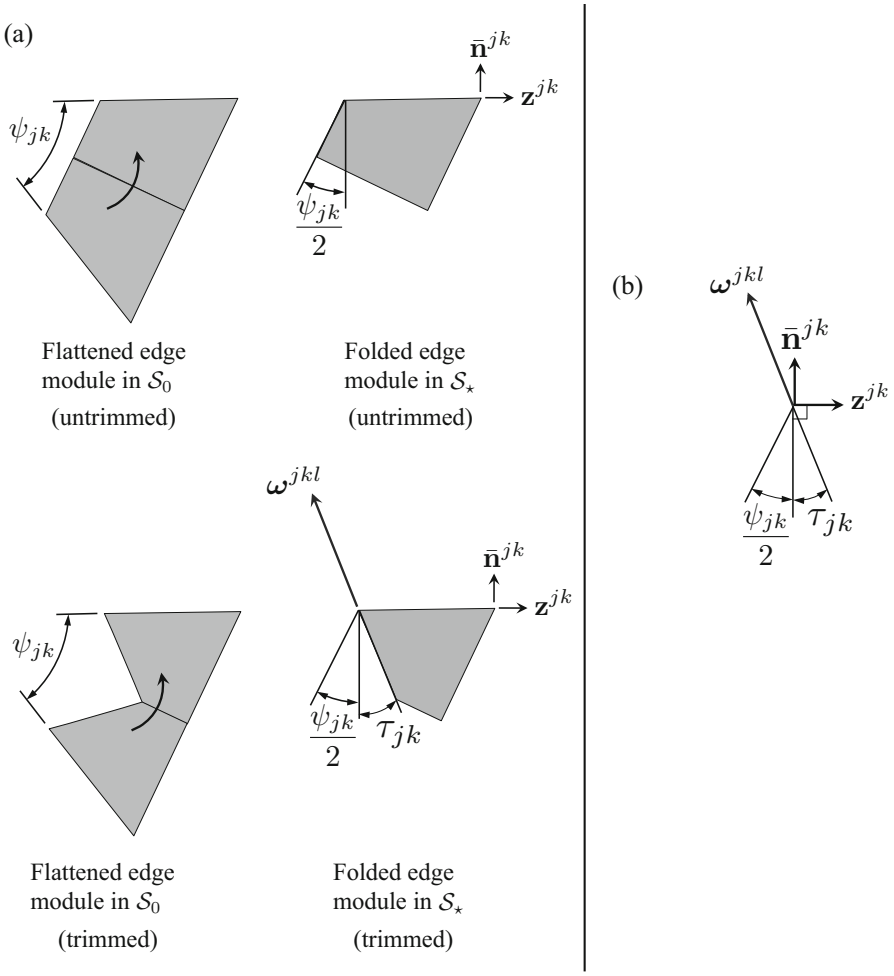


Fig. 4.12 (**a**) Configurations of an edge module in the reference configuration $\mathcal{S}_0$ (left) and the goal configuration $\mathcal{S}_\star$ (right). (**b**) View of the plane spanned by $\bar{\mathbf{n}}^{jk}$ and $\mathbf{z}^{jk}$ showing the intersection axis along the vector $\boldsymbol{\omega}^{jkl}$ and the trim angle τ_{jk} determined via (4.16) and (4.19), respectively

$$\bar{\mathbf{n}}^{jk} = \frac{\mathbf{n}^{j\,k-1} + \mathbf{n}^{jk}}{\|\mathbf{n}^{j\,k-1} + \mathbf{n}^{jk}\|}, \tag{4.15}$$

where the face unit normal vectors $\mathbf{n}^{jk}$, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, are defined in (3.19). The rigid faces of the tuck-folded edge modules are located in the plane spanned by $\mathbf{z}^{jk}$ and $\bar{\mathbf{n}}^{jk}$ as indicated in Fig. 4.12a.

The direction of the *intersection axis* between any two planes containing adjacent tucked edge modules is defined by the unit vector $\boldsymbol{\omega}^{jkl} \in \mathbb{R}^3$, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, $l = 1, \ldots, n_j^{\mathcal{N}}$, $k \neq l$ (refer to Figs. 4.12b and 4.13):

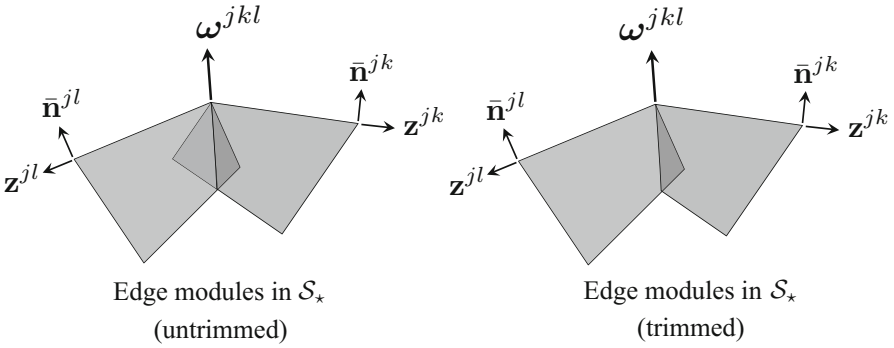


Fig. 4.13 Two tuck-folded edge modules in the goal configuration $\mathcal{S}_\star$ (untrimmed and trimmed forms). The tuck-folded edge modules shown in this figure are *not* located in the same plane. The vector $\boldsymbol{\omega}^{jkl}$ provides the direction of their intersection axis. As indicated in the figure, we only need to trim one of the two edge modules to preclude self-intersections between them. The edge module that is trimmed between each pair of adjacent edge modules is determined via (4.18)

$$\boldsymbol{\omega}^{jkl} = \begin{cases} \dfrac{(\bar{\mathbf{n}}^{jk} \times \mathbf{z}^{jk}) \times (\bar{\mathbf{n}}^{jl} \times \mathbf{z}^{jl})}{\|(\bar{\mathbf{n}}^{jk} \times \mathbf{z}^{jk}) \times (\bar{\mathbf{n}}^{jl} \times \mathbf{z}^{jl})\|}; & (\bar{\mathbf{n}}^{jk} \times \mathbf{z}^{jk}) \times (\bar{\mathbf{n}}^{jl} \times \mathbf{z}^{jl}) \neq 0 \\ \bar{\mathbf{n}}^{jl}; & (\bar{\mathbf{n}}^{jk} \times \mathbf{z}^{jk}) \times (\bar{\mathbf{n}}^{jl} \times \mathbf{z}^{jl}) = 0. \end{cases} \tag{4.16}$$

As assumed in (4.16), the vector $\bar{\mathbf{n}}^{jl}$ is utilized to define $\boldsymbol{\omega}^{jkl}$ in the case where the rigid faces of the kth and the lth tuck-folded edge modules are located in the same plane.

The angles τ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^{I}$, $k = 1, \ldots, n_j^{\mathcal{N}}$, are determined through the following equations:

$$\tau_{jkl}^{(1)} = \begin{cases} 0; & \bar{\mathbf{n}}^{jk} \cdot \mathbf{z}^{jl} > 0 \text{ and } \bar{\mathbf{n}}^{jl} \cdot \mathbf{z}^{jk} > 0 \\ -\dfrac{\pi}{2} + \begin{cases} \cos^{-1}\left(\dfrac{\mathbf{z}^{jk}}{\|\mathbf{z}^{jk}\|} \cdot \boldsymbol{\omega}^{jkl}\right); & \bar{\mathbf{n}}^{jk} \cdot \boldsymbol{\omega}^{jkl} \geq 0 \\ \cos^{-1}\left(-\dfrac{\mathbf{z}^{jk}}{\|\mathbf{z}^{jk}\|} \cdot \boldsymbol{\omega}^{jkl}\right); & \bar{\mathbf{n}}^{jk} \cdot \boldsymbol{\omega}^{jkl} < 0, \end{cases} & \text{otherwise} \end{cases} \tag{4.17}$$

$$\tau_{jkl}^{(2)} = \begin{cases} \tau_{jkl}^{(1)}; & \tau_{jkl}^{(1)} \leq \tau_{jlk}^{(1)} \\ 0; & \tau_{jkl}^{(1)} > \tau_{jlk}^{(1)}, \end{cases} \tag{4.18}$$

$$\tau_{jk} = \max_{l \in \left\{1, \ldots, n_j^{\mathcal{N}}\right\},\ l \neq k} \left(\tau_{jkl}^{(2)}\right). \tag{4.19}$$

From (4.17), it is noted that if $\bar{\mathbf{n}}^{jk} \cdot \mathbf{z}^{jl} > 0$ and $\bar{\mathbf{n}}^{jl} \cdot \mathbf{z}^{jk} > 0$, the angles $\tau_{jkl}^{(1)}$ are set to 0 as no further trimming is required in such cases. In (4.18), the lower trim angle required to prevent intersections between each pair of edge modules is selected while the larger is set to zero. In (4.19), we determine the trim angle τ_{jk} for an edge module required to prevent intersections with all of its adjacent edge modules, corresponding to the maximum value among $\tau_{jkl}^{(2)}$, $l = 1, \ldots, n_j^{\mathcal{N}}$, $l \neq k$. In Problem 4.11, the reader is asked to calculate the trim angles τ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, for the goal mesh $\mathcal{M}$ illustrated in Fig. 3.9.

4.2.5 *Determination of Design Variables*

A numerical procedure for the determination of the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, subject to the equality constraints (4.6) and (4.7) and inequality constraints (4.11)–(4.13) is described in this section. After the values for these variables are determined, the full geometry of the origami sheet design can be generated.

Given the input parameters defining the goal mesh $\mathcal{M}$ listed in Table 3.1, we can compute the geometric parameters employed in the constraints derived in Sect. 4.2.3. Afterwards, the numerical procedure presented in this section is employed to determine the values of the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, that satisfy the constraints derived in Sect. 4.2.3.

Let $\mathbf{d} \in \mathbb{R}^{2N_{\mathcal{E}}^I}$ be the vector whose components correspond to the design variables:

$$\mathbf{d} = \begin{bmatrix} \hat{W}_1 & \cdots & \hat{W}_{N_{\mathcal{E}}^I} & \hat{\psi}_1 & \cdots & \hat{\psi}_{N_{\mathcal{E}}^I} \end{bmatrix}^\top. \tag{4.20}$$

The equality constraints (4.6) and (4.7) are set to the form $\boldsymbol{h} = \mathbf{0}_{3N_{\mathcal{N}}^I}$ while the inequality constraints (4.11), (4.12), and (4.13) are set to the form $\boldsymbol{g} \leq \mathbf{0}_{4N_{\mathcal{E}}^I}$.[7] Then, the problem statement for the determination of $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, can be expressed as follows:

[7] For numerical implementation, strict inequality constraints of the form $A < 0$ in (4.11) and (4.13) are expressed as non-strict inequalities of the form $A + \epsilon \leq 0$ where $\epsilon > 0$.

$$\begin{array}{ll} \text{Find} & \mathbf{d} \\ \text{That minimizes} & f(\mathbf{d}) \\ \text{Subject to} & \boldsymbol{h}(\mathbf{d}) = \mathbf{0}_{3N_{\mathcal{N}}^I} \\ & \boldsymbol{g}(\mathbf{d}) \leq \mathbf{0}_{4N_{\mathcal{E}}^I} \\ & \text{and application-dependent constraints,} \end{array} \tag{4.21}$$

where $f(\mathbf{d})$ is an application-dependent function of the variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$.

The numerical procedure employed here to determine $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, does not account for any function $f(\mathbf{d})$ (i.e., the procedure only iteratively corrects an initial guess solution until the constraints $\boldsymbol{h} = \mathbf{0}_{3N_{\mathcal{N}}^I}$ and $\boldsymbol{g} \leq \mathbf{0}_{4N_{\mathcal{E}}^I}$ are satisfied). The reader is asked in Problem 4.5 to propose and implement a procedure to determine the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, by solving the design problem stated in (4.21) where a function $f(\mathbf{d})$ is considered.

Let $\mathfrak{R} \in \mathbb{R}^{3N_{\mathcal{N}}^I + 4N_{\mathcal{E}}^I}$ be the *residual vector* from equality and inequality constraints defined as follows:

$$\mathfrak{R}(\mathbf{d}) = \begin{bmatrix} \boldsymbol{h}(\mathbf{d}) \\ \max(\mathbf{0}_{4N_{\mathcal{E}}^I}, \ \boldsymbol{g}(\mathbf{d})) \end{bmatrix}, \tag{4.22}$$

where the $\max(\cdot, \cdot)$ operator in the previous equation is applied component-wise. In the present numerical procedure, an initial guess for the design variables denoted $\mathbf{d}^{(0)}$ is provided as input. Then, the design variables are iteratively corrected until the derived constraints are satisfied. At iteration (l), if $\|\mathfrak{R}(\mathbf{d}^{(l)})\|/(3N_{\mathcal{N}}^I + 4N_{\mathcal{E}}^I) \geq \texttt{tol}$ (where `tol` is a numerical tolerance), the design variables are corrected using the generalized Newton's method as follows:

$$\Delta\mathbf{d}^{(l)} = -\left(\frac{\partial\mathfrak{R}(\mathbf{d}^{(l)})}{\partial\mathbf{d}}\right)^{\dagger} \mathfrak{R}(\mathbf{d}^{(l)}), \tag{4.23}$$

$$\mathbf{d}^{(l+1)} = \mathbf{d}^{(l)} + \Delta\mathbf{d}^{(l)}, \tag{4.24}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse (refer to (A.37)). Given an initial guess $\mathbf{d}^{(0)}$, the design variables are iteratively corrected as indicated in (4.23) and (4.24) until $\|\mathfrak{R}(\mathbf{d}^{(l)})\|/(3N_{\mathcal{N}}^I + 4N_{\mathcal{E}}^I) < \texttt{tol}$. Table 4.1 summarizes the procedure used to determine a set of design variables that satisfies the constraints of the tuck-folding method for origami design.

Table 4.1 Numerical procedure used to compute the values of the design variables that satisfy the constraints of the tuck-folding method for origami design

1:	Set $l \leftarrow 0$ and provide initial guess $\mathbf{d}^{(0)}$
2:	Determine $\|\mathfrak{R}(\mathbf{d}^{(l)})\|$
3:	`IF` $\|\mathfrak{R}(\mathbf{d}^{(l)})\|/(3N_{\mathcal{N}}^{I} + 4N_{\mathcal{E}}^{I}) <$ `tol THEN RETURN` $\mathbf{d}^{(l)}$ and `EXIT ELSE CONTINUE`
4:	Determine $\mathbf{d}^{(l+1)}$ using (4.23) and (4.24)
5:	Set $l \leftarrow l + 1$ and `GOTO` 2

4.2.6 Determination of Folding Motion

Following the steps of the tuck-folding method listed in Sect. 4.2.1, once an origami sheet design has been obtained for a given goal mesh $\mathcal{M}$, the next step is to determine a history of folding motion from the computed reference configuration $\mathcal{S}_0$ towards the *goal configuration* $\mathcal{S}_\star$ that matches $\mathcal{M}$. The determination of such a history of folding motion is addressed in this section.

We use the model for origami with creased folds presented in Chap. 2 to simulate the folding motion of the sheets determined via the tuck-folding method. As explained in Sect. 2.6, the folding motion is simulated by *incrementally* updating the values of the fold angles using guess increments and then applying any required corrections such that every folded configuration is valid as described in Sect. 2.2.

We define $\Delta\hat{\boldsymbol{\theta}}^{l} \in \mathbb{R}^{3N_{\mathcal{E}}^{I}}$ as the vector whose components correspond to the lth set of guess increments for the fold angles (see (2.72)):

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \begin{bmatrix} \Delta\hat{\theta}_1^{l} & \cdots & \Delta\hat{\theta}_{3N_{\mathcal{E}}^{I}}^{l} \end{bmatrix}^{\top} \qquad l = 1, \ldots, N_{\text{inc}}. \tag{4.25}$$

where N_{inc} is the *total number of increments*. The guess increments for the fold angles $\Delta\hat{\boldsymbol{\theta}}^{l}, l = 1, \ldots, N_{\text{inc}}$, must be determined from the given goal mesh $\mathcal{M}$. For this purpose, first let $\hat{\theta}_1^{\star}, \ldots, \hat{\theta}_{3N_{\mathcal{E}}^{I}}^{\star}$ be the fold angles of each creased fold in the goal configuration $\mathcal{S}_\star$ (i.e., the *goal fold angles*). To determine the goal fold angles, the *dihedral angle* $\hat{\Theta}_i$ between the two faces in $\mathcal{M}$ connected by the edge associated with the ith edge module is calculated as follows (see Fig. 4.14):

$$\hat{\Theta}_i = \begin{cases} \pi + \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for convex edges} \\ \\ \pi - \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for concave edges,} \end{cases} \tag{4.26}$$

$$i = 1, \ldots, N_{\mathcal{E}}^{I}.$$

where $\hat{\mathbf{n}}^{i1}, \hat{\mathbf{n}}^{i2} \in \mathbb{R}^3$ are the unit normal vectors in $\mathcal{M}$ of the two faces connected by the edge associated with the ith edge module. As shown in Fig. 4.14, the goal

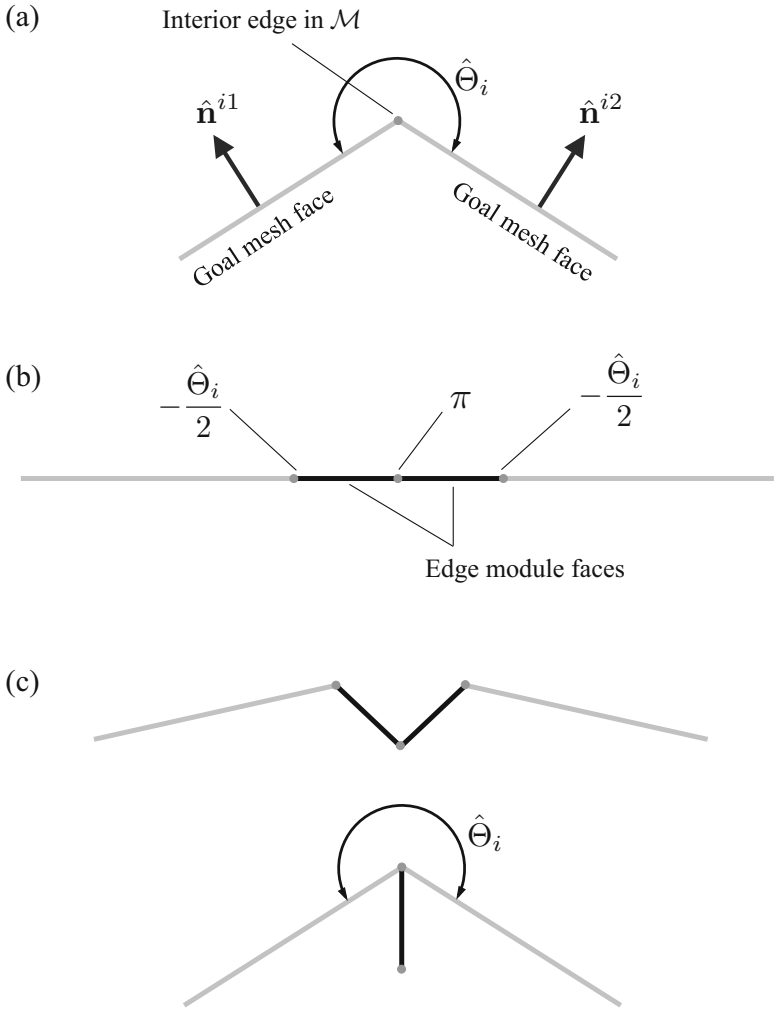


Fig. 4.14 (**a**) Side view of the faces of the goal mesh $\mathcal{M}$ connected by an interior edge. The edge dihedral angle $\hat{\Theta}_i$ and unit normal vectors $\hat{\mathbf{n}}^{i1}$ and $\hat{\mathbf{n}}^{i2}$ of the adjacent faces are shown. (**b**) Configuration of such mesh faces and the edge module that connects them in the reference configuration $\mathcal{S}_0$ of the origami sheet design. The goal fold angles of each creased fold are indicated. (**c**) Folding motion of the faces shown in (**b**) towards their goal configuration

fold angle for the interior fold of each edge module is equal to π while those for the exterior folds are equal to $-\hat{\Theta}_i/2$. We define $\hat{\boldsymbol{\theta}}^\star \in \mathbb{R}^{3N_\mathcal{E}^I}$ as the vector whose components are the goal fold angles:

$$\hat{\boldsymbol{\theta}}^\star = \begin{bmatrix} \hat{\theta}_1^\star & \cdots & \hat{\theta}_{3N_\mathcal{E}^I}^\star \end{bmatrix}^\top . \tag{4.27}$$

The simulation of the folding motion of the determined origami sheet designs uses a "fold and adjust" approach and is performed in $N_{\text{fol}} + N_{\text{adj}}$ increments (i.e., $N_{\text{inc}} = N_{\text{fol}} + N_{\text{adj}}$). The N_{fol} guess increments for the fold angles are given as follows:

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \frac{\hat{\boldsymbol{\theta}}^{\star}}{N_{\text{fol}}} \qquad l = 1, \ldots, N_{\text{fol}}. \tag{4.28}$$

Since the set of fold angles at each increment is subject to iterative corrections to ensure that the kinematic constraints derived in Problem 4.1 are met for every interior fold intersection, the configuration determined at the N_{fol} increment may not exactly correspond to $\mathcal{S}_{\star}$. Therefore, N_{adj} *adjusting increments* are subsequently applied. These adjusting fold angle increments are calculated as follows:

$$\Delta\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{\star} - \hat{\boldsymbol{\theta}}^{l-1} \qquad l = N_{\text{fol}} + 1, \ldots, N_{\text{fol}} + N_{\text{adj}}, \tag{4.29}$$

where the components of $\hat{\boldsymbol{\theta}}^{l-1}$ correspond to the values of the fold angles at the increment $l - 1$. The simple fold angle increments provided in (4.28) and (4.29) are effective for the determination of a folding motion from the reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_{\star}$ in all the examples presented in this chapter. However, it is not guaranteed to work for any arbitrary goal mesh or sheet reference configuration. For information on more complex motion planning procedures for origami, the reader is referred to [18–23].

4.2.7 Design Requirements of the Tuck-Folding Method

An expanded discussion of the tuck-folding method for origami design is provided here. First, let us consider the *number of degrees of freedom* (N_{dof}) in the present origami design problem. Such a number of degrees of freedom is given as follows:

$$N_{\text{dof}} = 2N_{\mathcal{E}}^{I} - 3N_{\mathcal{N}}^{I}, \tag{4.30}$$

where $2N_{\mathcal{E}}^{I}$ is the number of design variables corresponding to $\hat{W}_i$, $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^{I}$, and $3N_{\mathcal{N}}^{I}$ is the number of equality constraints from (4.6) and (4.7). It is assumed in (4.30) that there are no redundant equality constraints and that the inequality constraints (4.11)–(4.13) permit solutions to the equality constraints (4.6) and (4.7). The number of interior edges $N_{\mathcal{E}}^{I}$ can be additively decomposed as follows:

$$N_{\mathcal{E}}^{I} = N_{\mathcal{E}}^{I0} + N_{\mathcal{E}}^{I1} + N_{\mathcal{E}}^{I2}, \tag{4.31}$$

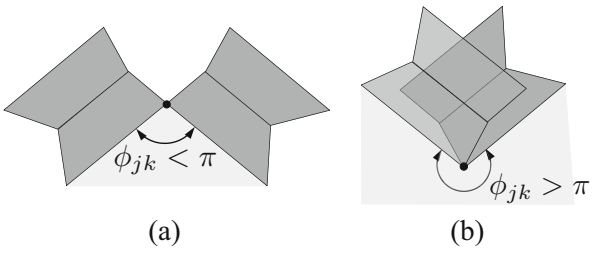


Fig. 4.15 (**a**) Edge modules connected to a face with a convex face corner angle. (**b**) Edge modules connected to a face with a concave face corner angle

where $N_{\mathcal{E}}^{Im}$, $m = 0, 1, 2$, is the number of interior edges connected to m interior nodes. Also consider the following equality:

$$\bar{n}^{\mathcal{N}} N_{\mathcal{N}}^{I} = \sum_{j=1}^{N_{\mathcal{N}}^{I}} n_j^{\mathcal{N}} = 2N_{\mathcal{E}}^{I2} + N_{\mathcal{E}}^{I1}, \tag{4.32}$$

where $\bar{n}^{\mathcal{N}}$ is the average number of edges incident to each interior node in $\mathcal{M}$. Substituting (4.31) and (4.32) into (4.30), the following relation between the number of degrees of freedom and the number of interior edges is obtained:

$$N_{\text{dof}} = \left(2 - \frac{6}{\bar{n}^{\mathcal{M}}}\right) N_{\mathcal{E}}^{I2} + \left(2 - \frac{3}{\bar{n}^{\mathcal{M}}}\right) N_{\mathcal{E}}^{I1} + 2N_{\mathcal{E}}^{I0}. \tag{4.33}$$

For example, a structured triangulated mesh with $\bar{n}^{\mathcal{M}} = 6$ has $N_{\text{dof}} = N_{\mathcal{E}}^{I2} + (3/2)N_{\mathcal{E}}^{I1} + 2N_{\mathcal{E}}^{I0}$ which is always greater than 0 and a structured quadrilateral mesh with $\bar{n}^{\mathcal{M}} = 4$ has $N_{\text{dof}} = (1/2)N_{\mathcal{E}}^{I2} + (5/4)N_{\mathcal{E}}^{I1} + 2N_{\mathcal{E}}^{I0}$ which is also always greater than 0. Therefore, the existence of a set of design variables $\hat{W}_i$, $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^{I}$, that satisfies the design constraints is generally dependent on the inequality constraints (4.11)–(4.13).

A requirement in the tuck-folding method is that the determined sheet design must not have any overlapping regions in its reference configuration $\mathcal{S}_0$ (see Sect. 2.2). To prevent overlaps of edge modules in $\mathcal{S}_0$ connected to a common interior node of $\mathcal{M}$, the face corner angles associated with such an interior node must satisfy the following constraint (refer to Fig. 4.15):

$$\phi_{jk} \leq \pi \qquad j = 1, \ldots, N_{\mathcal{N}}^{I}, \quad k = 1, \ldots, n_j^{\mathcal{N}}. \tag{4.34}$$

If $\phi_{jk} > \pi$, the edge modules connected to such an interior node overlap in $\mathcal{S}_0$, which is not allowed. Figure 4.15 illustrates this requirement.

Assuming that the inequality constraints (4.11)–(4.13) allow for a valid solution for the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^{I}$, such a solution is *not unique* because in general $N_{\text{dof}} > 0$ (refer to (4.33)). One global measure to discriminate

among various design solutions and drive toward a possible unique solution is referred to as the *surface area efficiency* E and is defined as follows:

$$\mathsf{E} = \frac{\text{Area}(\mathcal{M})}{\text{Area}(\mathcal{S}_0)}. \tag{4.35}$$

The numerical procedure outlined in Sect. 4.2.5 for the determination of a sheet design does not account for any function to minimize or maximize during the design process (i.e., it only iteratively corrects an initial guess solution until the presented constraints are satisfied). However, other methods can be utilized to determine a sheet design that satisfies the presented constraints while optimizing a given function (e.g., maximize surface area efficiency E). Such extensions are application-dependent and are recommended for future studies (see Problem 4.5).

4.3 Examples of the Tuck-Folding Method

In this section, we provide representative examples of the tuck-folding method described in Sect. 4.2. This method has been implemented in MATLAB®. The faces of the goal meshes and the edge modules are visualized as filled three-dimensional polygons using the MATLAB command `fill3`. The complete set of MATLAB scripts used to generate the examples is included in the Supplemental Materials for this chapter and described in Appendix B.3. We use the MATLAB implementation of the kinematic model presented in Chap. 2 to simulate the folding motion of the designed origami sheets.

Figure 4.16 shows examples of saddle meshes for which the unfolding polyhedra method studied in Chap. 3 cannot provide valid (non-overlapping) sheet designs (see Fig. 3.21a, b). It is shown in Fig. 4.16 that the tuck-folding method presented in this chapter successfully provides valid sheet designs for the considered saddle meshes. Similarly, the goal mesh $\mathcal{M}$ of a triangulated hat for which the unfolding polyhedra method cannot determine a valid sheet design as shown in Fig. 3.21c is considered in Fig. 4.17. It is another example where the tuck-folding method can be used to find a valid sheet design that cannot be obtained via the unfolding polyhedra method.

The example presented in Fig. 4.18 considers a goal mesh $\mathcal{M}$ associated with a shape of constant positive Gaussian curvature [17, 24]. The reference configuration $\mathcal{S}_0$ of a sheet design determined using the tuck-folding method and the folding motion towards its corresponding goal configuration $\mathcal{S}_\star$ are shown in such a figure. A goal mesh associated with a shape of constant negative Gaussian curvature is considered in Fig. 4.19. It is interesting to contrast the similarities in $\mathcal{S}_0$ between Figs. 4.18 and 4.19 with the clear differences in the associated goal configurations $\mathcal{S}_\star$.

To illustrate the multiplicity of possible solutions discussed in Sect. 4.2.7, four determined sheet designs associated with a single goal mesh $\mathcal{M}$ are presented in

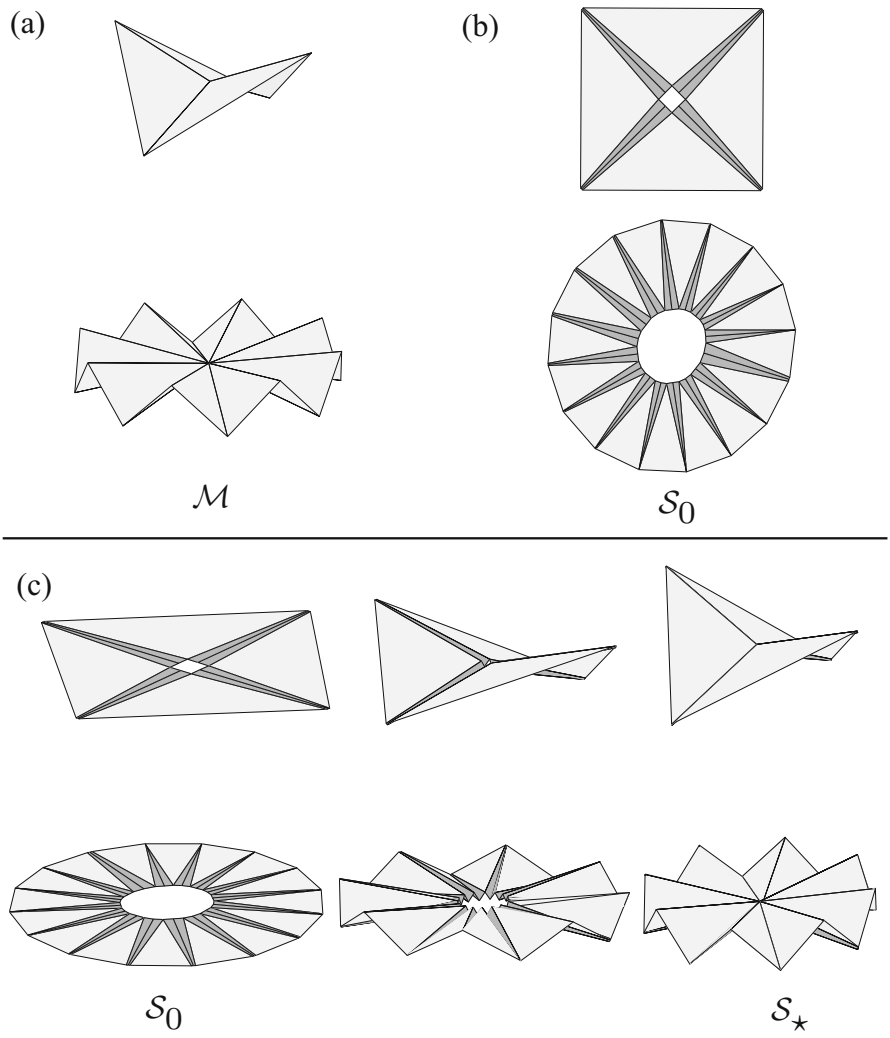


Fig. 4.16 (**a**) Goal meshes for saddle shapes that cannot be feasibly unfolded using the unfolding polyhedra method. (**b**) Valid origami sheet designs determined using the tuck-folding method (cf. Fig. 3.21a, b). (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

Fig. 4.20. These different sheet designs are obtained by considering different initial guesses for the sheet design variables in the iterative solution procedure described in Sect. 4.2.5. Values of the surface area efficiency E defined in (4.35) for the determined sheet designs are also provided.

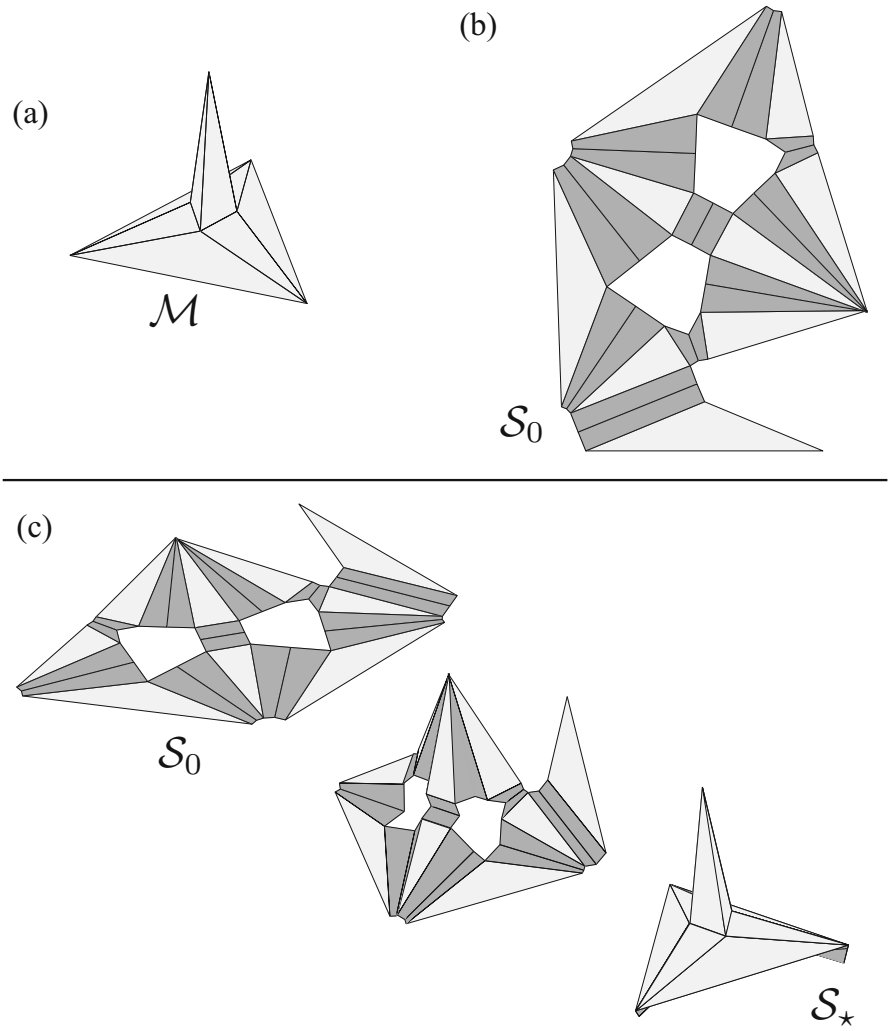


Fig. 4.17 (**a**) Goal mesh $\mathcal{M}$ for a triangulated hat open at the base that cannot be feasibly unfolded using the unfolding polyhedra method. (**b**) Valid origami sheet design determined using the tuck-folding method (cf. Fig. 3.21c). (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

Figure 4.21 shows sheet designs obtained through the tuck-folding method for two of the Kepler–Poinsot polyhedra [25, 26]: the great dodecahedron and the small stellated dodecahedron.

The examples in the previous figures consider goal meshes comprised of triangular faces only. However, the tuck-folding method is applicable to goal meshes

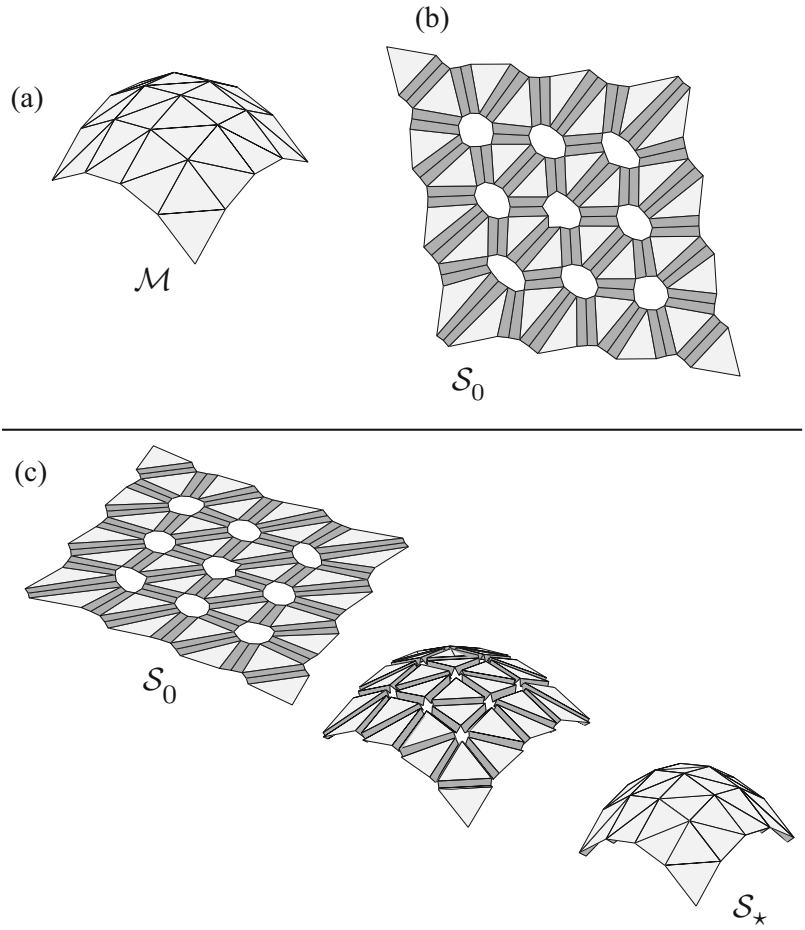


Fig. 4.18 (**a**) Goal mesh $\mathcal{M}$ associated with a shape of constant positive Gaussian curvature. (**b**) Determined origami sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

comprised of arbitrary convex polygonal faces. For example, Fig. 4.22 shows results associated with a goal mesh comprised of quadrilateral faces.

More complex examples are shown in Figs. 4.23, 4.24, 4.25, and 4.26 where goal meshes associated a parabolic disk shape, the exterior of a car, a dirigible balloon, and the Stanford bunny are considered, respectively. In all cases, the tuck-folding method successfully provides feasible origami sheet designs.

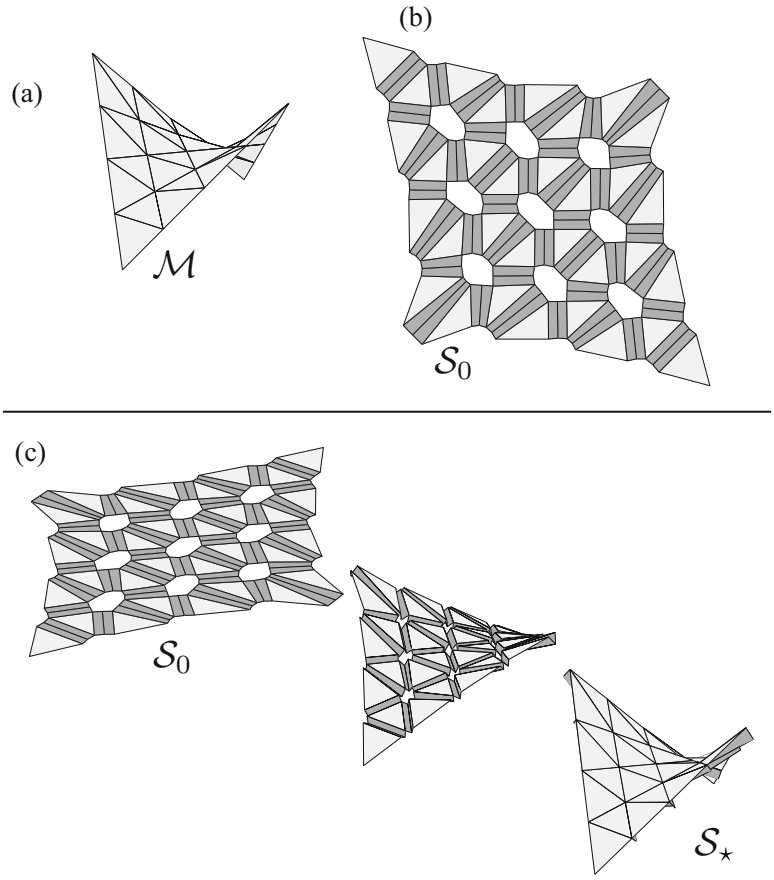


Fig. 4.19 (**a**) Goal mesh $\mathcal{M}$ associated with a shape of constant negative Gaussian curvature. (**b**) Determined origami sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

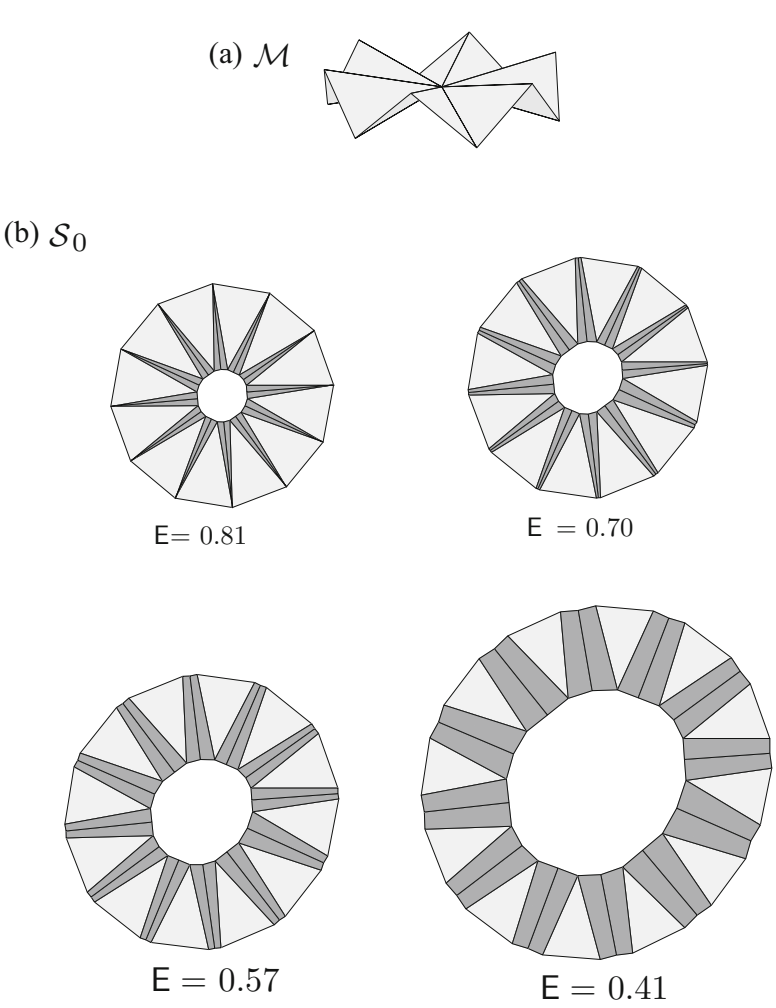


Fig. 4.20 (**a**) A goal mesh $\mathcal{M}$ and (**b**) reference configurations of four determined sheet designs with different values of surface area efficiency E

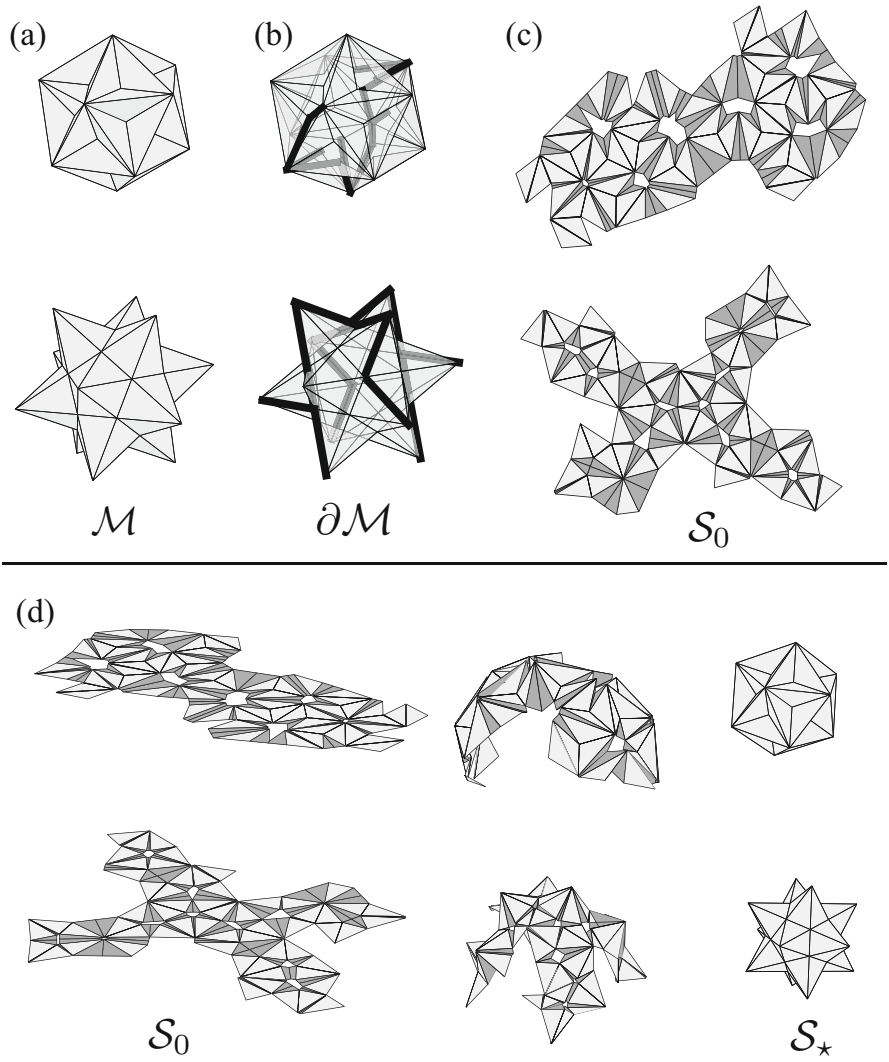


Fig. 4.21 (**a**) Goal meshes for the great dodecahedron (top) and the small stellated dodecahedron (bottom). (**b**) Boundaries of the goal meshes. (**c**) Associated origami sheet designs. (**d**) Continuous folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

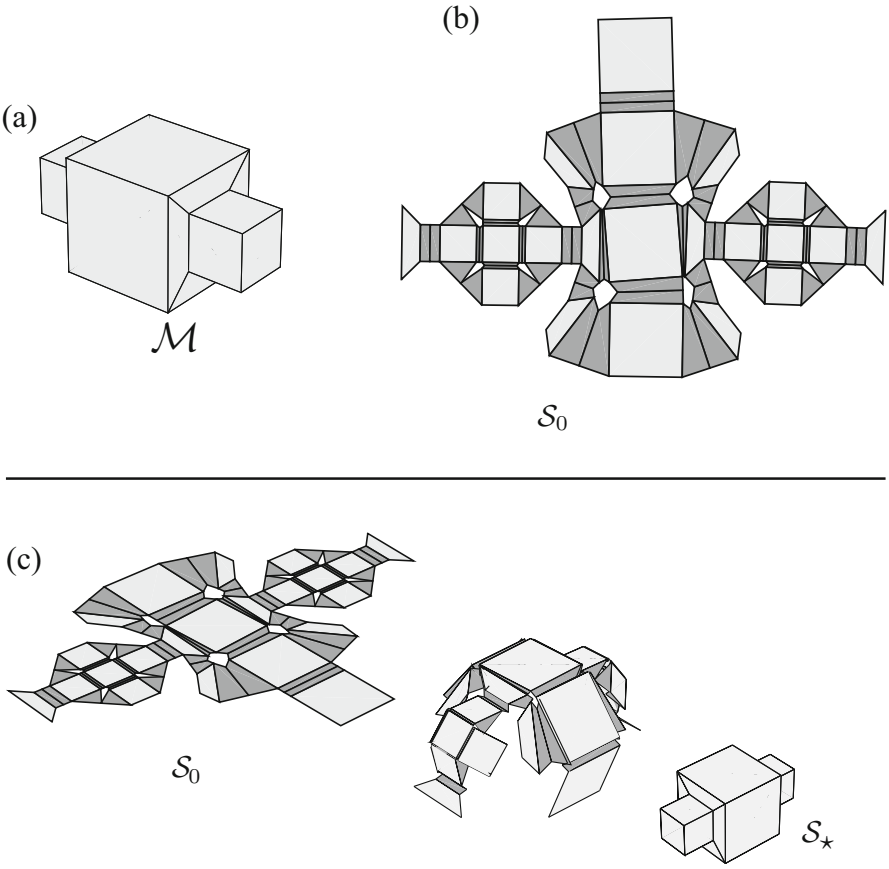


Fig. 4.22 (**a**) Goal mesh $\mathcal{M}$ comprised of quadrilateral faces. (**b**) Determined sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

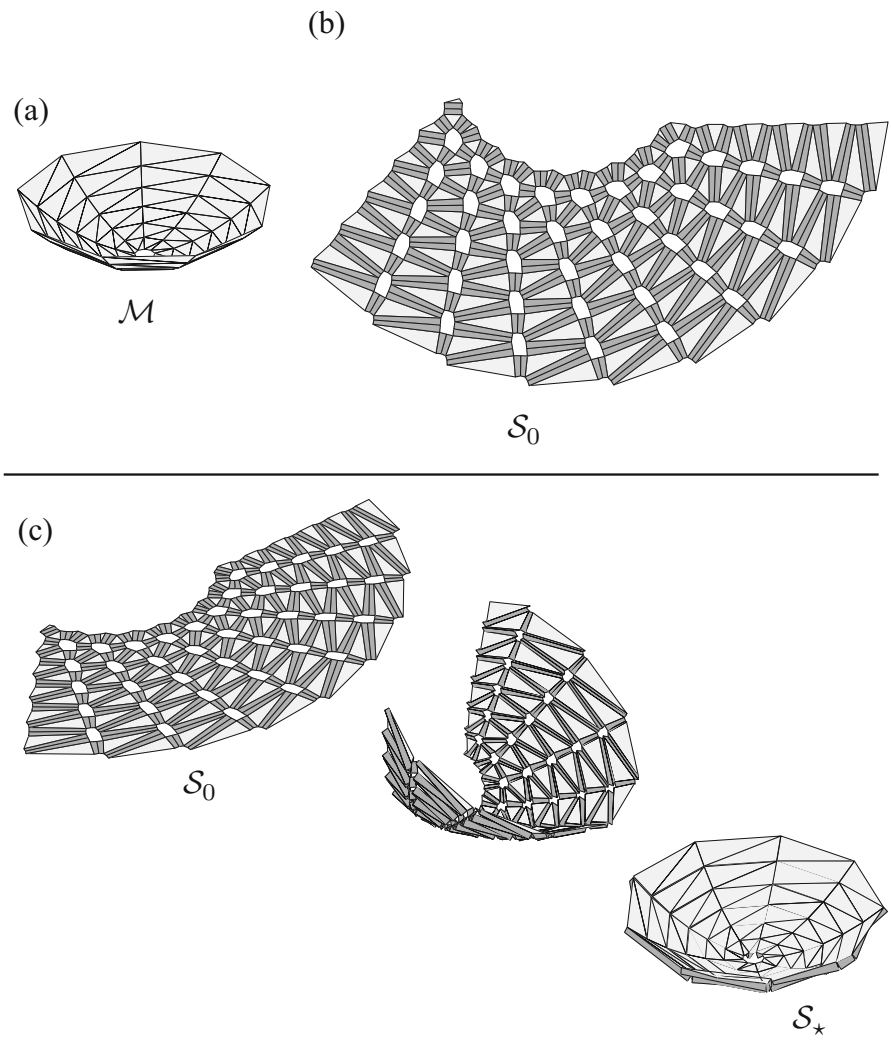


Fig. 4.23 (**a**) Goal mesh $\mathcal{M}$ for a parabolic shape. (**b**) Determined sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

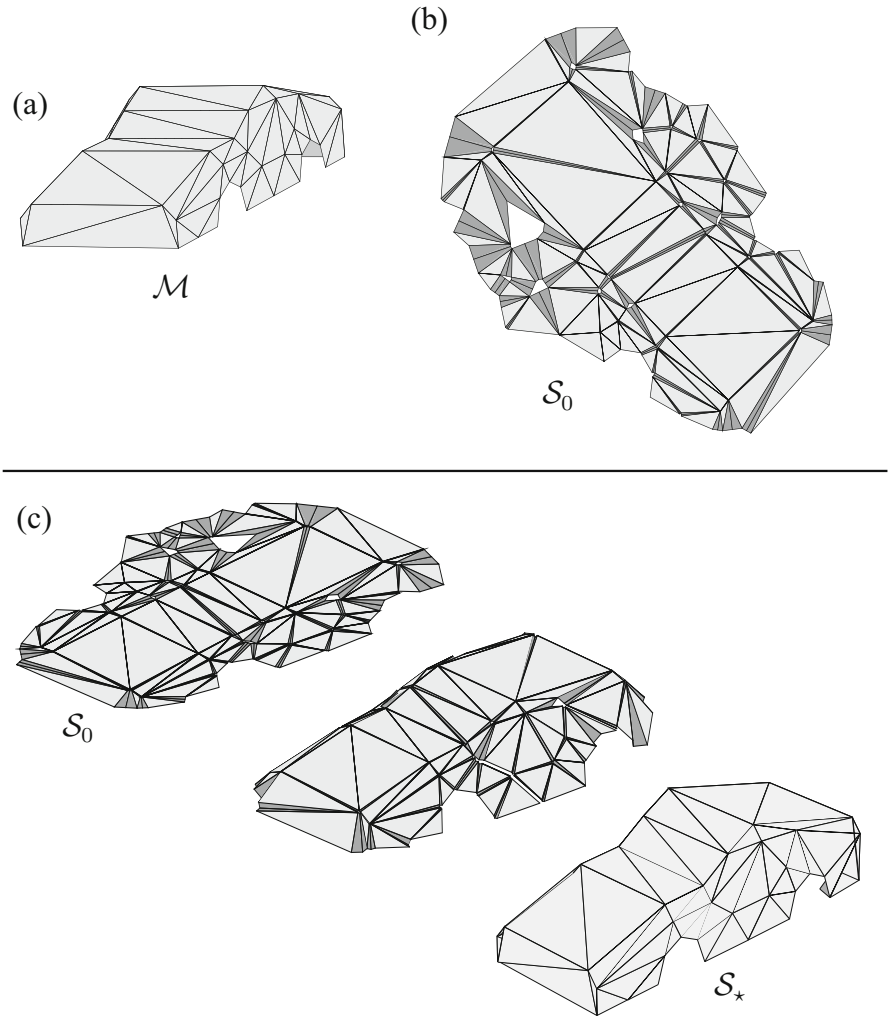


Fig. 4.24 (**a**) Goal mesh $\mathcal{M}$ for the shape of a car exterior. (**b**) Determined sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

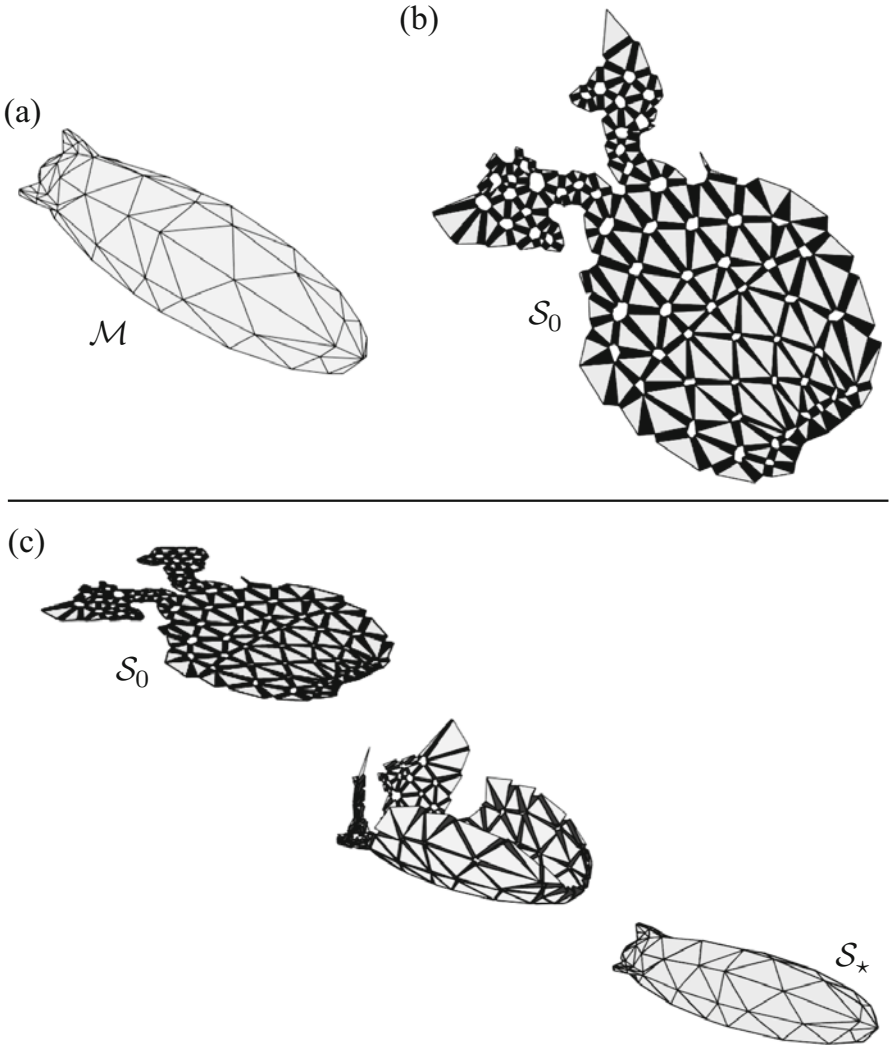


Fig. 4.25 (**a**) Goal mesh $\mathcal{M}$ of a dirigible balloon. (**b**) Determined sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

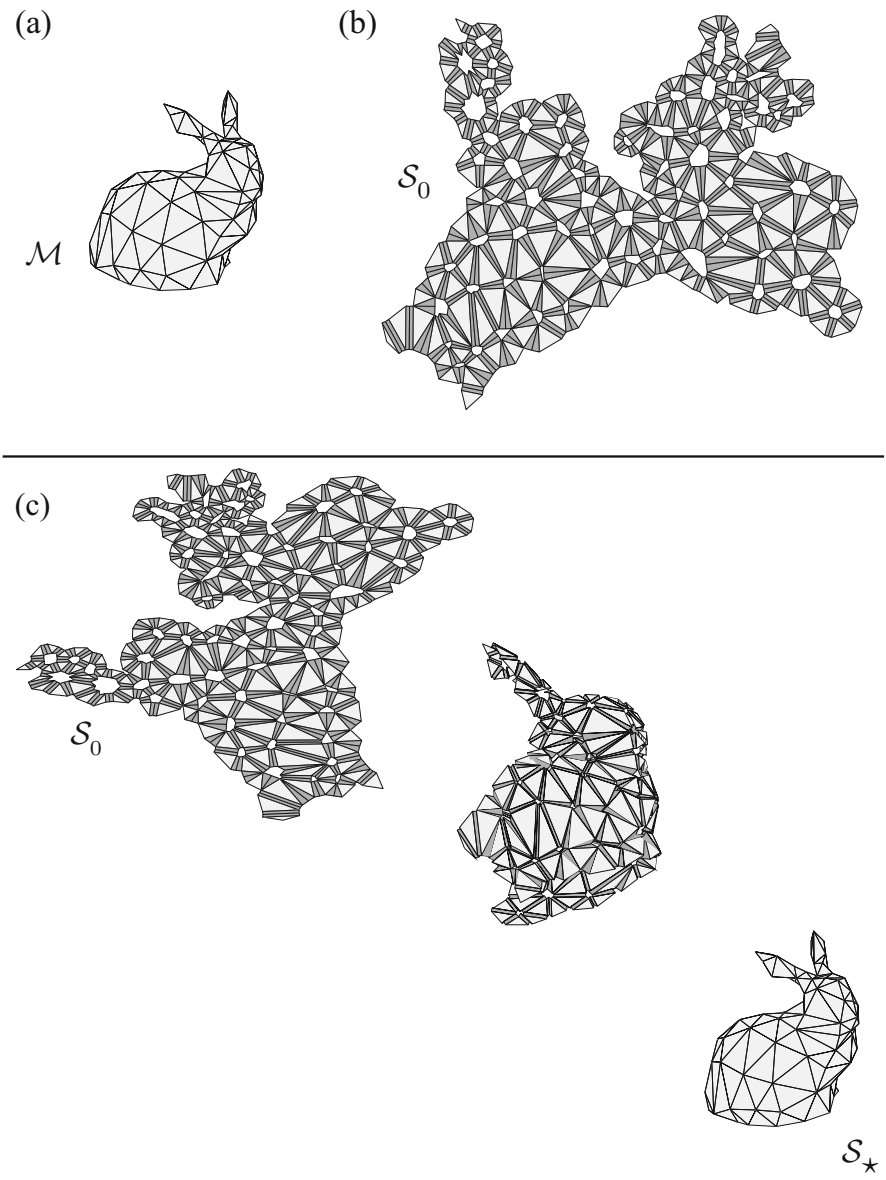


Fig. 4.26 (**a**) Goal mesh $\mathcal{M}$ of the Stanford Bunny [27]. (**b**) Determined sheet design. (**c**) Folding motion from the planar reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$

Chapter Summary

This chapter presented a method for the design of origami sheets that achieve targeted three-dimensional shapes termed here as the *tuck-folding method*. It is applicable to a much larger spectrum of three-dimensional goal shapes than the unfolding polyhedra method studied in Chap. 3 as illustrated in Figs. 4.16 and 4.17. In the tuck-folding method, the goal shape is represented as a three-dimensional polygonal mesh termed as the goal mesh. The method is based on the introduction of regions having two rigid faces and three creased folds placed between any two faces of the goal mesh originally connected by an interior edge. The creased folds are used to *tuck-fold* these introduced regions to form the shape of the goal mesh starting from a planar configuration.

Section 4.2.1 summarized the problem description and solution approach of the tuck-folding method. The data required to define the goal mesh and any required mesh modifications were described in Sect. 4.2.2. We then addressed the parameterization of the origami sheet design and the associated constraints on the design variables in Sect. 4.2.3. A numerical approach to determine the design variables subject to such constraints was presented in Sect. 4.2.5. The simulation of the folding motion between the planar reference configuration of the determined sheet designs and the goal configuration that matches the goal mesh was addressed in Sect. 4.2.6. An expanded discussion of the tuck-folding method was presented in Sect. 4.2.7 and representative examples of the implemented method considering various goal shapes were provided in Sect. 4.3.

Problems

4.1 Using the approach presented in Sect. 2.4.2, derive kinematic constraints for origami sheets designed using the method presented in this chapter for which the folds associated with an interior fold intersection are incident to a hole rather than to a single vertex. Hint: Formulate the transformation associated with folding each creased fold incident to the hole using the transformation matrix provided in (2.58).

4.2 Investigate and implement two algorithms for the determination of a boundary $\partial\mathcal{M}^C$ in an arbitrary 2-manifold surface mesh such that the mesh becomes topologically equivalent to a disk (refer to Sect. 4.2.2).

4.3 Derive the loop closure constraints provided in Sect. 4.2.3 considering vectors $\mathbf{b}^{j1}, \ldots, \mathbf{b}^{jn_j^{\mathcal{N}}}$ (refer to Fig. 4.7) defined in clockwise order instead of the counter-clockwise order assumed in Sect. 4.2.3. Are the resulting constraints equivalent to those provided in (4.6) and (4.7)?

4.4 An alternative formulation of the constraints on the design variables W_i, ψ_i, $i = 1, \ldots, N_{\mathcal{E}}^I$, can be obtained by replacing the inequality constraint (4.12) with the following equality constraint:

$$\hat{W}_i = \|\hat{\mathbf{z}}^i\| \sin\left(\left|\frac{\hat{\psi}_i}{2}\right|\right). \tag{4.36}$$

What would be the implications of using the more restrictive constraint (4.36) instead of (4.12)? How many equality and inequality constraints would this alternative formulation have?

Implement this alternative formulation in a computational environment and determine origami sheet designs for two examples from Sect. 4.3 and compare your results with those of Sect. 4.3.

4.5 Propose and implement a procedure to determine the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, by solving the design problem stated in (4.21) where the function $f(\mathbf{D})$ to be minimized is given as follows:

$$f(\mathbf{D}) = \mathsf{E}(\mathbf{D}), \tag{4.37}$$

and the surface area efficiency E is defined in (4.35).

4.6 Following the approach presented in Sect. 3.2.4, formulate the mapping of the mesh faces from their position in the goal mesh $\mathcal{M}$ to their corresponding position in the reference configuration $\mathcal{S}_0$ of an origami sheet designed using the tuck-folding method.

4.7 Implement the tuck-folding method for origami with creased folds in a computational environment. Then, determine a sheet design for the goal mesh with one interior node of positive Gaussian curvature shown in Fig. 4.27. Assume fold widths of 0.003 for all the smooth folds. The position vectors of the nodes are given as follows:

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\ \hat{\mathbf{y}}^7\right]^\top = \begin{bmatrix} 0.000 & 0.000 & 0.000 \\ 0.060 & 0.000 & -0.040 \\ 0.030 & 0.052 & -0.040 \\ -0.030 & 0.052 & -0.040 \\ -0.060 & 0.000 & -0.040 \\ -0.030 & -0.052 & -0.040 \\ 0.030 & -0.052 & -0.040 \end{bmatrix}, \tag{4.38}$$

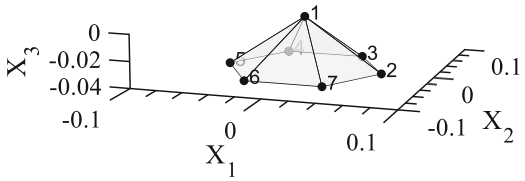


Fig. 4.27 Goal mesh $\mathcal{M}$ for Problem 4.7

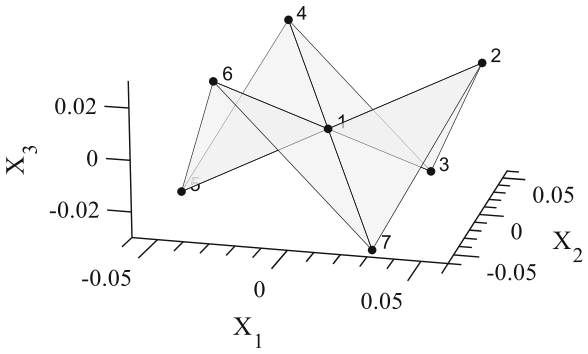


Fig. 4.28 Goal mesh $\mathcal{M}$ for Problem 4.8

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 3 & 4 \\ 1 & 4 & 5 \\ 1 & 5 & 6 \\ 1 & 6 & 7 \\ 1 & 7 & 2 \end{bmatrix}. \tag{4.39}$$

4.8 Implement the tuck-folding method for origami with creased folds in a computational environment. Then, determine a sheet design for the goal mesh with one interior node of negative Gaussian curvature shown in Fig. 4.28. Assume fold widths of 0.003 for all the smooth folds. The position vectors of the nodes are given as follows:

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\ \hat{\mathbf{y}}^7\right]^{\top} = \begin{bmatrix} 0.000 & 0.000 & 0.000 \\ 0.060 & 0.000 & 0.030 \\ 0.030 & 0.052 & -0.030 \\ -0.030 & 0.052 & 0.030 \\ -0.060 & 0.000 & -0.030 \\ -0.030 & -0.052 & 0.030 \\ 0.030 & -0.052 & -0.030 \end{bmatrix}, \tag{4.40}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given in (4.39).

4.9 Implement the tuck-folding method in a computational environment and determine sheet designs for the 13 Archimedean polyhedra. Compare your results with those of Problem 3.7.

4.10 Implement the tuck-folding method in a computational environment and determine sheet designs for two different non-convex goal meshes (your choice).

4.11 Determine the trim angles τ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^{I}$, $k = 1, \ldots, n_j^{\mathcal{N}}$, for the edge modules associated with the goal mesh $\mathcal{M}$ illustrated in Fig. 3.9.

4.12 Repeat Problem 4.11 considering the goal mesh $\mathcal{M}$ illustrated in Fig. 3.31.

References

1. R.J. Lang, A computational algorithm for origami design, in *Proceedings of the Twelfth Annual Symposium on Computational Geometry* (ACM, New York, 1996), pp. 98–105
2. T. Tachi, Origamizing polyhedral surfaces. IEEE Trans. Vis. Comput. Graph. **16**(2), 298–311 (2010)
3. E.D. Demaine, J. O'Rourke, *Geometric Folding Algorithms* (Cambridge University Press, Cambridge, 2007)
4. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr, D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
5. C.C. Chu, C.K. Keong, The review on tessellation origami inspired folded structure, in *AIP Conference Proceedings*, vol. 1892 (AIP Publishing, 2017), p. 020025. https://doi.org/10.1063/1.5005665
6. W. Schlickenrieder, Nets of polyhedra. Master's thesis, Technische Universität Berlin (1997)
7. M. Bern, E.D. Demaine, D. Eppstein, E. Kuo, A. Mantler, J. Snoeyink, Ununfoldable polyhedra with convex faces. Comput. Geom. **24**(2), 51–62 (2003)
8. T. Tachi, 3D origami design based on tucking molecule, in *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education* (2009), pp. 259–272
9. E.D. Demaine, T. Tachi, Origamizer: a practical algorithm for folding any polyhedron, in *33rd International Symposium on Computational Geometry (SoCG 2017)*. Leibniz International Proceedings in Informatics (LIPIcs), vol. 77 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, 2017), pp. 34:1–34:16
10. T. Tachi, Designing freeform origami tessellations by generalizing Resch's patterns. J. Mech. Des. **135**(11), 111006 (2013)
11. R. Engelking, *General Topology* (Heldermann Verlag, Berlin, 1989)
12. J. Erickson, S. Har-Peled. Optimally cutting a surface into a disk. Discret. Comput. Geom. **31**(1), 37–59 (2004)
13. T.K. Dey, H. Schipper, A new technique to compute polygonal schema for 2-manifolds with application to null-homotopy detection. Discret. Comput. Geom. **14**(1), 93–110 (1995)
14. F. Lazarus, M. Pocchiola, G. Vegter, A. Verroust, Computing a canonical polygonal schema of an orientable triangulated surface, in *Proceedings of the Seventeenth Annual Symposium on Computational Geometry* (ACM, New York, 2001), pp. 80–89
15. E. C. de Verdiere, F. Lazarus, Optimal polygonal schema on an orientable surface, in *18th European Workshop on Computational Geometry* (2002), pp. 627–636
16. J.M. Sullivan, Curvatures of smooth and discrete surfaces, in *Discrete Differential Geometry* (Springer, Berlin, 2008), pp. 175–188
17. E. Akleman, J. Chen, Insight for practical subdivision modeling with discrete gauss-bonnet theorem, in *International Conference on Geometric Modeling and Processing* (Springer, Berlin, 2006), pp. 287–298
18. G. Song, N.M. Amato, A motion-planning approach to folding: from paper craft to protein folding. IEEE Trans. Rob. Autom. **20**(1), 60–71 (2004)
19. Z. Xi, J.-M. Lien, Continuous unfolding of polyhedra – a motion planning approach, in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, New York, 2015), pp. 3249–3254

20. N.M. Amato, K.A. Dill, G. Song, Using motion planning to map protein folding landscapes and analyze folding kinetics of known native structures. J. Comput. Biol. **10**(3–4), 239–255 (2003)
21. Z. Xi, J.-M. Lien, Plan folding motion for rigid self-folding machine via discrete domain sampling, in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, New York, 2015), pp. 2938–2943
22. Z. Xi, J.-M. Lien, Folding and unfolding origami tessellation by reusing folding path, in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, New York, 2015), pp. 4155–4160
23. Z. Xi, J.-M. Lien, Folding rigid origami with closure constraints, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE* (American Society of Mechanical Engineers, New York, 2014), pages No. DETC2014–35556, pp. V05BT08A052
24. C.R. Calladine, *Theory of Shell Structures* (Cambridge University Press, Cambridge, 1989)
25. B. Grünbaum, Regular polyhedra–old and new. Aequationes Math. **16**(1), 1–20 (1977)
26. D. Sutton, *Platonic & Archimedean Solids* (Walker & Company, New York, 2002)
27. G. Turk, M. Levoy, The Stanford Bunny. The Stanford 3D Scanning Repository (1994). http://graphics.stanford.edu/data/3Dscanrep/

Chapter 5
Kinematics of Origami Structures with Smooth Folds

Abstract A kinematic model for origami with creased folds was presented in Chap. 2 and most existing models for origami also assume that folds are straight creases. However, such previous models are not intended for origami structures having non-negligible fold thickness or maximum fold curvature constraints based on material or structural limitations. In this chapter, we develop a model that captures the kinematic response of sheets having realistic folds of non-zero surface area and exhibiting higher-order geometric continuity, referred to as *smooth folds*. The geometry of smooth folds and the constraints on their associated kinematic variables are presented. We also address the implementation of the model in a computational environment and provide various representative examples.

5.1 Introduction

As stated in Chap. 2, the following assumptions underlie most origami models proposed to date [1, 2]: (1) folds are straight creases (Fig. 5.1a), and (2) planar faces bounded by the folds and the sheet boundary are rigid (i.e., these faces neither stretch nor bend). Origami modeling under these assumptions has been studied in the past and remains an active research subject [2–6] (see Sect. 1.4).

The idealization of origami structures as sheets having creased folds has been useful in the modeling and design of many deployable structures [2, 4–11] and mechanisms to date [12–25]. However, this simplification may not be appropriate for structures having non-negligible fold thickness or that are comprised of materials incapable of the strain magnitudes needed to generate the high curvatures compatible with a creased idealization. For these structures, the folded regions are not accurately represented as creases but rather as *bent* sheet regions having higher-order geometric continuity. These folded regions are referred to as *smooth folds* (Fig. 5.1b).

Electronic Supplementary Material The online version of this article (https://doi.org/10.1007/978-3-319-91866-2_5) contains supplementary material, which is available to authorized users.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_5

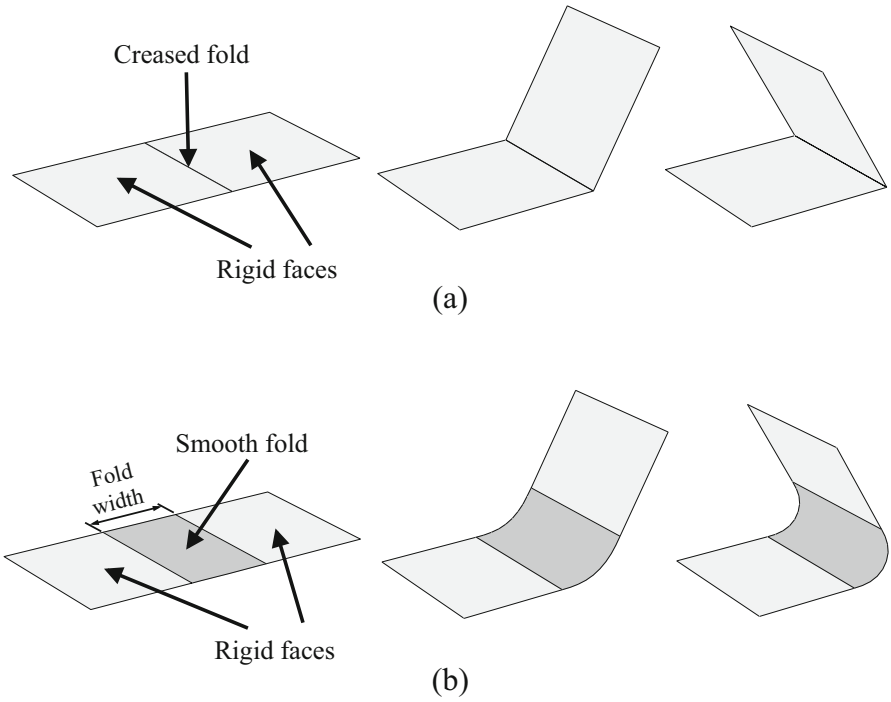


Fig. 5.1 (**a**) A sheet having a creased fold. (**b**) A sheet having a *smooth fold* of non-zero surface area

There have been past efforts to model and render surfaces that exhibit smooth folding. Bent and creased surfaces have been simulated using collections of developable surfaces [26–30] to allow for the realistic animation and rendering of curved folds and combinations of creases and bent regions. Hwang and Yoon modeled origami-like developable surfaces via bending operations analogous to wrapping regions of an initially planar surface onto cylindrical and conical sections [28]. Developable surfaces that represent bent and creased sheets have also been simulated using collections of planar, cylindrical, conical, and other ruled surface subdomains [26, 27]. However, none of the aforementioned works [26–30] has considered constraints on the geometry and deformation of the bent folded regions that are required to preserve rigid faces as in analogous models that assume rigid faces and creased folds. In view of this, we provide a fully general model for the kinematic response of origami structures having smooth folds and rigid faces in this chapter. Such a model then includes as a special case the deformation of structures with creased (i.e., zero width) folds.

The model presented in this chapter is particularly useful for the kinematic analysis of sheets folded via active material actuation, where the achievable curvature at the folds is limited by the maximum strain magnitude provided by such materials [31]. Examples of these active origami structures are reviewed in Chap. 1

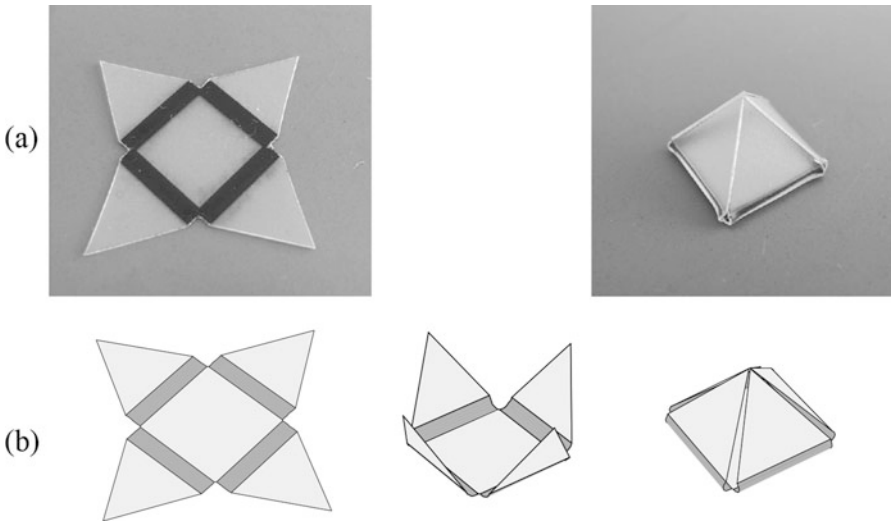


Fig. 5.2 Motivation for the current model based on active materials: (**a**) Photographs of a pyramid created via thermally activated folding of an SMP sheet. *Adapted from [34] with permission of The Royal Society of Chemistry. Credit: Ying Liu and Michael Dickey*; (**b**) Schematics of the same pyramid simulated using the kinematic model presented in this chapter

and they include folds driven by liquid-crystal elastomers [32, 33], shape memory polymers (SMPs) [34] (see Fig. 5.2), shape memory alloys (SMAs) [35–38], dielectric elastomers [39, 40], and others [31, 41–43]. The different assumptions regarding strain distributions at the fold regions associated with the chosen active materials require an arbitrary order of geometric continuity at the fold/face junctions. This aspect is considered in the present chapter.

In Sect. 5.2, we formalize the concepts and assumptions made in the new kinematic model, especially as related to the model for origami with creased folds presented in Chap. 2. The chapter continues with the mathematical description of smooth folds in Sect. 5.3. In Sect. 5.4, we provide the mathematical description of the fold pattern in origami sheets with smooth folds. Subsequently, we address the kinematic constraints that restrict the possible fold patterns and folding motions for origami structures with smooth folds in Sect. 5.5. The formulation that maps the reference configuration of a sheet to all other possible configurations is examined in Sect. 5.6. The implementation of the model in a computational framework is addressed in Sect. 5.7, while Sect. 5.8 presents simulation results of sheets having different fold patterns that demonstrate the kinematic responses captured by the model. Upon the study of this chapter, the reader will be able to derive the kinematic model for origami with smooth folds and to implement it in a computational framework. The Supplemental Material associated with this chapter provides the reader with a number of MATLAB® scripts that represent such an implementation.

5.2 Fundamental Concepts

In this section, we present the fundamental concepts used in the kinematic modeling of origami structures with smooth folds. The modeling approach for origami adopted here is primarily based on that presented in Chap. 2. First, the continuum surface termed as the *sheet* and the kinematic variables associated with the smooth folds in the sheet (e.g., the *fold angles*) are defined. This continuum surface studied throughout the chapter represents the *mid-surface* of an origami structure of finite thickness.

The layout of the folds in the sheet (i.e., the *fold pattern*) is then determined in terms of *vertices* (start points and end points of the line segments coincident with the centerlines of the smooth folds in a planar reference configuration) and their connectivity. Here the required fold pattern data (e.g., vertex coordinates and fold connectivity) are assumed given. We address two methods for designing these fold patterns to achieve goal shapes in Chaps. 6 and 7. After the geometric parameters of the fold pattern are defined, mathematical *constraints* on the fold kinematic variables required for valid configurations are derived. The *continuous motion* of the sheet is represented as a continuous evolution of the fold kinematic variables such that any attained configuration satisfies the kinematic constraints.

Let us consider origami sheets such as those illustrated in Fig. 5.3, which are surfaces in three-dimensional space. Further, the sheets are *connected* surfaces, which means that any two points in the sheets can be joined by a path fully contained within them. Each sheet has an outer closed boundary and thus they are also *bounded* surfaces. As also shown in Fig. 5.3, the sheets are divided into faces and smooth folds. The sheet has $N_{\mathcal{P}}$ faces and $N_{\mathcal{F}}$ smooth folds ($N_{\mathcal{P}} = 4$ and $N_{\mathcal{F}} = 3$ for the strip in Fig. 5.3; $N_{\mathcal{P}} = 16$ and $N_{\mathcal{F}} = 19$ for the crane in Fig. 5.3).

Figure 5.3 shows different configurations of the two example origami sheets during their folding motion. We consider the folding motion of such origami sheets from $t = 0$ to $t = t_f$ where $t \in [0, t_f]$ is a *time parameter* that tracks the history of folding motion. In a given configuration, each point in the sheet is located at some definite position in three-dimensional space $\mathbb{R}^3$. The *current configuration* of a sheet at any t is denoted $\mathcal{S}_t$. The geometric configurations of the $N_{\mathcal{P}}$ faces and the $N_{\mathcal{F}}$ smooth folds in $\mathcal{S}_t$ are denoted $\mathcal{P}_t^1, \ldots, \mathcal{P}_t^{N_{\mathcal{P}}}$ and $\mathcal{F}_t^1, \ldots, \mathcal{F}_t^{N_{\mathcal{F}}}$, respectively. Therefore, $\mathcal{S}_t = \left(\bigcup_{i=1}^{N_{\mathcal{P}}} \mathcal{P}_t^i\right) \cup \left(\bigcup_{i=1}^{N_{\mathcal{F}}} \mathcal{F}_t^i\right)$.

Following the notation of Chap. 2, the orthonormal vectors $\mathbf{e}_i \in \mathbb{R}^3,\ i = 1, 2, 3$ form the basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ that defines the fixed global coordinate system. The *reference configuration* of the sheet is denoted $\mathcal{S}_0$ and is defined such that it is contained in the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$ with its surface subdomains (the faces and the smooth folds) not overlapping as illustrated in Fig. 5.3. The configuration of the $N_{\mathcal{P}}$ faces and the $N_{\mathcal{F}}$ smooth folds in $\mathcal{S}_0$ are denoted $\mathcal{P}_0^1, \ldots, \mathcal{P}_0^{N_{\mathcal{P}}}$ and $\mathcal{F}_0^1, \ldots, \mathcal{F}_0^{N_{\mathcal{F}}}$, respectively. The side of $\mathcal{S}_0$ with normal vector $\mathbf{e}_3$ is selected as the *positive side* of the sheet. The configuration of the sheet at $t = t_f$ is termed as the *final configuration* $\mathcal{S}_{t_f}$.

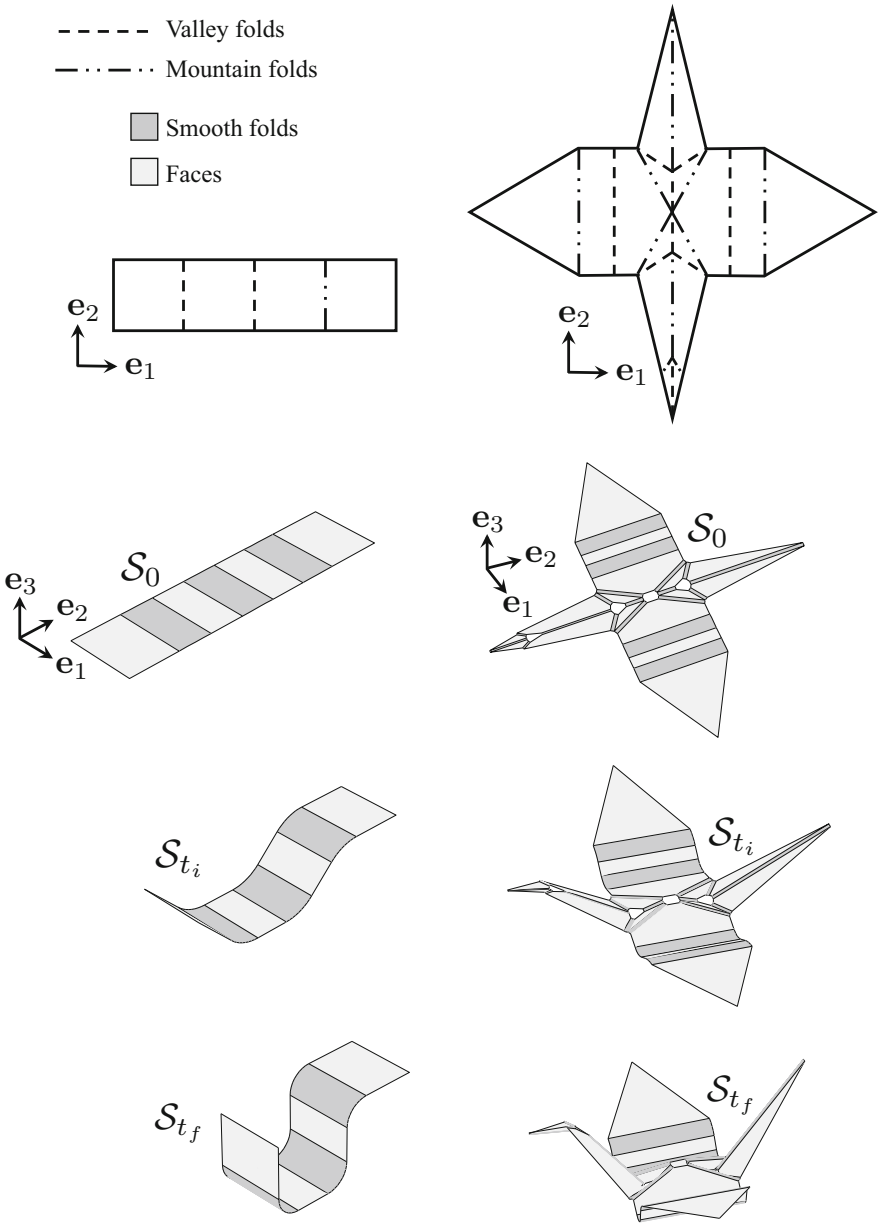


Fig. 5.3 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ for two sheets with smooth folds

A *valid* configuration $\mathcal{S}_t$ has the following characteristics:

1. The faces undergo only rigid deformations (i.e., they do not stretch or bend),
2. The sheet is not torn (initially joined surface subdomains of the sheet remain joined), and
3. The sheet does not self-intersect.

These characteristics of a valid configuration are the same as those of a valid configuration for a sheet with creased folds presented in Sect. 2.2. However, it is noted that unlike a sheet with creased folds, the sheets considered here are also comprised of other surface subdomains besides the faces (the smooth folds), for which bending and stretching are permitted.

As illustrated in Figs. 5.1, 5.2 and 5.3, the smooth folds are surfaces that locally bend along a single direction. To capture such a response, we parameterize the *smooth folds* $\mathcal{F}_t^1, \ldots, \mathcal{F}_t^{N_\mathcal{F}}$ as *ruled surfaces*[1] of the following form:

$$\boldsymbol{\mathcal{F}}_t^i(\zeta_1, \zeta_2) = \mathbf{c}_t^i(\zeta_1) + \zeta_2 \mathbf{h}_t^i, \qquad \frac{\mathrm{d}\mathbf{c}_t^i(\zeta_1)}{\mathrm{d}\zeta_1} \cdot \mathbf{h}_t^i = 0, \tag{5.1}$$

where $\boldsymbol{\mathcal{F}}_t^i(\zeta_1, \zeta_2) \in \mathbb{R}^3$ is a parameterization of $\mathcal{F}_t^i$. Without loss of generality, the range for the parameters ζ_1 and ζ_2 are taken as the intervals $[-1, 1]$ and $[0, 1]$, respectively. To simplify the notation, the dependence of $\boldsymbol{\mathcal{F}}_t^i(\zeta_1, \zeta_2)$, $\mathbf{c}_t^i(\zeta_1)$, and $\mathbf{h}_t^i$ on t is taken as implicit for the remainder of the chapter and they are denoted $\boldsymbol{\mathcal{F}}^i(\zeta_1, \zeta_2)$, $\mathbf{c}^i(\zeta_1)$, and $\mathbf{h}^i$, respectively. We illustrate a parametric surface $\boldsymbol{\mathcal{F}}^i(\zeta_1, \zeta_2)$ in Fig. 5.4.

In (5.1), $\mathbf{h}^i \in \mathbb{R}^3$ provides the direction of the rulings comprising $\mathcal{F}_t^i$ while $\mathbf{c}^i(\zeta_1) \in \mathbb{R}^3$ is the parametric curve that defines the *cross-section* of $\mathcal{F}_t^i$ and is contained in a plane orthogonal to $\mathbf{h}^i$.[2] The dimension of the folds along their length direction is given as $\|\mathbf{h}^i\|$. It is assumed that $\|\mathbf{h}^i\|$ is constant for all configurations and therefore *the smooth folds do not exhibit stretch along their length direction* (i.e., the direction along $\mathbf{h}^i$). As a consequence, the only non-rigid deformations allowed for the smooth folds are continuous bending or stretching of their cross-sections defined by $\mathbf{c}^1(\zeta_1), \ldots, \mathbf{c}^{N_\mathcal{F}}(\zeta_1)$.

In this work, the deformation of the smooth folds is simplified to *cylindrical bending* (i.e., the deformation of such surfaces is limited to bending and stretch along only one direction). Cylindrical bending is an assumption widely employed in the simplification of the analysis of plate structures that have a high length-to-width ratio and reduces the two-dimensional plate models to one-dimensional ones [45–47]. The assumption of cylindrical bending is made here because the

[1] A *ruled surface* is formed by the union of straight lines, called the *rulings* or generators of the surface [44].

[2] The *tangent vector* of the parametric curve $\mathbf{c}^i(\zeta_1)$ is given as $\frac{\mathrm{d}\mathbf{c}^i(\zeta_1)}{\mathrm{d}\zeta_1}$. Therefore, the expression $\frac{\mathrm{d}\mathbf{c}^i(\zeta_1)}{\mathrm{d}\zeta_1} \cdot \mathbf{h}^i = 0$ in (5.1) indicates that $\mathbf{c}^i(\zeta_1)$ is contained in a plane orthogonal to $\mathbf{h}^i$.

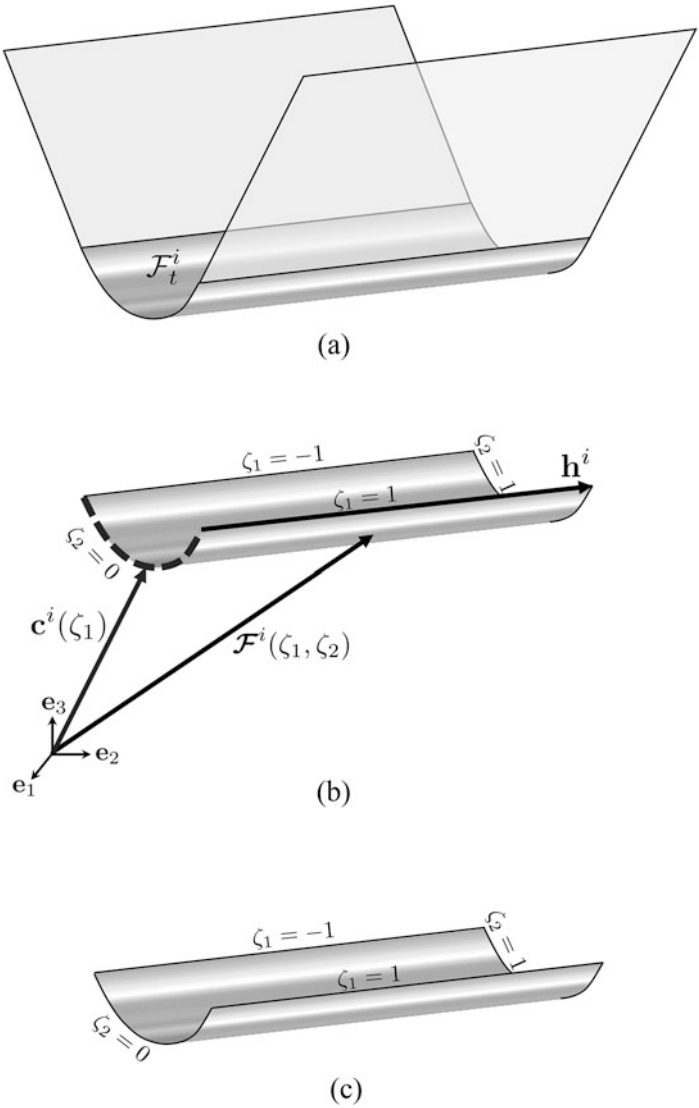


Fig. 5.4 (**a**) Smooth fold surface $\mathcal{F}_t^i$ and its adjacent faces. (**b**) Parameterization $\boldsymbol{\mathcal{F}}^i(\zeta_1, \zeta_2)$ describing the shape of $\mathcal{F}_t^i$. (**c**) Boundaries of $\mathcal{F}_t^i$

length of the smooth folds along the direction parallel to $\mathbf{h}^i$ (refer to Fig. 5.4) is often much larger than their fold width. This allows us to neglect stretch and bending along the direction parallel to $\mathbf{h}^i$. However, there are cases where cylindrical bending is not an accurate assumption. For example, origami structures having folds comprised of highly anisotropic materials may also not exhibit cylindrical bending.

Also, there are material systems used in origami structures that exhibit significant *anticlastic* bending (i.e., when they are folded along one direction, a bending deformation of opposite curvature is exhibited along the perpendicular direction). Liquid crystal elastomers [32] are examples of such material systems. The present model can be extended to account for general deformation of the smooth fold surfaces by generalizing (5.1) and allowing bending and stretch along the direction perpendicular to the fold width. Nevertheless, the resulting model would require a higher number of kinematic variables and more complex kinematic relations. Such a model is beyond the scope of this chapter. Finite element approaches using plate, shell, or three-dimensional continuum elements can also be employed to model the deformation of more general cases of folds that do not exhibit cylindrical bending.

A non-rigid deformation of the sheet is achieved by rotating pairs of faces that are joined to shared smooth folds relative to one another in such a manner that the sheet only attains valid configurations. The primary kinematic variable associated with a smooth fold describes the relative rotation between the two faces joined by the fold and is denoted as the *fold angle*. Based on the kinematics of creased folds, (Chap. 2), this *fold angle* $\hat{\theta}_i(t)$ is defined as π radians less the dihedral angle between the positive sides of the two faces joined to the ith smooth fold (see Fig. 5.5). The dependence of the fold angles on t is left implicit for the remainder of the chapter.

Comparing the crane with smooth folds in Fig. 5.3 against its counterpart with creased folds in Fig. 2.2, it is noted that the interior fold intersections for origami with smooth folds are taken to be holes in the sheet rather than single points.[3] The number of interior fold intersections in the sheet is denoted $N_{\mathcal{I}}$ ($N_{\mathcal{I}} = 4$ for the crane in Fig. 5.3; $N_{\mathcal{I}} = 0$ for the strip in Fig. 5.3).

5.3 Shape Formulation of Smooth Folds

This section presents the detailed geometrical description of smooth folds. The conditions required for various orders of geometric continuity and associated formulations for the parametric curves $\mathbf{c}^1(\zeta_1), \ldots, \mathbf{c}^{N_{\mathcal{F}}}(\zeta_1)$ that describe the cross-sectional shape of the smooth folds (see Fig. 5.5) are also provided.

The distance between the two end points of $\mathbf{c}^i(\zeta_1)$ with position vectors $\mathbf{c}^i(-1), \mathbf{c}^i(1) \in \mathbb{R}^3$ in any configuration is denoted by $\hat{w}_i$:

$$\hat{w}_i = \|\mathbf{c}^i(1) - \mathbf{c}^i(-1)\|. \tag{5.2}$$

[3]This is primarily to simplify the mathematical formulation, as the compound curvature of fold intersections will necessarily be much harder to model than that of the smooth folds $\boldsymbol{\mathcal{F}}^i(\zeta_1, \zeta_2)$, which have zero Gaussian curvature. Rigorous modeling of these interior fold intersections is left for future studies.

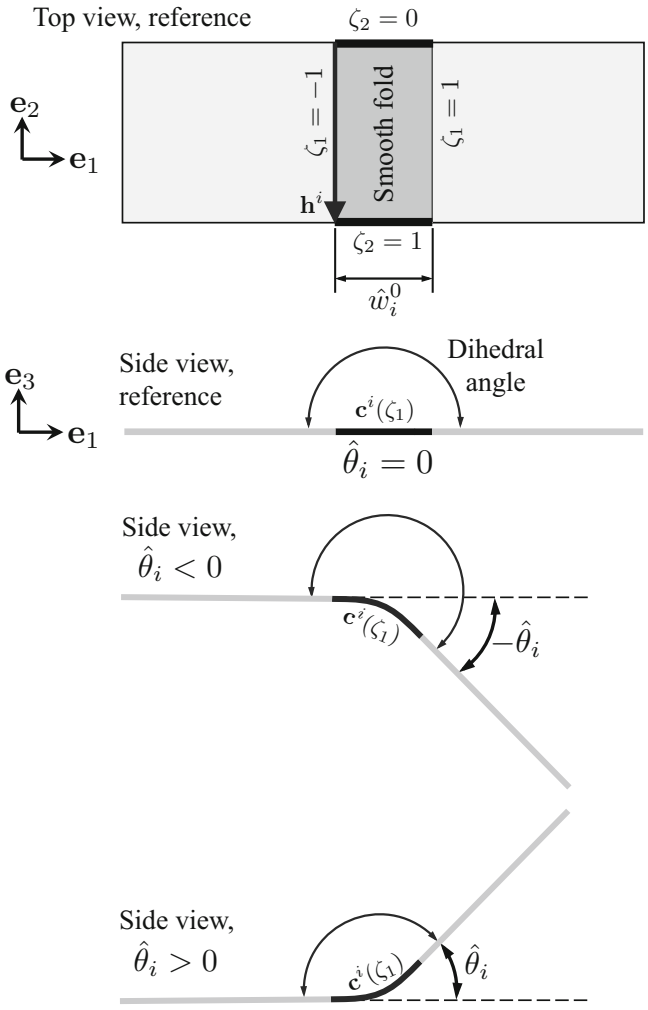


Fig. 5.5 Schematics showing the reference and folded configurations of a smooth fold and the concept of fold angle $\hat{\theta}_i$

A schematic of the cross-section of a smooth fold showing $\hat{w}_i$ is shown in Fig. 5.6. The *fold width* $\hat{w}_i^0$ is the value of $\hat{w}_i$ at $t = 0$ (i.e., at the reference configuration).

The fold-attached orthonormal vectors $\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i \in \mathbb{R}^3, i = 1, \ldots, N_{\mathcal{F}}$, form the bases $\{\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i\}$ that define the local *fold coordinate system* of each smooth fold. The origin of this coordinate system is located at the middle distance between the end points of $\mathbf{c}^i(\zeta_1)$ (i.e., at $\frac{1}{2}\left(\mathbf{c}^i(-1) + \mathbf{c}^i(1)\right)$). The vector $\hat{\mathbf{e}}_1^i$ is aligned to $\mathbf{h}^i$ while $\hat{\mathbf{e}}_2^i$ is aligned with $\mathbf{c}^i(1) - \mathbf{c}^i(-1)$.

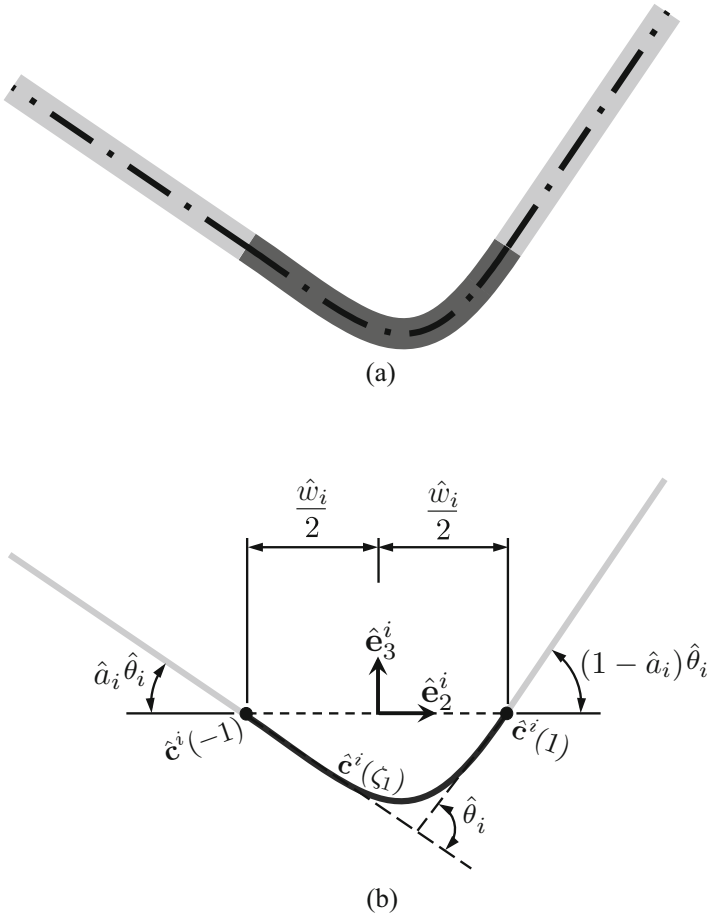


Fig. 5.6 Cross-section of a general smooth fold: (**a**) Origami sheet of non-zero thickness; (**b**) Mid-surface of the origami sheet modeled in this chapter. The fold kinematic variables $\hat{\theta}_i$, $\hat{w}_i$, and $\hat{a}_i$ and the fold-attached coordinate system are shown

In general, the face adjacent to a smooth fold at the boundary $\mathcal{F}^i(-1, \zeta_2)$ makes an angle of $\hat{a}_i\hat{\theta}_i$ with $-\hat{\mathbf{e}}_2^i$ in the plane spanned by $\hat{\mathbf{e}}_2^i$ and $\hat{\mathbf{e}}_3^i$. The face adjacent to a smooth fold at the boundary $\mathcal{F}^i(1, \zeta_2)$ makes an angle of $(1-\hat{a}_i)\hat{\theta}_i$ with $\hat{\mathbf{e}}_2^i$ in the same plane. This is shown in Fig. 5.6.

For the remainder of the book, we focus only on folds having *symmetric* cross-sections (i.e., those for which $\hat{a}_i = \frac{1}{2}$). The reader is referred to other sources for the treatment of more general smooth folds [48]. The cross-section of a symmetric smooth fold and its associated parameters are illustrated in Fig. 5.7.

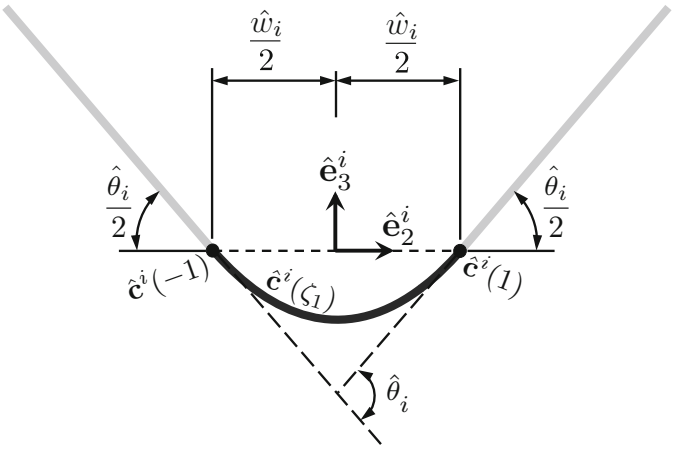


Fig. 5.7 Cross-section of a *symmetric* smooth fold

Let $\hat{\mathbf{c}}^i(\zeta_1)$ be the parametric curve $\mathbf{c}^i(\zeta_1)$ expressed in the fold coordinate system of $\mathcal{F}_t^i$:

$$\hat{\mathbf{c}}^i(\zeta_1) = \begin{bmatrix} \hat{\mathbf{e}}_1^i & \hat{\mathbf{e}}_2^i & \hat{\mathbf{e}}_3^i \end{bmatrix}^\top \left[\mathbf{c}^i(\zeta_1) - \tfrac{1}{2}\big(\mathbf{c}^i(-1) + \mathbf{c}^i(1)\big)\right]. \tag{5.3}$$

If $\hat{\mathbf{c}}^i(\zeta_1)$ is at least first-order differentiable for $\zeta_1 \in [-1, 1]$, the *total arc-length of the fold cross-section*, denoted as the *fold arc-length* $\hat{s}_i$, is determined as follows:

$$\hat{s}_i = \int_{-1}^{1} \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\| \mathrm{d}\zeta_1, \tag{5.4}$$

and the *arc-length coordinate* $s(\zeta_1) \in \left[-\frac{\hat{s}_i}{2}, \frac{\hat{s}_i}{2}\right]$ is determined as follows:

$$s(\zeta_1) = -\frac{\hat{s}_i}{2} + \int_{-1}^{\zeta_1} \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta)}{\mathrm{d}\zeta} \right\| \mathrm{d}\zeta. \tag{5.5}$$

The tangent direction of the parametric curve $\hat{\mathbf{c}}^i(\zeta_1)$ is determined by the *unit tangent vector* $\mathbf{t}^i(\zeta_1) \in \mathrm{span}(\hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i)$ and is defined at the points where $\hat{\mathbf{c}}^i(\zeta_1)$ is at least first-order differentiable:

$$\mathbf{t}^i(\zeta_1) = \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \Bigg/ \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\|. \tag{5.6}$$

The *curvature* $\hat{\kappa}_i(\zeta_1)$ is defined at the points where $\hat{\mathbf{c}}^i(\zeta_1)$ is at least second-order differentiable:

$$\hat{\kappa}_i(\zeta_1) = \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} \right\| \Bigg/ \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\|^3 , \tag{5.7}$$

and the *signed curvature* $\kappa_i(\zeta_1)$ is given as follows[4]:

$$\kappa_i(\zeta_1) = \hat{\kappa}_i(\zeta_1)\ \mathrm{sgn}\left(\left(\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} \right) \cdot \hat{\mathbf{e}}_1^i \right). \tag{5.8}$$

Example 5.1 Geometry of a quadratic form for $\hat{\mathbf{c}}^i(\zeta_1)$.
Statement: Consider a fold cross-section curve $\hat{\mathbf{c}}^i(\zeta_1)$ having the following quadratic form:

$$\hat{\mathbf{c}}^i(\zeta_1) = \begin{bmatrix} 0 \\ b_{20} + b_{21}\zeta_1 + b_{22}\zeta_1^2 \\ b_{30} + b_{31}\zeta_1 + b_{32}\zeta_1^2 \end{bmatrix}, \tag{5.9}$$

where $b_{20}, b_{21}, b_{22}, b_{30}, b_{31}, b_{32} \in \mathbb{R}$. Derive expressions for the unit tangent vector, curvature, and signed curvature for the curve $\hat{\mathbf{c}}^i(\zeta_1)$ provided in (5.9).
Solution: We first proceed by calculating the first- and second-order derivatives of $\hat{\mathbf{c}}^i(\zeta_1)$ with respect to ζ_1 and their cross product, which are required to determine the unit tangent vector, curvature, and signed curvature:

$$\begin{aligned} \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} &= \begin{bmatrix} 0 \\ b_{21} + 2b_{22}\zeta_1 \\ b_{31} + 2b_{32}\zeta_1 \end{bmatrix}, \\ \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} &= \begin{bmatrix} 0 \\ 2b_{22} \\ 2b_{32} \end{bmatrix}, \\ \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} &= \begin{bmatrix} 2b_{21}b_{32} - 2b_{22}b_{31} \\ 0 \\ 0 \end{bmatrix}. \end{aligned} \tag{5.10}$$

[4]The sign function is defined as follows: $\mathrm{sgn}(y) = \begin{cases} -1; & y < 0 \\ 1; & y > 0 \\ 0; & y = 0 \end{cases}$.

The unit tangent vector $\mathbf{t}^i(\zeta_1)$ is calculated via (5.6):

$$\begin{aligned}\mathbf{t}^i(\zeta_1) &= \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \Bigg/ \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\| \\ &= \frac{1}{((b_{21}+2b_{22}\zeta_1)^2+(b_{31}+2b_{32}\zeta_1)^2)^{\frac{1}{2}}} \begin{bmatrix} 0 \\ b_{21}+2b_{22}\zeta_1 \\ b_{31}+2b_{32}\zeta_1 \end{bmatrix}.\end{aligned} \tag{5.11}$$

The curvature $\hat{\kappa}_i(\zeta_1)$ is calculated via (5.7):

$$\begin{aligned}\hat{\kappa}_i(\zeta_1) &= \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} \right\| \Bigg/ \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\|^3 \\ &= \frac{|2b_{21}b_{32}-2b_{22}b_{31}|}{((b_{21}+2b_{22}\zeta_1)^2+(b_{31}+2b_{32}\zeta_1)^2)^{\frac{3}{2}}},\end{aligned} \tag{5.12}$$

and the signed curvature $\kappa_i(\zeta_1)$ is calculated via (5.8):

$$\begin{aligned}\kappa_i(\zeta_1) &= \hat{\kappa}_i(\zeta_1)\ \operatorname{sgn}\left(\left(\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2}\right) \cdot \hat{\mathbf{e}}_1^i\right) \\ &= \frac{|2b_{21}b_{32}-2b_{22}b_{31}|\operatorname{sgn}(2b_{21}b_{32}-2b_{22}b_{31})}{((b_{21}+2b_{22}\zeta_1)^2+(b_{31}+2b_{32}\zeta_1)^2)^{\frac{3}{2}}} \\ &= \frac{2b_{21}b_{32}-2b_{22}b_{31}}{((b_{21}+2b_{22}\zeta_1)^2+(b_{31}+2b_{32}\zeta_1)^2)^{\frac{3}{2}}}.\end{aligned} \tag{5.13}$$

5.3.1 Continuity Conditions for Smooth Folds

Since the faces of the origami sheets are assumed planar and rigid, the order of geometry continuity of the sheets is manifested in the order of continuity of the smooth fold cross-sections $\hat{\mathbf{c}}^1(\zeta_1), \ldots, \hat{\mathbf{c}}^{N_\mathcal{F}}(\zeta_1)$ and the fold/face junctions at $\hat{\mathbf{c}}^1(\pm 1), \ldots, \hat{\mathbf{c}}^{N_\mathcal{F}}(\pm 1)$ (refer to Fig. 5.8). Only continuity of position is required for G^0 continuity (Fig. 5.8a). Such a case corresponds to conventional creased folds. Continuity of position and unit tangent vector is required for G^1 continuity (Fig. 5.8b) [49]. Continuity of position, unit tangent vector, and signed curvature is required for G^2 continuity (Fig. 5.8c). Structures having discontinuities of thickness and/or materials at the interface between the faces and the smooth folds would exhibit discontinuities in curvature at such a location and thus would have G^1 continuity while those without such discontinuities would have G^2 continuity.

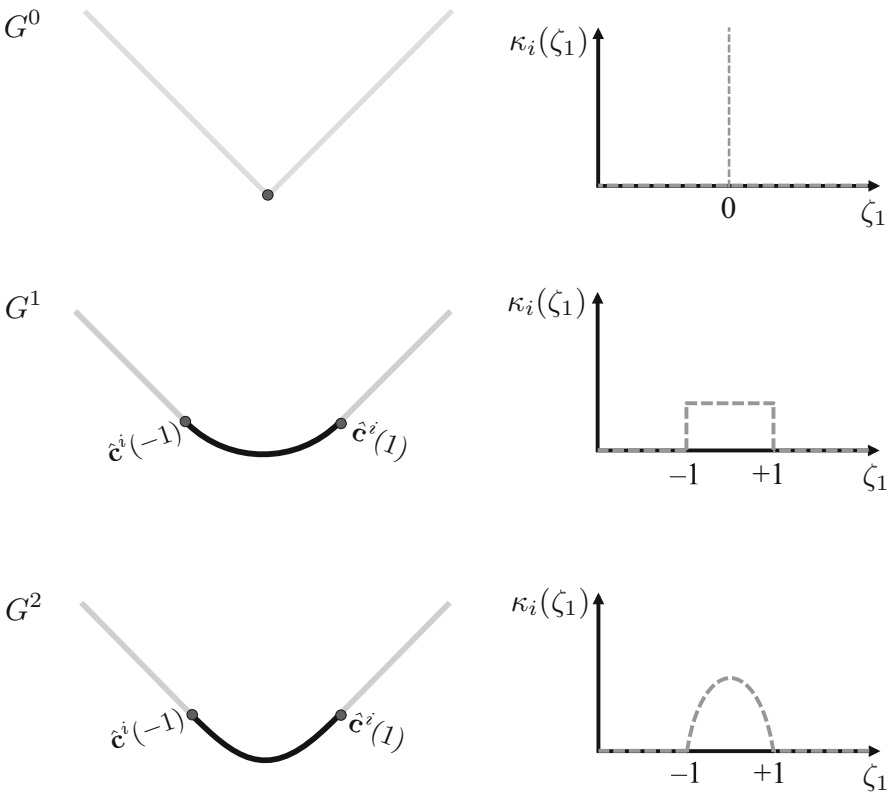


Fig. 5.8 Schematics showing folds of various orders of geometric continuity and their associated signed curvature

The order of geometric continuity of a current configuration $\mathcal{S}_t$ at the interior of a smooth fold surface $\mathcal{F}_t^i$ is determined by the order of geometric continuity of $\hat{\mathbf{c}}^i(\zeta_1)$, $\zeta_1 \in (-1, 1)$, while that at the junctions with the planar faces adjacent to $\mathcal{F}_t^i$ depends on the particular values of $\hat{\mathbf{c}}^i(\pm 1)$ and its derivatives. We provide such conditions for $\hat{\mathbf{c}}^i(\pm 1)$ and its derivatives required for G^0, G^1, and G^2 continuity in the subsequent sections.

Conditions for G^0 Continuity
For G^0 continuity at the junctions $\mathcal{F}_t^i$ and its adjacent faces (i.e., simply continuity of position), the following conditions on $\hat{\mathbf{c}}^i(\pm 1)$ must be met (refer to Fig. 5.7):

$$\hat{\mathbf{c}}^i(-1) = \begin{bmatrix} 0 \\ -\frac{\hat{w}_i}{2} \\ 0 \end{bmatrix} = \hat{\mathbf{c}}^i_{L_0}, \qquad \hat{\mathbf{c}}^i(1) = \begin{bmatrix} 0 \\ \frac{\hat{w}_i}{2} \\ 0 \end{bmatrix} = \hat{\mathbf{c}}^i_{R_0}. \tag{5.14}$$

Conditions for G^1 Continuity
To achieve G^1 continuity at the junctions of $\mathcal{F}_t^i$ and its adjacent planar faces, both continuity of the unit tangent vector $\mathbf{t}^i(\zeta_1)$ and G^0 continuity at $\zeta_1 = \pm 1$ are required [49, 50]. Referring to Fig. 5.7, the following values of $\mathbf{t}^i(\pm 1)$ are required for G^1 continuity:

$$\mathbf{t}^i(-1) = \begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix}, \qquad \mathbf{t}^i(1) = \begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ \sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix}. \tag{5.15}$$

Using (5.6), we determine that the following conditions on the first derivatives of $\hat{\mathbf{c}}^i(\pm 1)$ are required for G^1 continuity at the junctions of $\mathcal{F}_t^i$ with its adjacent planar faces:

$$\begin{aligned} \left.\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1}\right|_{\zeta_1=-1} &= \beta_{i1} \begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix} = \hat{\mathbf{c}}^i_{L_1}, \\ \left.\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1}\right|_{\zeta_1=1} &= \beta_{i1} \begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ \sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix} = \hat{\mathbf{c}}^i_{R_1}, \end{aligned} \tag{5.16}$$

where $\beta_{i1} \in \mathbb{R}_{>0}$.

Example 5.2 Conditions for G^1 continuity of a general (unsymmetrical) fold
Statement: Based on Fig. 5.6, state the conditions on the unit tangent vector $\mathbf{t}^i(\zeta_1)$ and the first derivative of $\hat{\mathbf{c}}^i(\zeta_1)$ required for G^1 continuity of a general fold for which $\hat{a}_i$ is not necessarily equal to $\frac{1}{2}$.
Solution: Referring to Fig. 5.6, continuity of the unit tangent vector $\mathbf{t}^i(\zeta_1)$ at $\zeta_1 = \pm 1$ requires the following values of $\mathbf{t}^i(\pm 1)$:

$$\mathbf{t}^i(-1) = \begin{bmatrix} 0 \\ \cos\left(\hat{a}_i\hat{\theta}_i\right) \\ -\sin\left(\hat{a}_i\hat{\theta}_i\right) \end{bmatrix}, \qquad \mathbf{t}^i(1) = \begin{bmatrix} 0 \\ \cos\left((1-\hat{a}_i)\hat{\theta}_i\right) \\ \sin\left((1-\hat{a}_i)\hat{\theta}_i\right) \end{bmatrix}. \tag{5.17}$$

Thus, the following conditions on the first derivatives of $\hat{\mathbf{c}}^i(\pm 1)$ are required for G^1 continuity:

$$\begin{aligned}
\left.\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1}\right|_{\zeta_1=-1} &= \beta_{i1}^L \begin{bmatrix} 0 \\ \cos\left(\hat{a}_i\hat{\theta}_i\right) \\ -\sin\left(\hat{a}_i\hat{\theta}_i\right) \end{bmatrix}, \\
\left.\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1}\right|_{\zeta_1=1} &= \beta_{i1}^R \begin{bmatrix} 0 \\ \cos\left((1-\hat{a}_i)\hat{\theta}_i\right) \\ \sin\left((1-\hat{a}_i)\hat{\theta}_i\right) \end{bmatrix},
\end{aligned} \tag{5.18}$$

where $\beta_{i1}^L,\ \beta_{i1}^R \in \mathbb{R}_{>0}$, cf. (5.16).

Conditions for G^2 Continuity

To achieve G^2 continuity at the junctions of $\mathcal{F}_t^i$ and its adjacent planar faces, both continuity of the signed curvature and G^1 continuity at $\zeta_1 = \pm 1$ are required. Thus, the curvature $\hat{\kappa}_i(\pm 1)$ must be zero since $\mathcal{F}_t^i$ is connected to planar faces. As such, the following is required to allow $\hat{\kappa}_i(\pm 1) = 0$ according to (5.7):

$$\begin{aligned}
\left(\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2}\right)\Bigg|_{\zeta_1=-1} &= \mathbf{0}_3, \\
\left(\frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2}\right)\Bigg|_{\zeta_1=1} &= \mathbf{0}_3,
\end{aligned} \tag{5.19}$$

where $\mathbf{0}_n$ is the zero vector in $\mathbb{R}^n$. Considering (5.16) and (5.19), the following conditions on the second derivatives of $\hat{\mathbf{c}}^i(\zeta_1)$ are needed for G^2 continuity at the junctions of $\mathcal{F}_t^i$ with its adjacent planar faces:

$$\begin{aligned}
\left.\frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2}\right|_{\zeta_1=-1} &= \beta_{i2} \begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix} = \hat{\mathbf{c}}_{L_2}^i, \\
\left.\frac{\mathrm{d}^2\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2}\right|_{\zeta_1=1} &= \beta_{i2} \begin{bmatrix} 0 \\ -\cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix} = \hat{\mathbf{c}}_{R_2}^i,
\end{aligned} \tag{5.20}$$

where $\beta_{i2} \in \mathbb{R}$. Conditions on higher-order derivatives of $\hat{\mathbf{c}}^i(\zeta_1)$ required for higher-order geometric continuity can be provided in a similar manner [49, 50].

5.3.2 *Fold Parameterization Examples*

Polynomials of the minimum order required to satisfy the previous continuity conditions for $\hat{\mathbf{c}}^i(\pm 1)$ and its derivatives are used to define such a parametric curve. We use *Hermite interpolation polynomials* [51] to represent $\hat{\mathbf{c}}^i(\zeta_1)$. Alternative representations (e.g., Bezier curves) are also applicable if they satisfy the continuity conditions. For smooth folds having G^1 continuity, $\hat{\mathbf{c}}^i(\zeta_1)$ is expressed as follows:

$$\hat{\mathbf{c}}^i(\zeta_1) = h_{30}(\zeta_1)\hat{\mathbf{c}}^i_{L_0} + h_{31}(\zeta_1)\hat{\mathbf{c}}^i_{R_0} + h_{32}(\zeta_1)\hat{\mathbf{c}}^i_{L_1} + h_{33}(\zeta_1)\hat{\mathbf{c}}^i_{R_1}, \tag{5.21}$$

where $\hat{\mathbf{c}}^i_{L_0}$ and $\hat{\mathbf{c}}^i_{R_0}$ are defined in (5.14) and $\hat{\mathbf{c}}^i_{L_1}$ and $\hat{\mathbf{c}}^i_{R_1}$ are defined in (5.16). The cubic Hermite interpolation polynomials $h_{30}(\zeta_1), \ldots, h_{33}(\zeta_1)$ are given as follows:

$$\begin{aligned}
h_{30}(\zeta_1) &= \tfrac{1}{4}\zeta_1^3 - \tfrac{3}{4}\zeta_1 + \tfrac{1}{2},\\
h_{31}(\zeta_1) &= -\tfrac{1}{4}\zeta_1^3 + \tfrac{3}{4}\zeta_1 + \tfrac{1}{2},\\
h_{32}(\zeta_1) &= \tfrac{1}{4}\zeta_1^3 - \tfrac{1}{4}\zeta_1^2 - \tfrac{1}{4}\zeta_1 + \tfrac{1}{4},\\
h_{33}(\zeta_1) &= \tfrac{1}{4}\zeta_1^3 + \tfrac{1}{4}\zeta_1^2 - \tfrac{1}{4}\zeta_1 - \tfrac{1}{4}.
\end{aligned} \tag{5.22}$$

For smooth folds having G^2 continuity, $\hat{\mathbf{c}}^i(\zeta_1)$ is expressed as follows:

$$\begin{aligned}
\hat{\mathbf{c}}^i(\zeta_1) = {} & h_{50}(\zeta_1)\hat{\mathbf{c}}^i_{L_0} + h_{51}(\zeta_1)\hat{\mathbf{c}}^i_{R_0} + h_{52}(\zeta_1)\hat{\mathbf{c}}^i_{L_1}\\
& + h_{53}(\zeta_1)\hat{\mathbf{c}}^i_{R_1} + h_{54}(\zeta_1)\hat{\mathbf{c}}^i_{L_2} + h_{55}(\zeta_1)\hat{\mathbf{c}}^i_{R_2},
\end{aligned} \tag{5.23}$$

where $\hat{\mathbf{c}}^i_{L_2}$ and $\hat{\mathbf{c}}^i_{R_2}$ are defined in (5.20). The fifth-order Hermite interpolation polynomials $h_{50}(\zeta_1), \ldots, h_{55}(\zeta_1)$ are given as follows:

$$\begin{aligned}
h_{50}(\zeta_1) &= -\tfrac{3}{16}\zeta_1^5 + \tfrac{5}{8}\zeta_1^3 - \tfrac{15}{16}\zeta_1 + \tfrac{1}{2},\\
h_{51}(\zeta_1) &= \tfrac{3}{16}\zeta_1^5 - \tfrac{5}{8}\zeta_1^3 + \tfrac{15}{16}\zeta_1 + \tfrac{1}{2},\\
h_{52}(\zeta_1) &= -\tfrac{3}{16}\zeta_1^5 + \tfrac{1}{16}\zeta_1^4 + \tfrac{5}{8}\zeta_1^3 - \tfrac{3}{8}\zeta_1^2 - \tfrac{7}{16}\zeta_1 + \tfrac{5}{16},\\
h_{53}(\zeta_1) &= -\tfrac{3}{16}\zeta_1^5 - \tfrac{1}{16}\zeta_1^4 + \tfrac{5}{8}\zeta_1^3 + \tfrac{3}{8}\zeta_1^2 - \tfrac{7}{16}\zeta_1 - \tfrac{5}{16},\\
h_{54}(\zeta_1) &= -\tfrac{1}{16}\zeta_1^5 + \tfrac{1}{16}\zeta_1^4 + \tfrac{1}{8}\zeta_1^3 - \tfrac{1}{8}\zeta_1^2 - \tfrac{1}{16}\zeta_1 + \tfrac{1}{16},\\
h_{55}(\zeta_1) &= \tfrac{1}{16}\zeta_1^5 + \tfrac{1}{16}\zeta_1^4 - \tfrac{1}{8}\zeta_1^3 - \tfrac{1}{8}\zeta_1^2 + \tfrac{1}{16}\zeta_1 + \tfrac{1}{16}.
\end{aligned} \tag{5.24}$$

The reader is asked in Problem 5.5 to verify that the parameterization of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ in (5.21) satisfies the conditions for G^1 continuity. Likewise, Problem 5.6 asks to verify that the parameterization of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ in (5.23) satisfies the conditions for G^2 continuity.

5.4 Fold Pattern Description

In this section, we examine the geometric description of the fold pattern, which is the layout of the smooth folds in the reference configuration $\mathcal{S}_0$ of an origami sheet. We consider the Miura-Ori sheet previously introduced for origami with creased folds in Fig. 2.5 to help the reader visualize the concepts introduced in this section. The reference configuration $\mathcal{S}_0$ of the Miura-Ori sheet having smooth folds is shown in Fig. 5.9.

The *centerlines* of the smooth folds are defined by their end points, denoted as *vertices* (refer to Fig. 5.9a). Each vertex has an associated position vector denoted $\mathbf{v}^j \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ (see Fig. 5.9b). The number of vertices located at the interior of $\mathcal{S}_0$ is denoted $N_{\mathcal{I}}$ and the number of vertices located at the boundary or outside $\mathcal{S}_0$ is denoted $N_{\mathcal{B}}$. For the Miura-Ori sheet in Fig. 5.9, $N_{\mathcal{I}} = 1$ and $N_{\mathcal{B}} = 4$.

The vertices are enumerated starting from those located at the interior of $\mathcal{S}_0$ (with corresponding position vectors $\mathbf{v}^1, \ldots, \mathbf{v}^{N_{\mathcal{I}}}$) followed by those located at the boundary or outside $\mathcal{S}_0$ (with corresponding position vectors $\mathbf{v}^{N_{\mathcal{I}}+1}, \ldots, \mathbf{v}^{N_{\mathcal{I}}+N_{\mathcal{B}}}$). We illustrate this convention in Fig. 5.9a. The position vectors $\mathbf{v}^1, \ldots, \mathbf{v}^5$ for the Miura-Ori sheet are provided in Example 2.1.

To identify which vertices are the start points and end points of each fold centerline in the fold pattern, we introduce the *fold connectivity matrix* $\mathbf{C}^{\mathcal{F}} \in \mathbb{R}^{N_{\mathcal{F}} \times 2}$ with components $C^{\mathcal{F}}_{ij}$ defined as follows:

$$
\begin{aligned}
&C^{\mathcal{F}}_{i1} = \text{Index of the vertex corresponding to the start point of the } i\text{th fold centerline,}\\
&C^{\mathcal{F}}_{i2} = \text{Index of the vertex corresponding to the end point of the } i\text{th fold centerline,}\\
&i = 1, \ldots, N_{\mathcal{F}}.
\end{aligned}
\tag{5.25}
$$

The choice for the start point and end point of a given fold centerline can be swapped and the same fold pattern would result. However, such a choice must be consistent throughout the various input parameters required to define the fold pattern, which are presented subsequently in this section. The fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$ for the Miura-Ori sheet shown in Fig. 5.9 is provided in Example 2.2.

Let $\hat{\mathbf{v}}^{i1}, \hat{\mathbf{v}}^{i2} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, $i = 1, \ldots, N_{\mathcal{F}}$, be the position vectors of the vertices from which each fold centerline in the sheet emanates and ends, respectively (see Fig. 5.10). These vectors are determined from the fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$ as follows:

$$
\hat{\mathbf{v}}^{i1} = \mathbf{v}^{C^{\mathcal{F}}_{i1}}, \qquad \hat{\mathbf{v}}^{i2} = \mathbf{v}^{C^{\mathcal{F}}_{i2}} \qquad\qquad i = 1, \ldots, N_{\mathcal{F}}. \tag{5.26}
$$

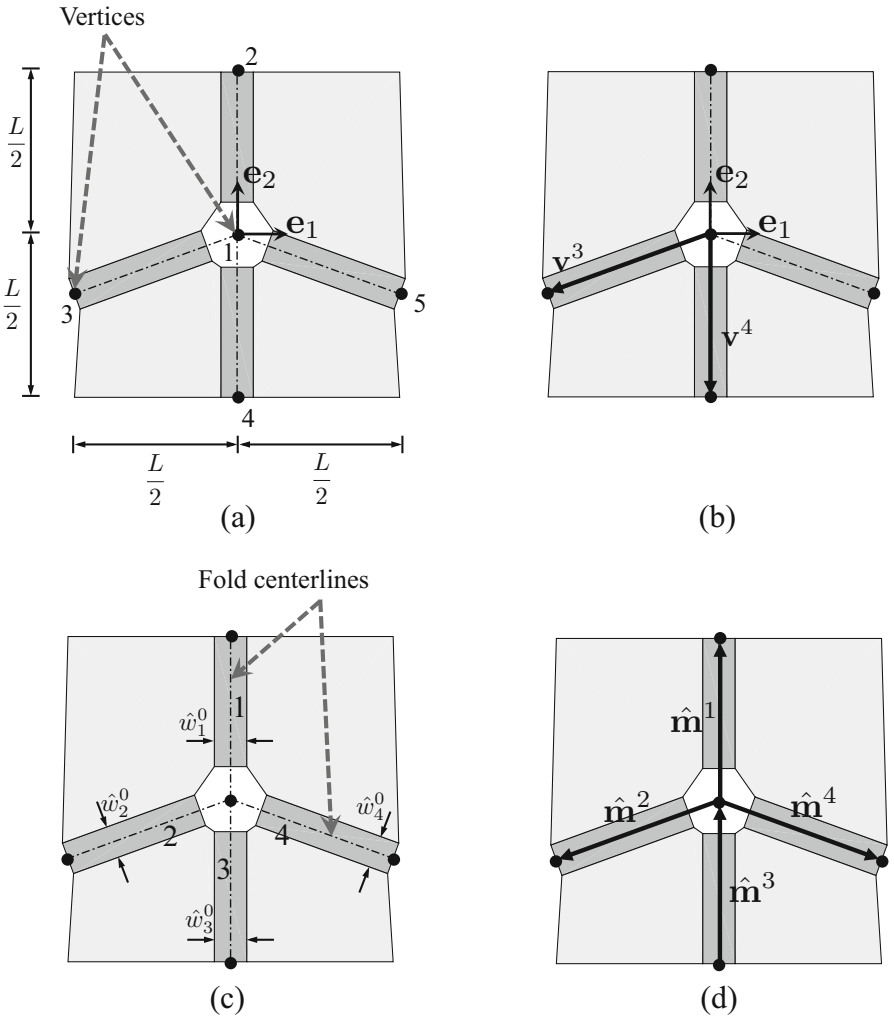


Fig. 5.9 Reference configuration $\mathcal{S}_0$ of a sheet with smooth folds having the Miura-Ori pattern (its counterpart with creased folds is shown in Fig. 2.5): (**a**) Numbering of the vertices and dimensions of the sheet; (**b**) Position vectors of the 3rd and 4th vertices; (**c**) Numbering of the folds. The *fold widths* $\hat{w}_1^0, \ldots \hat{w}_4^0$ are also shown; (**d**) Fold vectors along the length of each fold centerline (the choice for the start point and end point of a given fold vector can be swapped and the same fold pattern would result, but the choice must be consistent throughout the various input parameters required to define the fold pattern)

Let $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_\mathcal{F}} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the *fold vectors*, which are those along the length of each fold centerline. These vectors are determined as follows (see Fig. 5.9d):

$$\hat{\mathbf{m}}^i = \hat{\mathbf{v}}^{i2} - \hat{\mathbf{v}}^{i1} \qquad i = 1, \ldots, N_\mathcal{F}. \tag{5.27}$$

The vectors $\hat{\mathbf{v}}^{11}, \ldots, \hat{\mathbf{v}}^{41}, \hat{\mathbf{v}}^{12}, \ldots, \hat{\mathbf{v}}^{42}$, and $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^4$ for the Miura-Ori sheet shown in Fig. 5.9 are provided in Example 2.3.

The four corner points of $\mathcal{F}_0^i$ having associated position vectors $\hat{\mathbf{p}}^{i1}, \hat{\mathbf{p}}^{i2}, \hat{\mathbf{p}}^{i3}, \hat{\mathbf{p}}^{i4} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, $i = 1, \ldots, N_{\mathcal{F}}$, are determined as follows:

$$\begin{aligned}
\hat{\mathbf{p}}^{i1} &= \hat{\mathbf{v}}^{i1} - \frac{\hat{w}_i^0}{2}\left(\mathbf{e}_3 \times \frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}\right) + \hat{r}_{i1}\frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}, \\
\hat{\mathbf{p}}^{i2} &= \hat{\mathbf{v}}^{i2} - \frac{\hat{w}_i^0}{2}\left(\mathbf{e}_3 \times \frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}\right) - \hat{r}_{i2}\frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}, \\
\hat{\mathbf{p}}^{i3} &= \hat{\mathbf{v}}^{i2} + \frac{\hat{w}_i^0}{2}\left(\mathbf{e}_3 \times \frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}\right) - \hat{r}_{i2}\frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}, \\
\hat{\mathbf{p}}^{i4} &= \hat{\mathbf{v}}^{i1} + \frac{\hat{w}_i^0}{2}\left(\mathbf{e}_3 \times \frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}\right) + \hat{r}_{i1}\frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|},
\end{aligned} \tag{5.28}$$

where $\hat{r}_{i1}, \hat{r}_{i2} \in \mathbb{R}$ are the distances between the end points of the fold centerline and the boundary of the fold as shown in Fig. 5.10.

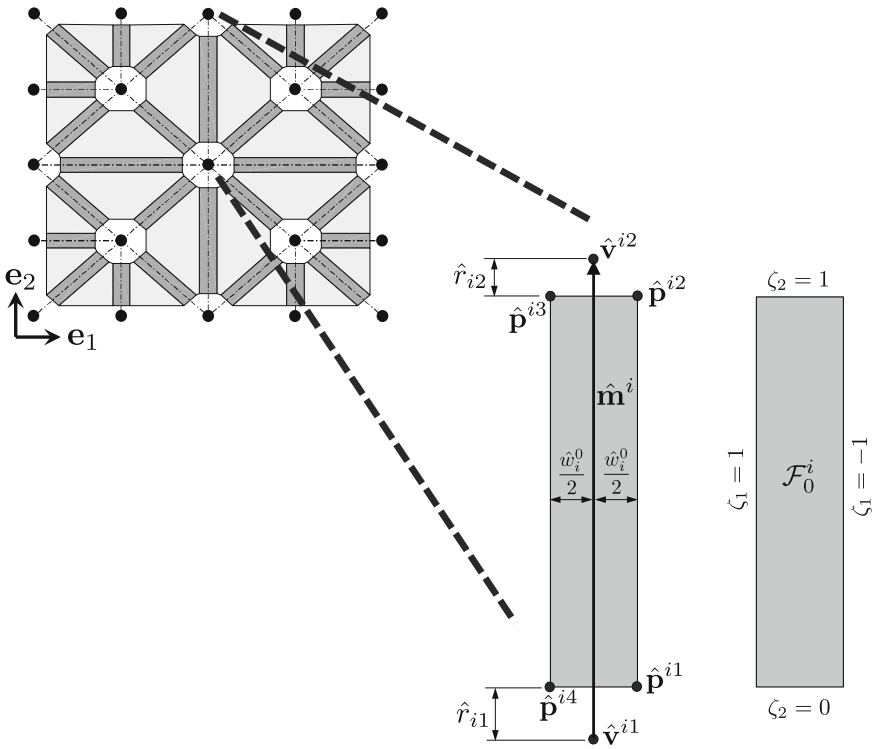


Fig. 5.10 Geometric parameters defining any $\mathcal{F}_0^i$: Coordinates of the end points of the fold centerline $\hat{\mathbf{v}}^{i1}$ and $\hat{\mathbf{v}}^{i2}$, fold vector $\hat{\mathbf{m}}^i$, fold width $\hat{w}_i^0$, and length parameters $\hat{r}_{i1}$ and $\hat{r}_{i2}$

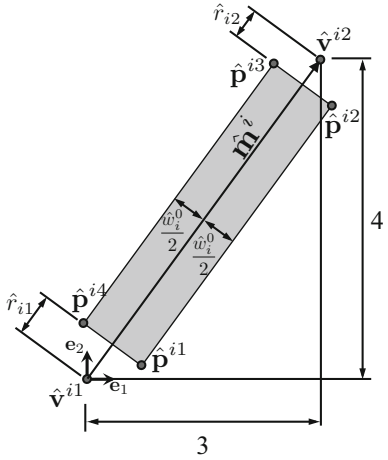


Fig. 5.11 Schematic for Example 5.3: Geometric parameters of $\mathcal{F}_0^i$

Example 5.3 Determination of fold corner points.
Statement: Consider the reference configuration of a smooth fold shown in Fig. 5.11. Let $\hat{w}_i^0 = 1$, $\hat{r}_{i1} = \frac{3}{5}$, $\hat{r}_{i2} = \frac{2}{5}$, and $\hat{\mathbf{v}}^{i1} = [\ 0\ \ 0\ \ 0]^\top$. Determine $\hat{\mathbf{p}}^{i2}$.
Solution: Based on Fig. 5.11, the vectors $\hat{\mathbf{m}}^i$ and $\hat{\mathbf{v}}^{i2}$ are given as:

$$\hat{\mathbf{m}}^i = \begin{bmatrix} 3 \\ 4 \\ 0 \end{bmatrix}, \qquad \hat{\mathbf{v}}^{i2} = \begin{bmatrix} 3 \\ 4 \\ 0 \end{bmatrix}. \tag{5.29}$$

Therefore:

$$\frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|} = \begin{bmatrix} \frac{3}{5} \\ \frac{4}{5} \\ 0 \end{bmatrix}, \qquad \mathbf{e}_3 \times \frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|} = \begin{bmatrix} -\frac{4}{5} \\ \frac{3}{5} \\ 0 \end{bmatrix}. \tag{5.30}$$

We can then determine $\hat{\mathbf{p}}^{i2}$ via (5.28):

$$\begin{aligned}\hat{\mathbf{p}}^{i2} &= \hat{\mathbf{v}}^{i2} - \frac{\hat{w}_i^0}{2}\left(\mathbf{e}_3 \times \frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|}\right) - \hat{r}_{i2}\frac{\hat{\mathbf{m}}^i}{\|\hat{\mathbf{m}}^i\|} \\ &= \begin{bmatrix} 3 \\ 4 \\ 0 \end{bmatrix} - \frac{1}{2}\begin{bmatrix} -\frac{4}{5} \\ \frac{3}{5} \\ 0 \end{bmatrix} - \frac{2}{5}\begin{bmatrix} \frac{3}{5} \\ \frac{4}{5} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \frac{79}{25} \\ \frac{169}{50} \\ 0 \end{bmatrix}.\end{aligned} \tag{5.31}$$

The reader can similarly determine the position vectors of the other three corners ($\hat{\mathbf{p}}^{i1}$, $\hat{\mathbf{p}}^{i3}$, and $\hat{\mathbf{p}}^{i4}$) of the fold shown in Fig. 5.11.

Let $n_1, \ldots, n_{N_\mathcal{I}}$ be the number of fold centerlines incident to each *interior fold intersection* of $\mathcal{S}_0$ (corresponding to the holes associated with interior vertices having position vectors $\mathbf{v}^1, \ldots, \mathbf{v}^{N_\mathcal{I}}$). Also, let $\mathbf{m}^{jk} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, $j = 1, \ldots, N_\mathcal{I}$, $k = 1, \ldots, n_j$, be the vector along the length of the kth fold centerline incident to the jth interior fold intersection that emanates from such an intersection. The vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ associated with the interior fold intersection of the Miura-Ori sheet shown in Fig. 5.9 are illustrated in Fig. 5.12a.

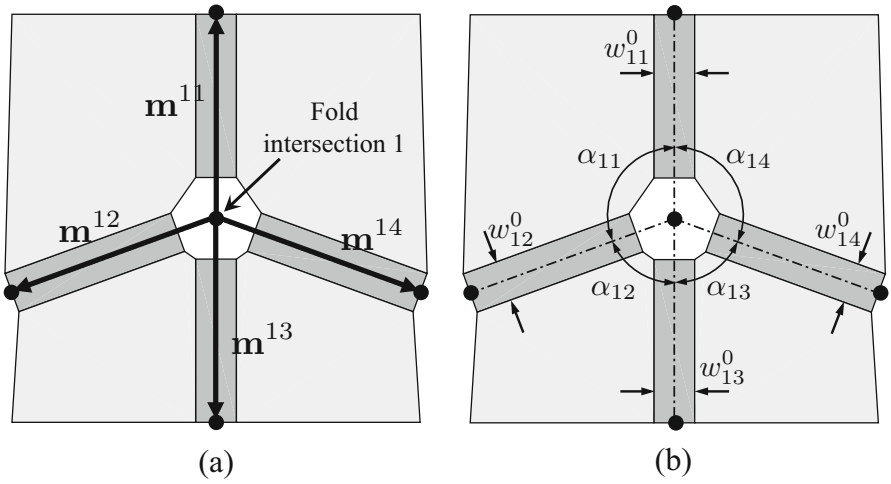


Fig. 5.12 Parameters associated with the interior fold intersection of the Miura-Ori sheet: (**a**) Vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ along the length of the fold centerlines incident to the interior fold intersection and that emanate from the intersection; (**b**) Face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ around the interior fold intersection and widths $w_{11}^0, \ldots, w_{14}^0$ of the folds adjacent to the fold intersection

For the jth interior fold intersection, the associated vectors $\mathbf{m}^{j1}, \ldots, \mathbf{m}^{jn_j}$ are arranged in counterclockwise order (see Figs. 5.12 and 5.13). The *fold intersection connectivity matrix* $\mathbf{C}^{\mathcal{I}} \in \mathbb{R}^{N_{\mathcal{I}} \times \max(n_j)}$ is used for the identification and ordering of the folds adjacent to the interior fold intersections and its components $C^{\mathcal{I}}_{jk}$ are defined follows:

$$\begin{aligned} C^{\mathcal{I}}_{jk} = &\text{ Index of the } k\text{th fold adjacent to the } j\text{th interior fold intersection} \\ &\text{(multiplied by } -1 \text{ if the fold centerline ends at the interior fold} \\ &\text{intersection),} \\ j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots n_j. \end{aligned} \tag{5.32}$$

The mapping from the fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_{\mathcal{F}}}$ (see (5.27)) to the vectors $\mathbf{m}^{j1}, \ldots, \mathbf{m}^{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$, is given as follows:

$$\mathbf{m}^{jk} = \begin{cases} \hat{\mathbf{m}}^{C^{\mathcal{I}}_{jk}}; & C^{\mathcal{I}}_{jk} > 0 \\ -\hat{\mathbf{m}}^{|C^{\mathcal{I}}_{jk}|}; & C^{\mathcal{I}}_{jk} < 0, \end{cases} \tag{5.33}$$

$$j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots, n_j.$$

Note that $\mathbf{m}^{jk}$ has the opposite direction of its associated fold vector $\hat{\mathbf{m}}^{|C^{\mathcal{I}}_{jk}|}$ if $C^{\mathcal{I}}_{jk} < 0$. This is applied such that the vectors $\mathbf{m}^{jk}$ always emanate from the interior fold intersection (since $C^{\mathcal{I}}_{jk} < 0$ if the considered fold centerline *ends* at the interior fold intersection).

The face corner angles surrounding each interior fold intersection are denoted as $\alpha_{j1}, \ldots, \alpha_{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$, and are calculated as follows (see Fig. 5.13)[5]:

$$\alpha_{jk} = \begin{cases} \varphi(\mathbf{m}^{j\,k+1}, \mathbf{m}^{jk}); & k = 1, \ldots, n_j - 1 \\ \varphi(\mathbf{m}^{j1}, \mathbf{m}^{jk}); & k = n_j. \end{cases} \tag{5.34}$$

The face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ surrounding the interior fold intersection of the Miura-Ori sheet are illustrated in Fig. 5.12b. The vectors $\mathbf{m}^{11}, \ldots, \mathbf{m}^{14}$ and the face corner angles $\alpha_{11}, \ldots, \alpha_{14}$ for the interior fold intersection of the Miura-Ori sheet are also calculated in Example 2.4.

The fold widths associated with the smooth folds adjacent to the jth interior fold intersection are denoted $w^0_{j1}, \ldots, w^0_{jn_j}$, $j = 1, \ldots, N_{\mathcal{I}}$. The mapping from the

[5]Recalling (2.13), $\varphi(\mathbf{y}, \mathbf{z})$ is defined as the angle from a vector $\mathbf{z} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ to a vector $\mathbf{y} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$. To simplify the notation, if only one argument is provided in the function φ, it is implicitly assumed that $\mathbf{z} = \mathbf{e}_1$ (i.e., $\varphi(\mathbf{y}) = \varphi(\mathbf{y}, \mathbf{e}_1)$).

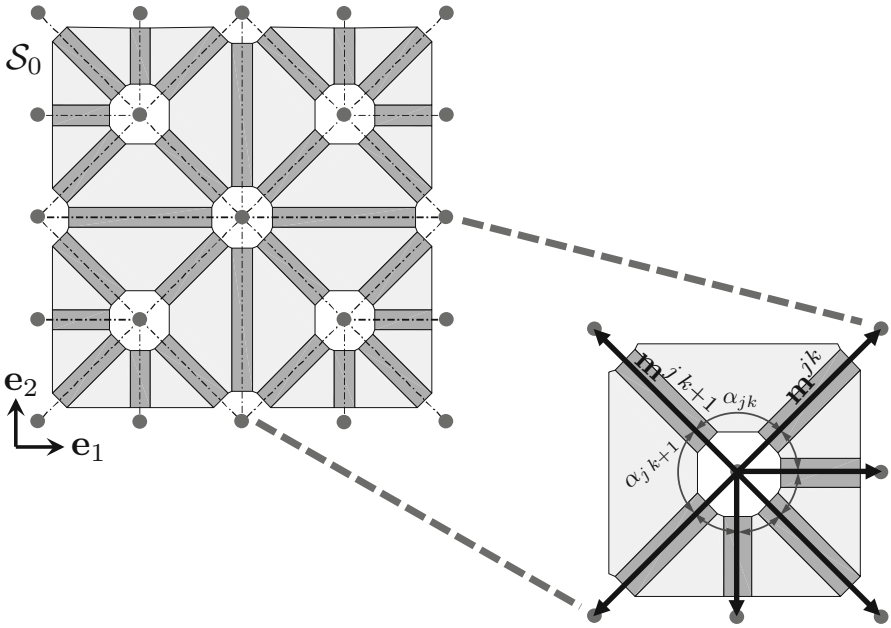


Fig. 5.13 Schematic showing faces and smooth folds connected to an interior fold intersection and their associated geometric parameters. The overall domain $\mathcal{S}_0$ has 5 interior vertices and 16 outer vertices ($N_{\mathcal{I}} = 5$, $N_{\mathcal{B}} = 16$)

fold widths of each fold in the sheet ($\hat{w}^0_1, \ldots, \hat{w}^0_{N_{\mathcal{F}}}$) to $w^0_{j1}, \ldots, w^0_{jn_j}$ is given as follows:

$$w^0_{jk} = \hat{w}^0_{|C^{\mathcal{I}}_{jk}|} \qquad j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots, n_j, \tag{5.35}$$

We summarize the input and calculated parameters required to define the fold pattern of an origami sheet with smooth folds in Tables 5.1 and 5.2.

Table 5.1 Input parameters required to define the fold pattern of an origami sheet with smooth folds

Parameter	Definition
Position vectors of the vertices $\mathbf{v}^1, \ldots, \mathbf{v}^{N_{\mathcal{I}}+N_{\mathcal{B}}}$	
Fold connectivity matrix $\mathbf{C}^{\mathcal{F}}$	(5.25)
Fold intersection connectivity matrix $\mathbf{C}^{\mathcal{I}}$	(5.32)
Fold widths $\hat{w}^0_1, \ldots, \hat{w}^0_{N_{\mathcal{F}}}$	
Fold length parameters $\hat{r}_{11}, \ldots, \hat{r}_{N_{\mathcal{F}}1}, \hat{r}_{12}, \ldots, \hat{r}_{N_{\mathcal{F}}2}$	

The parameters that are not present in a sheet with creased folds are listed below the dashed line

Table 5.2 Calculated parameters required to define the fold pattern of an origami sheet with smooth folds

Parameter	Equation
Fold centerline start points $\hat{\mathbf{v}}^{11}, \ldots, \hat{\mathbf{v}}^{N_\mathcal{F}1}$	(5.26)
Fold centerline end points $\hat{\mathbf{v}}^{12}, \ldots, \hat{\mathbf{v}}^{N_\mathcal{F}2}$	(5.26)
Fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_\mathcal{F}}$	(5.27)
Vectors along fold centerlines incident to interior fold intersections $\mathbf{m}^{j1}, \ldots, \mathbf{m}^{jn_j}, j = 1, \ldots, N_\mathcal{I}$	(5.33)
Face corner angles $\alpha_{j1}, \ldots, \alpha_{jn_j}, j = 1, \ldots, N_\mathcal{I}$	(5.34)
Position vectors of the corner points of each smooth fold $\hat{\mathbf{p}}^{i1}, \hat{\mathbf{p}}^{i2}, \hat{\mathbf{p}}^{i3}, \hat{\mathbf{p}}^{i4}$, $i = 1, \ldots, N_\mathcal{F}$	(5.28)
Fold widths of the smooth folds adjacent to interior fold intersections $w^0_{j1}, \ldots, w^0_{jn_j}, j = 1, \ldots, N_\mathcal{I}$	(5.35)

The parameters that are not present for a sheet with creased folds are listed below the dashed line

5.5 Kinematic Constraints for Origami with Smooth Folds

In this section, we describe the fundamental kinematic constraints for origami with smooth folds. These are the *developability constraint* (Sect. 5.5.1) and the *loop closure constraints* (Sect 5.5.2).

5.5.1 Developability Constraint

After the geometry of the fold pattern is defined, constraints on the fold kinematic variables are formulated such that every current configuration attained by an origami sheet is valid according to Sect. 5.2.[6] In addition to constraints allowing for valid configurations (to be addressed in Sect. 5.5.2), the condition of *developability* [52] is also conventionally imposed in origami. We refer the reader to Sect. 2.4.1 for a brief discussion on developability. Just as in origami with creased folds, the face corner angles $\alpha_{j1}, \ldots, \alpha_{jn_j}$ determined via (5.34) must satisfy the following constraint for origami with smooth folds:

$$K_j = 2\pi - \sum_{k=1}^{n_j} \alpha_{jk} = 0 \qquad j = 1, \ldots, N_\mathcal{I}, \tag{5.36}$$

cf. (2.18). In the model presented in this chapter, the face corner angles α_{jk} are defined in the reference configuration $\mathcal{S}_0$, which is planar and free of face

[6] As stated in Sect. 2.4, self-intersection avoidance is not considered in this work.

overlaps. Therefore, these angles sum to 2π for each interior fold intersection and the developability constraint (5.36) is satisfied in $\mathcal{S}_0$. No further consideration of this constraint is required because the face corner angles are constant during the folding motion of the sheet (since the faces undergo only rigid deformations for valid configurations) and hence they hold their associated values α_{jk} as defined in $\mathcal{S}_0$.

5.5.2 *Loop Closure Constraints*

In this section, we formulate constraints on the fold kinematic variables allowing for valid configurations in origami structures with smooth folds. We first derive the *mapping* from the reference to current configurations considering only the faces and folds adjacent to an interior fold intersection following the process presented in Sect. 2.4.2. This represents a kind of "local" case where only the faces and folds adjacent to a single interior fold intersection of a more complex fold pattern are examined. The goal is to determine the constraints that allow any such set of faces and folds to attain valid configurations.

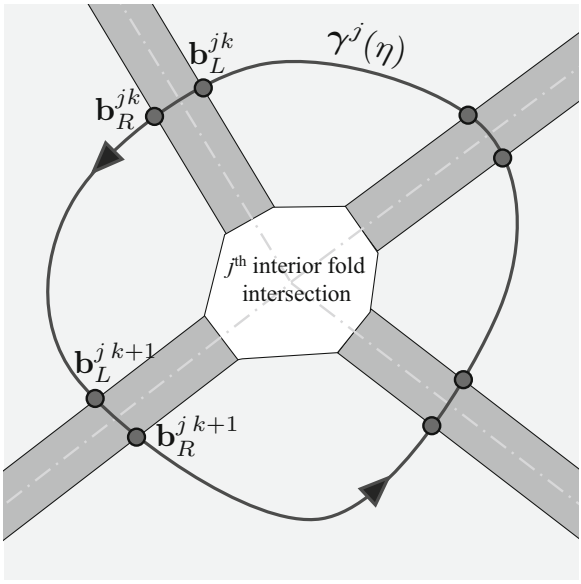
The kinematic variables describing the deformation of a smooth fold correspond to (1) The fold angle $\hat{\theta}_i$ and (2) The distance $\hat{w}_i$ between the end points of the cross-section curve $\hat{\mathbf{c}}^i(\zeta_1)$. These variables are illustrated in Fig. 5.7. The variables θ_{jk} and w_{jk} are those associated with the kth smooth fold adjacent to the jth interior fold intersection and are determined as follows:

$$\theta_{jk} = \hat{\theta}_{|C^{\mathcal{I}}_{jk}|}, \quad w_{jk} = \hat{w}_{|C^{\mathcal{I}}_{jk}|} \qquad j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots, n_j, \tag{5.37}$$

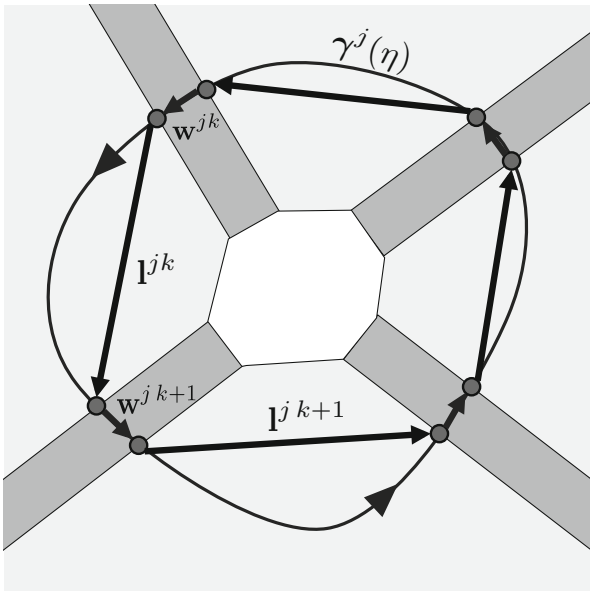
where components of the fold intersection connectivity matrix $\mathbf{C}^{\mathcal{I}}$ are defined in (5.32).

Let $\boldsymbol{\gamma}^j(\eta) : [0, 1] \rightarrow \mathcal{S}_0$ be an arbitrary simple closed path enclosing the jth interior fold intersection and crossing each of its adjacent smooth folds once as illustrated in Fig. 5.14. The point having position $\boldsymbol{\gamma}^j(0) = \boldsymbol{\gamma}^j(1)$ is defined such that it is located in the face adjacent to the smooth folds with corresponding vectors $\mathbf{m}^{j1}$ and $\mathbf{m}^{jn_j}$. Also, the path $\boldsymbol{\gamma}^j(\eta)$ is defined such that it crosses the folds with associated vectors $\mathbf{m}^{jk}$ in counterclockwise order (i.e., $\mathbf{m}^{j1}, \mathbf{m}^{j2}, \ldots, \mathbf{m}^{jn_j}$). The position vector of the point where path $\boldsymbol{\gamma}^j(\eta)$ *enters* the kth smooth fold is denoted $\mathbf{b}_L^{jk} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ and the position vector of the point where $\boldsymbol{\gamma}^j(\eta)$ *exits* such a fold is denoted $\mathbf{b}_R^{jk} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ as shown in Fig. 5.14a.

The selection of the vectors $\mathbf{b}_L^{jk}, \mathbf{b}_R^{jk}$, $k = 1, \ldots, n_j$, is not unique and depends on the particular choice of the arbitrary path $\boldsymbol{\gamma}^j(\eta)$ (see Fig. 5.14a). The corner points of the smooth folds defined in (5.28) provide a simple choice for the points where $\boldsymbol{\gamma}^j(\eta)$ crosses each fold adjacent to the jth interior fold intersection. Thus, they are used here to define $\mathbf{b}_L^{jk}, \mathbf{b}_R^{jk}, k = 1, \ldots, n_j$:



(a)



(b)

Fig. 5.14 (**a**) Path $\boldsymbol{\gamma}^j(\eta)$ crossing the faces and the smooth folds adjacent to the jth interior fold intersection. (**b**) Vectors $\mathbf{w}^{jk}$ and $\mathbf{l}^{jk}$ having start points and end points corresponding to the points where the path $\boldsymbol{\gamma}^j(\eta)$, respectively, crosses the folds and the faces

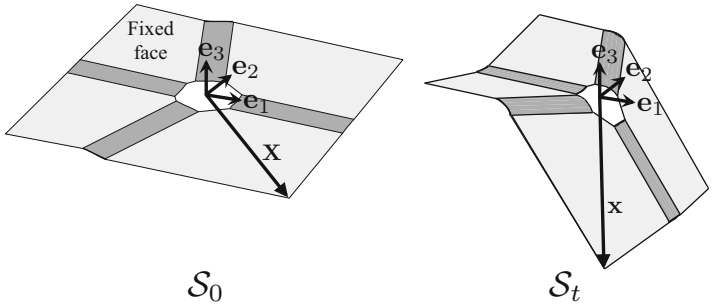


Fig. 5.15 Position vector $\mathbf{X}$ of a point in the reference configuration $\mathcal{S}_0$ and position vector $\mathbf{x}$ of the same point in a current configuration $\mathcal{S}_t$

$$\mathbf{b}_L^{jk} = \begin{cases} \hat{\mathbf{p}}^{C_{jk}^{\mathcal{I}}1}; & C_{jk}^{\mathcal{I}} > 0 \\ \hat{\mathbf{p}}^{|C_{jk}^{\mathcal{I}}|4}; & C_{jk}^{\mathcal{I}} < 0 \end{cases}, \qquad \mathbf{b}_R^{jk} = \begin{cases} \hat{\mathbf{p}}^{C_{jk}^{\mathcal{I}}4}; & C_{jk}^{\mathcal{I}} > 0 \\ \hat{\mathbf{p}}^{|C_{jk}^{\mathcal{I}}|1}; & C_{jk}^{\mathcal{I}} < 0 \end{cases} \tag{5.38}$$

$$j = 1, \ldots, N_{\mathcal{I}}, \quad k = 1, \ldots, n_j.$$

Note in (5.38) that we account for whether the interior vertex $\mathbf{v}^j$ in the jth interior fold intersection is the start point (case $C_{jk}^{\mathcal{I}} > 0$) or the end point (case $C_{jk}^{\mathcal{I}} < 0$) of the centerline of the adjacent smooth fold (see (5.32) and Fig. 5.10).

The following assumptions are made to simplify the derivation of kinematic constraints for the case of a single interior fold intersection: (1) the vertex at the fold intersection is assumed to be located at the origin and (2) the face containing the point with position $\boldsymbol{\gamma}^j(0)$ is assumed *fixed* in space (not translating or rotating). The position vector of a point in a face in the reference configuration $\mathcal{S}_0$ is denoted $\mathbf{X} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ and the position vector of the same point in a current configuration $\mathcal{S}_t$ is denoted $\mathbf{x} \in \mathbb{R}^3$ as illustrated in Fig. 5.15.

To define the mapping between reference and current configurations of the faces crossed by the path $\boldsymbol{\gamma}^j(\eta)$ (i.e., the map $\mathbf{X} \mapsto \mathbf{x}$), we must recall the definitions of certain transformation matrices examined in Sect. 2.5.2. The matrix $\mathbf{T}(\mathbf{b}) \in \mathbb{R}^{4\times4}$ represents the transformation associated with a translation by vector $\mathbf{b} \in \mathbb{R}^3$ in homogeneous coordinates (see (2.53)). Also, $\mathbf{Q}_1(\phi) \in \mathbb{R}^{4\times4}$ and $\mathbf{Q}_3(\phi) \in \mathbb{R}^{4\times4}$, respectively, represent the transformations associated with rotations about axes aligned with $\mathbf{e}_1$ and $\mathbf{e}_3$ (see (2.55) and (2.57)). We summarize the definitions of these transformation matrices as follows:

$$\mathbf{T}(\mathbf{b}) = \begin{bmatrix} 1 & 0 & 0 & b_1 \\ 0 & 1 & 0 & b_2 \\ 0 & 0 & 1 & b_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{b} \\ \mathbf{0}_3^\top & 1 \end{bmatrix},$$

$$\mathbf{Q}_1(\phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) & 0 \\ 0 & \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1(\phi) & \mathbf{0}_3 \\ \mathbf{0}_3^\top & 1 \end{bmatrix}, \tag{5.39}$$

$$\mathbf{Q}_3(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 & 0 \\ \sin(\phi) & \cos(\phi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_3(\phi) & \mathbf{0}_3 \\ \mathbf{0}_3^\top & 1 \end{bmatrix},$$

where $\mathbf{R}_1(\phi)$ is defined in (2.21), $\mathbf{R}_3(\phi)$ is defined in (2.22), and $\mathbf{I}_n$ is the identity matrix in $\mathbb{R}^{n\times n}$. Considering an axis that crosses a point with position vector $\mathbf{b} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ and is aligned to a vector $\mathbf{y} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$, the transformation associated with a rotation by ϕ about such an axis can be represented as follows [53, 54]:

$$\mathbf{T}(\mathbf{b})\,\mathbf{Q}_3(\varphi(\mathbf{y}))\,\mathbf{Q}_1(\phi)\,\mathbf{Q}_3^{-1}(\varphi(\mathbf{y}))\,\mathbf{T}^{-1}(\mathbf{b}). \tag{5.40}$$

In this previous transformation, the axis of rotation is translated to the origin via the translation matrix $\mathbf{T}^{-1}(\mathbf{b})$. The axis of rotation is then aligned to $\mathbf{e}_1$ via the rotation matrix $\mathbf{Q}_3^{-1}(\varphi(\mathbf{y}))$. Then, a rotation of ϕ about $\mathbf{e}_1$ is performed via $\mathbf{Q}_1(\phi)$. Finally, the axis of rotation is aligned back to its original orientation via $\mathbf{Q}_3(\varphi(\mathbf{y}))$ and translated to its original position via $\mathbf{T}(\mathbf{b})$.

We use (5.40) to provide a matrix formulation for the transformations associated with the folds adjacent to an interior fold intersection. We illustrate the steps of such a transformation in Fig. 5.16 and they are enumerated as follows:

1. Rotation by $\frac{\theta_{jk}}{2}$ about an axis aligned to $\mathbf{m}^{jk}$ and crossing a point with position vector $\mathbf{b}_R^{jk}$. We represent this transformation via (5.40) as follows:

$$\mathbf{T}\left(\mathbf{b}_R^{jk}\right)\,\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\,\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_R^{jk}\right). \tag{5.41}$$

2. Translation of magnitude $-\left(w_{jk}^0 - w_{jk}\right)$ in the direction of $\mathbf{e}_3 \times \frac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}$. This transformation is represented as follows:

$$\mathbf{T}^{-1}\left(\left(w_{jk}^0 - w_{jk}\right)\left(\mathbf{e}_3 \times \tfrac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}\right)\right). \tag{5.42}$$

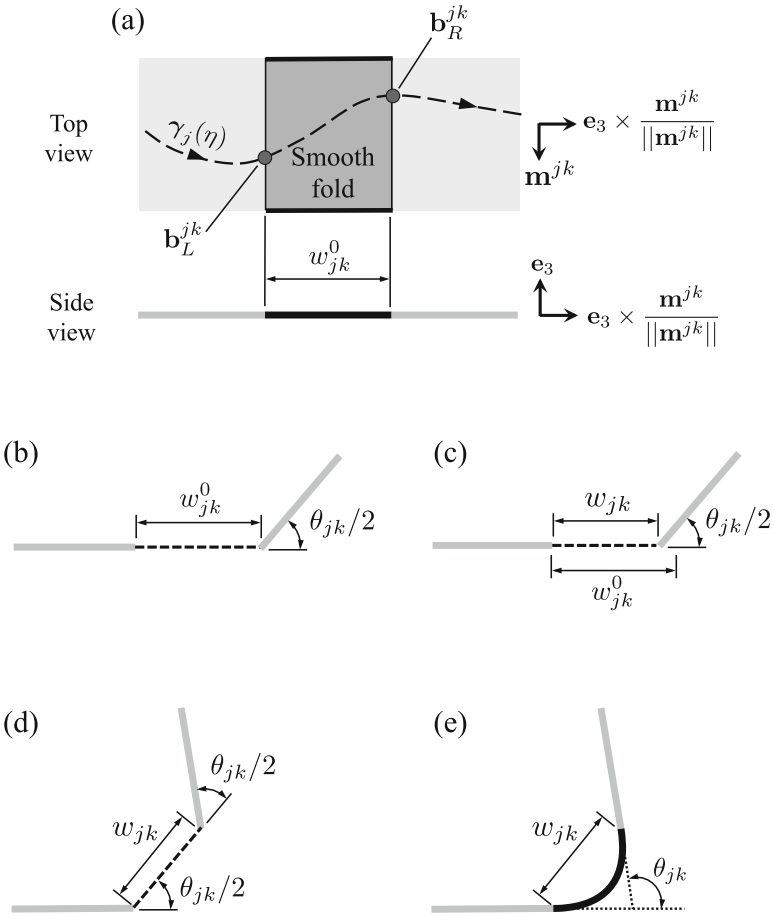


Fig. 5.16 Illustration of the transformation associated with folding the kth smooth fold crossed by $\boldsymbol{\gamma}^j(\eta)$. (**a**) Reference configuration of the fold. (**b**) Rotation by $\frac{\theta_{jk}}{2}$ about an axis aligned to $\mathbf{m}^{jk}$ and crossing a point with position vector $\mathbf{b}_R^{jk}$. (**c**) Translation of magnitude $-(w_{jk}^0 - w_{jk})$ in the direction of $\mathbf{e}_3 \times \frac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}$. (**d**) Rotation by $\frac{\theta_{jk}}{2}$ about an axis aligned to $\mathbf{m}^{jk}$ and crossing a point with position vector $\mathbf{b}_L^{jk}$. (**e**) Resulting configuration of the smooth fold and its adjacent faces

3. Rotation by $\frac{\theta_{jk}}{2}$ about an axis aligned to $\mathbf{m}^{jk}$ and crossing a point with position vector $\mathbf{b}_L^{jk}$. We represent this transformation via (5.40) as follows:

$$\mathbf{T}\left(\mathbf{b}_L^{jk}\right)\,\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\,\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_L^{jk}\right). \tag{5.43}$$

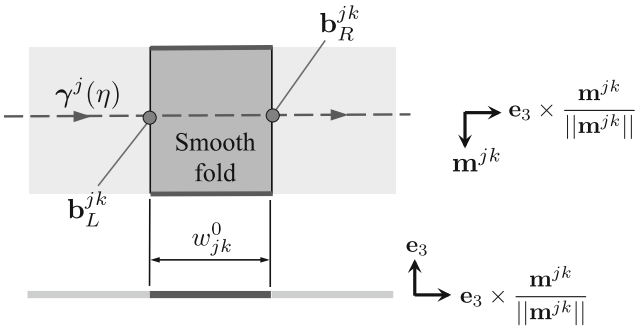


Fig. 5.17 Schematic for Example 5.4. The components of the shown vectors are given as follows: $\mathbf{m}^{jk} = [\ 0\ \ -1\ \ 0]^\top$, $\mathbf{b}_L^{jk} = [\ 0\ 0\ 0]^\top$, $\mathbf{b}_R^{jk} = \begin{bmatrix} w_{jk}^0 & 0 & 0 \end{bmatrix}^\top$

We multiply the transformation matrices of (5.41)–(5.43) to determine the matrix $\mathbf{H}^{jk}$ representing the transformation associated with folding the kth smooth fold adjacent to the jth interior fold intersection:

$$\begin{aligned}\mathbf{H}^{jk} = \mathbf{T}\left(\mathbf{b}_L^{jk}\right)\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_L^{jk}\right)\\ \times \mathbf{T}^{-1}\left(\left(w_{jk}^0 - w_{jk}\right)\left(\mathbf{e}_3 \times \tfrac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}\right)\right)\\ \times \mathbf{T}\left(\mathbf{b}_R^{jk}\right)\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_R^{jk}\right).\end{aligned} \tag{5.44}$$

Example 5.4 Determination of a matrix $\mathbf{H}^{jk}$.
Statement: Determine the transformation matrix $\mathbf{H}^{jk}$ associated with the smooth fold shown in Fig. 5.17.
Solution: With the information given in Fig. 5.17, we calculate the following matrices from (5.41)–(5.43):

$$\begin{aligned}&\mathbf{T}\left(\mathbf{b}_R^{jk}\right)\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_R^{jk}\right)\\ &= \mathbf{T}([\ 0\ 0\ 0]^\top)\,\mathbf{Q}_3\left(\tfrac{3\pi}{2}\right)\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}\left(\tfrac{3\pi}{2}\right)\mathbf{T}^{-1}([\ 0\ 0\ 0]^\top)\\ &= \begin{bmatrix} \cos\left(\frac{\theta_{jk}}{2}\right) & 0 & -\sin\left(\frac{\theta_{jk}}{2}\right) & 0\\ 0 & 1 & 0 & 0\\ \sin\left(\frac{\theta_{jk}}{2}\right) & 0 & \cos\left(\frac{\theta_{jk}}{2}\right) & 0\\ 0 & 0 & 0 & 1 \end{bmatrix},\end{aligned} \tag{5.45}$$

$$\mathbf{T}^{-1}\left(\left(w_{jk}^0 - w_{jk}\right)\left(\mathbf{e}_3 \times \frac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}\right)\right) = \mathbf{T}^{-1}\left(\left(w_{jk}^0 - w_{jk}\right)[\,1\;\;0\;\;0]^\top\right)$$

$$= \begin{bmatrix} 1 & 0 & 0 & -w_{jk}^0 + w_{jk} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{5.46}$$

$$\mathbf{T}\left(\mathbf{b}_L^{jk}\right)\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_L^{jk}\right)$$

$$= \mathbf{T}\left(\begin{bmatrix} w_{jk}^0 & 0 & 0 \end{bmatrix}^\top\right)\mathbf{Q}_3\left(\tfrac{3\pi}{2}\right)\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}\left(\tfrac{3\pi}{2}\right)\mathbf{T}^{-1}\left(\begin{bmatrix} w_{jk}^0 & 0 & 0 \end{bmatrix}^\top\right)$$

$$= \begin{bmatrix} \cos\left(\frac{\theta_{jk}}{2}\right) & 0 & -\sin\left(\frac{\theta_{jk}}{2}\right) & -w_{jk}^0\cos\left(\frac{\theta_{jk}}{2}\right) + w_{jk}^0 \\ 0 & 1 & 0 & 0 \\ \sin\left(\frac{\theta_{jk}}{2}\right) & 0 & \cos\left(\frac{\theta_{jk}}{2}\right) & -w_{jk}^0\sin\left(\frac{\theta_{jk}}{2}\right) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.47}$$

We then substitute (5.45)–(5.47) into (5.44) to determine $\mathbf{H}^{jk}$:

$$\mathbf{H}^{jk} = \mathbf{T}\left(\mathbf{b}_L^{jk}\right)\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_L^{jk}\right)$$
$$\times\,\mathbf{T}^{-1}\left(\left(w_{jk}^0 - w_{jk}\right)\left(\mathbf{e}_3 \times \frac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}\right)\right)$$
$$\times\,\mathbf{T}\left(\mathbf{b}_R^{jk}\right)\mathbf{Q}_3(\varphi(\mathbf{m}^{jk}))\,\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\mathbf{m}^{jk}))\,\mathbf{T}^{-1}\left(\mathbf{b}_R^{jk}\right)$$

$$= \mathbf{T}([\,0\;\;0\;\;0]^\top)\,\mathbf{Q}_3\left(\tfrac{3\pi}{2}\right)\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}\left(\tfrac{3\pi}{2}\right)\mathbf{T}^{-1}([\,0\;\;0\;\;0]^\top)$$
$$\times\,\mathbf{T}^{-1}\left(\left(w_{jk}^0 - w_{jk}\right)[\,1\;\;0\;\;0]^\top\right)$$
$$\times\,\mathbf{T}\left(\begin{bmatrix} w_{jk}^0 & 0 & 0 \end{bmatrix}^\top\right)\mathbf{Q}_3\left(\tfrac{3\pi}{2}\right)\mathbf{Q}_1\left(\tfrac{\theta_{jk}}{2}\right)\mathbf{Q}_3^{-1}\left(\tfrac{3\pi}{2}\right)\mathbf{T}^{-1}\left(\begin{bmatrix} w_{jk}^0 & 0 & 0 \end{bmatrix}^\top\right)$$

$$= \begin{bmatrix} \cos(\theta_{jk}) & 0 & -\sin(\theta_{jk}) & w_{jk}\cos\left(\frac{\theta_{jk}}{2}\right) - w_{jk}^0\cos(\theta_{jk}) \\ 0 & 1 & 0 & 0 \\ \sin(\theta_{jk}) & 0 & \cos(\theta_{jk}) & w_{jk}\sin\left(\frac{\theta_{jk}}{2}\right) - w_{jk}^0\sin(\theta_{jk}) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.48}$$

As in Sect. 2.4.2, the mapping $\mathbf{X} \mapsto \mathbf{x}$ is constructed as the composition of the transformations $\mathbf{H}^{jk}$ associated with the folds crossed by the segment of path $\boldsymbol{\gamma}^j(\eta)$ that connects $\boldsymbol{\gamma}^j(0)$ to the face containing the point with initial position $\mathbf{X}$:

$$\boxed{\begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = \left(\prod_{k=1}^{n_\gamma} \mathbf{H}^{jk} \right) \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}.} \tag{5.49}$$

where n_γ *is the number of folds crossed by the segment of the path* $\boldsymbol{\gamma}^j(\eta)$ *connecting* $\boldsymbol{\gamma}^j(0)$ *and the face containing the point with position vector* $\mathbf{X}$. Note that $\mathbf{x}$ is the position vector of such a point in a current configuration fully determined by $\theta_{j1} \ldots, \theta_{jn_j}$ and $w_{j1} \ldots, w_{jn_j}$.

As noted in the discussion of Example 2.7 for origami with creased folds, the mapping between reference and current configurations must be spatially continuous to prevent tearing between the faces/folds adjacent to the jth interior fold intersection. For origami with smooth folds, this is achieved if the mapping between reference and current configurations formulated in (5.49) yields $\mathbf{x} = \mathbf{X}$ for any point in the fixed face (i.e., if we substitute $n_\gamma = 0$ or $n_\gamma = n_j$ in (5.49), then $\mathbf{x} = \mathbf{X}$). This requires the following:

$$\prod_{k=1}^{n_j} \mathbf{H}^{jk} = \mathbf{I}_4. \tag{5.50}$$

We now proceed to derive constraints on the fold kinematic variables $\theta_{j1}, \ldots, \theta_{jn_j}$ and $w_{j1}, \ldots, w_{jn_j}$ by analyzing (5.50). Using the block matrix expressions for the rotation and translation matrices given in (5.39), the transformation matrices $\mathbf{H}^{jk} \in \mathbb{R}^{4\times4}$ provided in (5.44) can be partitioned into the four blocks $[\mathbf{H}^{jk}]_{11\,\text{block}} \in \mathbb{R}^{3\times3}$, $[\mathbf{H}^{jk}]_{12\,\text{block}} \in \mathbb{R}^{3\times1}$, $[\mathbf{H}^{jk}]_{21\,\text{block}} \in \mathbb{R}^{1\times3}$, and $[\mathbf{H}^{jk}]_{22\,\text{block}} \in \mathbb{R}^{1\times1}$ as follows[7]:

$$\mathbf{H}^{jk} = \begin{bmatrix} [\mathbf{H}^{jk}]_{11\,\text{block}} & [\mathbf{H}^{jk}]_{12\,\text{block}} \\ [\mathbf{H}^{jk}]_{21\,\text{block}} & [\mathbf{H}^{jk}]_{22\,\text{block}} \end{bmatrix}, \tag{5.51}$$

where:

$$\left[\mathbf{H}^{jk}\right]_{11\,\text{block}} = \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1(\theta_{jk})\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk})), \tag{5.52}$$

$$\begin{aligned} \left[\mathbf{H}^{jk}\right]_{12\,\text{block}} = \mathbf{b}_L^{jk} &+ \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1\left(\tfrac{\theta_{jk}}{2}\right)\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{w}^{jk} \\ &- \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1(\theta_{jk})\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{b}_R^{jk}, \end{aligned} \tag{5.53}$$

[7] Consult Sect. A.4 for a review on block partitioned matrices.

$$\left[\mathbf{H}^{jk}\right]_{21\,\text{block}} = \mathbf{0}_3^{\top}, \tag{5.54}$$

$$\left[\mathbf{H}^{jk}\right]_{22\,\text{block}} = 1, \tag{5.55}$$

and the vector $\mathbf{w}^{jk}$ introduced in (5.53) is defined as follows:

$$\mathbf{w}^{jk} = \mathbf{b}_R^{jk} - \mathbf{b}_L^{jk} - \left(w_{jk}^0 - w_{jk}\right)\left(\mathbf{e}_3 \times \frac{\mathbf{m}^{jk}}{\|\mathbf{m}^{jk}\|}\right). \tag{5.56}$$

The vectors $\mathbf{w}^{jk}$ for the schematic in Fig. 5.14a are shown in Fig. 5.14b.

Utilizing (5.52)–(5.55), the matrix equation (5.50) is partitioned into four blocks. The 11 block is given as:

$$\begin{aligned}\mathbf{I}_3 &= \prod_{j=1}^{n_j} \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\mathbf{R}_1(\theta_{jk})\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk})),\\ &= \mathbf{R}_3(\varphi(\mathbf{m}^{j1}))\left(\prod_{k=1}^{n_j-1} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk})\right)\mathbf{R}_1(\theta_{jn_j})\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jn_j})),\end{aligned} \tag{5.57}$$

where the following equality was used (refer to (5.34)):

$$\mathbf{R}_3(\alpha_{jk}) = \begin{cases} \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{R}_3(\varphi(\mathbf{m}^{j\,k+1})); & k = 1, \ldots, n_j - 1 \\ \\ \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{R}_3(\varphi(\mathbf{m}^{j1})); & k = n_j. \end{cases} \tag{5.58}$$

The following is obtained by multiplying the last expression in (5.57) by $\mathbf{R}_3^{-1}\left(\varphi(\mathbf{m}^{j1})\right)$ from the left and by $\mathbf{R}_3\left(\varphi(\mathbf{m}^{j1})\right)$ from the right:

$$\begin{aligned}\mathbf{I}_3 &= \left(\prod_{k=1}^{n_j-1} \mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3(\alpha_{jk})\right)\mathbf{R}_1(\theta_{jn_j})\,\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jn_j}))\,\mathbf{R}_3(\varphi(\mathbf{m}^{j1})),\\ &= \left(\prod_{k=1}^{n_j-1} \mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3(\alpha_{jk})\right)\mathbf{R}_1(\theta_{jn_j})\,\mathbf{R}_3(\alpha_{jn_j}),\\ &= \prod_{k=1}^{n_j} \mathbf{R}_1(\theta_{jk})\,\mathbf{R}_3(\alpha_{jk}).\end{aligned} \tag{5.59}$$

The 12 block of (5.50) is given as:

$$\begin{aligned}\mathbf{0}_3 = \sum_{k=1}^{n_j} \Bigg(\Bigg(\prod_{l=1}^{k-1} \mathbf{R}_3(\varphi(\mathbf{m}^{jl}))\, \mathbf{R}_1(\theta_{jl})\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jl})) \Bigg) \\ \times \Big(\mathbf{b}_L^{jk} + \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1\left(\tfrac{\theta_{jk}}{2}\right) \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{w}^{jk} \\ - \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1(\theta_{jk})\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{b}_R^{jk} \Big) \Bigg).\end{aligned} \tag{5.60}$$

We define the following vectors to simplify the previous expression:

$$\mathbf{l}^{jk} = \begin{cases} \mathbf{b}_L^{j\,k+1} - \mathbf{b}_R^{jk};\ k = 1, \ldots, n_j - 1 \\ \mathbf{b}_L^{j1} - \mathbf{b}_R^{jk};\ k = n_j. \end{cases} \tag{5.61}$$

$$\tilde{\mathbf{w}}^{jk} = \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{w}^{jk}, \tag{5.62}$$

$$\tilde{\mathbf{l}}^{jk} = \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jk}))\mathbf{l}^{jk}. \tag{5.63}$$

Substituting the last expression of (5.59) and (5.61)–(5.63) into (5.60), the following is obtained:

$$\begin{aligned}\mathbf{0}_3 = \sum_{k=1}^{n_j} \Bigg(\Bigg(\prod_{l=1}^{k-1} \mathbf{R}_3(\varphi(\mathbf{m}^{jl}))\, \mathbf{R}_1(\theta_{jl})\, \mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{jl})) \Bigg) \\ \times \Big(\mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1\left(\tfrac{\theta_{jk}}{2}\right) \tilde{\mathbf{w}}^{jk} + \mathbf{R}_3(\varphi(\mathbf{m}^{jk}))\, \mathbf{R}_1(\theta_{jk})\tilde{\mathbf{l}}^{jk} \Big) \Bigg).\end{aligned} \tag{5.64}$$

We leave to the reader the execution of the intermediate steps in the derivation of the previous expression. Finally, the following is obtained by multiplying both sides of (5.64) by $\mathbf{R}_3^{-1}(\varphi(\mathbf{m}^{j1}))$ and simplifying using (5.58):

$$\mathbf{0}_3 = \sum_{k=1}^{n_j} \left(\left(\prod_{l=1}^{k-1} \mathbf{R}_1(\theta_{jl})\mathbf{R}_3(\alpha_{jl}) \right) \left(\mathbf{R}_1\left(\tfrac{\theta_{jk}}{2}\right) \tilde{\mathbf{w}}^{jk} + \mathbf{R}_1\left(\theta_{jk}\right) \tilde{\mathbf{l}}^{jk} \right) \right). \tag{5.65}$$

The 21 and the 22 blocks of the left-hand side of (5.50) are equal to $\mathbf{0}_3^\top$ and 1, respectively. We have obtained in (5.59) and (5.65) the kinematic constraints allowing for valid configurations in origami with smooth folds. These are the *rotation constraint*:

$$\boxed{\mathbf{R}^j = \prod_{k=1}^{n_j} \mathbf{R}_1(\theta_{jk})\mathbf{R}_3(\alpha_{jk}) = \mathbf{I}_3,} \tag{5.66}$$

and the *translation constraint*:

$$\boxed{\mathbf{d}^j = \sum_{k=1}^{n_j} \left(\left(\prod_{l=1}^{k-1} \mathbf{R}_1(\theta_{jl})\mathbf{R}_3(\alpha_{jl}) \right) \left(\mathbf{R}_1\left(\tfrac{\theta_{jk}}{2}\right) \tilde{\mathbf{w}}^{jk} + \mathbf{R}_1\left(\theta_{jk}\right) \tilde{\mathbf{l}}^{jk} \right) \right) = \mathbf{0}_3,} \tag{5.67}$$

where $j = 1, \ldots, N_{\mathcal{I}}$. It can be shown that each face in a sheet having multiple interior fold intersections undergoes a rigid deformation and no tearing occurs provided the local constraints (5.66) and (5.67) are satisfied for each interior fold intersection (refer to Problems 5.14, 5.15, and 5.16).

As stated at the beginning of this section, the kinematic variables describing the deformation of a smooth fold correspond to the fold angle $\hat{\theta}_i$ and the distance $\hat{w}_i$ between the end points of the cross-section curve $\hat{\mathbf{c}}^i(\zeta_1)$. These variables appear in the kinematic constraints (5.66) and (5.67) ($\hat{w}_i$ appears implicitly in (5.67) because $\tilde{\mathbf{w}}^{jk}$ is a function of w_{jk}; refer to (5.56) and (5.62)). For simplicity in the implementation of the proposed model, assumptions on the curvature field of $\hat{\mathbf{c}}^i(\zeta_1)$ are made such that the overall deformation of a smooth fold becomes a function of the fold angle $\hat{\theta}_i$ and the arc-length $\hat{s}_i$ (i.e., $\hat{w}_i = \hat{w}_i(\hat{\theta}_i, \hat{s}_i)$ for each fold). Such a process involves non-dimensionalization of the parametric curve $\hat{\mathbf{c}}^i(\zeta_1)$ and is outlined in the following section for the cases of smooth folds exhibiting G^1 and G^2 continuity. We also provide a set of MATLAB scripts to perform such a process in the Supplemental Material of this chapter.

Determination of Kinematic Variables for Smooth Folds

Here, we describe the process used to numerically determine the kinematic variables describing the shape of a smooth fold such that its deformation is solely defined by its *fold angle* $\hat{\theta}_i$ and *fold arc-length* $\hat{s}_i$. A set of MATLAB scripts to execute such a process is provided in the Supplemental Material of this chapter.

The distances between the end points of the fold cross-sections denoted $\hat{w}_i$, $i = 1, \ldots, N_{\mathcal{F}}$ (see Fig. 5.7), which are required to evaluate the kinematic constraints of Sect. 5.5.2, are determined from the fold angles $\hat{\theta}_i$ and fold arc-lengths $\hat{s}_i$ by deriving relations $\hat{w}_i = \hat{w}_i(\hat{\theta}_i, \hat{s}_i)$. These relations are obtained by making assumptions on the form of the curvature field in the cross-section of the smooth folds.

Smooth folds having G^1 and G^2 continuous junctions with their adjacent faces are considered. The formulations of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ describing the cross-section of these smooth folds are provided in (5.21) and (5.23), respectively. In addition to

$\hat{w}_i(\hat{\theta}_i, \hat{s}_i)$, we also determine the relations $\beta_{i1}(\hat{\theta}_i, \hat{s}_i)$, $i = 1, \ldots, N_{\mathcal{F}}$. Even though the variables β_{i1} do not appear in the constraints of Sect. 5.5.2, these variables are required to fully define the shape of smooth folds exhibiting G^1 continuity (refer to (5.16)). The relations $\beta_{i2}(\hat{\theta}_i, \hat{s}_i)$, $i = 1, \ldots, N_{\mathcal{F}}$, are further required to fully define the shape of smooth folds exhibiting G^2 continuity (refer to (5.20)).

The process used to determine the relations for the kinematic variables $\hat{w}_i(\hat{\theta}_i, \hat{s}_i)$, $\beta_{i1}(\hat{\theta}_i, \hat{s}_i)$, and $\beta_{i2}(\hat{\theta}_i, \hat{s}_i)$, $i = 1, \ldots, N_{\mathcal{F}}$, involves the following steps:

1. The parametric curve $\hat{\mathbf{c}}^i(\zeta_1)$ is non-dimensionalized to first determine relations independently from the arc-length $\hat{s}_i$
2. Curvature fields that satisfy the required order of geometric continuity (G^1 or G^2) are assumed and non-dimensionalized
3. We numerically solve for the values of the non-dimensional forms of $\hat{w}_i$, β_{i1}, and β_{i2} that satisfy the assumed curvature fields for various folded states. After this step is finished, we have a list of values for the non-dimensional forms of $\hat{w}_i$, β_{i1}, and β_{i2} as functions of fold angle $\hat{\theta}_i$
4. We derive relations for the non-dimensional forms of $\hat{w}_i$, β_{i1}, and β_{i2} as functions of $\hat{\theta}_i$ by interpolating among the list of values for such kinematic variables determined in Step 3, and
5. The relations $\hat{w}_i(\hat{\theta}_i, \hat{s}_i)$, $\beta_{i1}(\hat{\theta}_i, \hat{s}_i)$ and $\beta_{i2}(\hat{\theta}_i, \hat{s}_i)$, $i = 1, \ldots, N_{\mathcal{F}}$, are then readily obtained through the equations used for non-dimensionalization

Step 1: Non-dimensionalization of the Fold Cross-section Curve First, we non-dimensionalize the parametric curve $\hat{\mathbf{c}}^i(\zeta_1)$ as follows:

$$\bar{\mathbf{c}}^i(\zeta_1) = \frac{\hat{\mathbf{c}}^i(\zeta_1)}{\hat{s}_i}. \tag{5.68}$$

Applying this non-dimensionalization to the formulations of $\hat{\mathbf{c}}^i(\zeta_1)$ presented in (5.21) and (5.23), the following is obtained:

$$\bar{\mathbf{c}}^i(\zeta_1) = h_{30}(\zeta_1)\bar{\mathbf{c}}^i_{L_0} + h_{31}(\zeta_1)\bar{\mathbf{c}}^i_{R_0} + h_{32}(\zeta_1)\bar{\mathbf{c}}^i_{L_1} + h_{33}(\zeta_1)\bar{\mathbf{c}}^i_{R_1}, \tag{5.69}$$

$$\begin{aligned}\bar{\mathbf{c}}^i(\zeta_1) = {} & h_{50}(\zeta_1)\bar{\mathbf{c}}^i_{L_0} + h_{51}(\zeta_1)\bar{\mathbf{c}}^i_{R_0} + h_{52}(\zeta_1)\bar{\mathbf{c}}^i_{L_1} \\ & + h_{53}(\zeta_1)\bar{\mathbf{c}}^i_{R_1} + h_{54}(\zeta_1)\bar{\mathbf{c}}^i_{L_2} + h_{55}(\zeta_1)\bar{\mathbf{c}}^i_{R_2},\end{aligned} \tag{5.70}$$

where:

$$\bar{\mathbf{c}}^i_{L_0} = \frac{\hat{\mathbf{c}}^i_{L_0}}{\hat{s}_i} = \begin{bmatrix} 0 \\ -\frac{1}{2}\frac{\hat{w}_i}{\hat{s}_i} \\ 0 \end{bmatrix}, \qquad \bar{\mathbf{c}}^i_{R_0} = \frac{\hat{\mathbf{c}}^i_{R_0}}{\hat{s}_i} = \begin{bmatrix} 0 \\ \frac{1}{2}\frac{\hat{w}_i}{\hat{s}_i} \\ 0 \end{bmatrix}, \tag{5.71}$$

$$\bar{\mathbf{c}}^i_{L_1} = \frac{\hat{\mathbf{c}}^i_{L_1}}{\hat{s}_i} = \frac{\beta_{i1}}{\hat{s}_i}\begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix}, \qquad \bar{\mathbf{c}}^i_{R_1} = \frac{\hat{\mathbf{c}}^i_{R_1}}{\hat{s}_i} = \frac{\beta_{i1}}{\hat{s}_i}\begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ \sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix}, \tag{5.72}$$

$$\bar{\mathbf{c}}^i_{L_2} = \frac{\hat{\mathbf{c}}^i_{L_2}}{\hat{s}_i} = \frac{\beta_{i2}}{\hat{s}_i}\begin{bmatrix} 0 \\ \cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix}, \qquad \bar{\mathbf{c}}^i_{R_2} = \frac{\hat{\mathbf{c}}^i_{R_2}}{\hat{s}_i} = \frac{\beta_{i2}}{\hat{s}_i}\begin{bmatrix} 0 \\ -\cos\left(\frac{\hat{\theta}_i}{2}\right) \\ -\sin\left(\frac{\hat{\theta}_i}{2}\right) \end{bmatrix}. \tag{5.73}$$

Equations (5.71)–(5.73) show that the non-dimensional fold cross-section $\bar{\mathbf{c}}^i(\zeta_1)$ is a function of the non-dimensional variables $\hat{\theta}_i$, $\hat{w}_i/\hat{s}_i$, $\beta_{i1}/\hat{s}_i$, and $\beta_{i2}/\hat{s}_i$.

The *non-dimensional arc-length* of $\bar{\mathbf{c}}^i(\zeta_1)$, denoted $\bar{s}_i$, is given as follows:

$$\begin{aligned} \bar{s}_i &= \int_{-1}^{1} \left\| \frac{\mathrm{d}\bar{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\| \mathrm{d}\zeta_1 \\ &= \frac{1}{\hat{s}_i} \int_{-1}^{1} \left\| \frac{\mathrm{d}\hat{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\| \mathrm{d}\zeta_1 \\ &= \frac{\hat{s}_i}{\hat{s}_i} \\ &= 1, \end{aligned} \tag{5.74}$$

and the *non-dimensional curvature* of $\bar{\mathbf{c}}^i(\zeta_1)$, denoted $\bar{\hat{\kappa}}_i(\zeta_1)$, is given as follows:

$$\bar{\hat{\kappa}}_i(\zeta_1) = \hat{s}_i \hat{\kappa}_i(\zeta_1) = \frac{\left\| \frac{\mathrm{d}\bar{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\bar{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} \right\|}{\left\| \frac{\mathrm{d}\bar{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \right\|^3}. \tag{5.75}$$

The *non-dimensional signed curvature* of $\bar{\mathbf{c}}^i(\zeta_1)$, denoted $\bar{\kappa}_i(\zeta_1)$, is given as:

$$\begin{aligned} \bar{\kappa}_i(\zeta_1) &= \hat{s}_i \kappa_i(\zeta_1) \\ &= \bar{\hat{\kappa}}_i(\zeta_1)\ \mathrm{sgn}\left(\left(\frac{\mathrm{d}\bar{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1} \times \frac{\mathrm{d}^2\bar{\mathbf{c}}^i(\zeta_1)}{\mathrm{d}\zeta_1^2} \right) \cdot \hat{\mathbf{e}}^i_1 \right). \end{aligned} \tag{5.76}$$

Step 2: Assumptions on Curvature Fields The next step is to make assumptions on the curvature field of the fold cross-sections. For smooth folds exhibiting G^1 continuity, a *uniform* goal signed curvature field $\kappa_i^G(\zeta_1)$ is assumed:

$$\kappa_i^G(\zeta_1) = \kappa^*, \tag{5.77}$$

where κ^* is a constant. For smooth folds exhibiting G^2 continuity, a *parabolic* form for the goal signed curvature field $\kappa_i^G(\zeta_1)$ is assumed:

$$\kappa_i^G(\zeta_1) = \kappa^* \left(1 - 4\frac{s(\zeta_1)^2}{\hat{s}_i^2}\right), \tag{5.78}$$

where $s(\zeta_1)$ is the arc-length coordinate defined in (5.5). Note that $\kappa_i^G(\zeta_1) = 0$ at $s(\zeta_1) = \pm\frac{\hat{s}_i}{2}$ in (5.78) to satisfy G^2 continuity.

The non-dimensional form of the goal signed curvature field in (5.77), denoted $\bar{\kappa}_i^G(\zeta_1)$, is the following:

$$\begin{aligned} \bar{\kappa}_i^G(\zeta_1) &= \hat{s}_i \kappa_i^G(\zeta_1) \\ &= \bar{\kappa}^*, \end{aligned} \tag{5.79}$$

while the non-dimensional form of the goal signed curvature field in (5.78) is the following:

$$\begin{aligned} \bar{\kappa}_i^G(\zeta_1) &= \hat{s}_i \kappa_i^G(\zeta_1) \\ &= \bar{\kappa}^* \left(1 - 4\bar{s}(\zeta_1)^2\right), \end{aligned} \tag{5.80}$$

where:

$$\bar{\kappa}^* = \hat{s}_i \kappa^*, \tag{5.81}$$

$$\bar{s}(\zeta_1) = \frac{s(\zeta_1)}{\hat{s}_i} = -\frac{1}{2} + \int_{-1}^{\zeta_1} \left\| \frac{\mathrm{d}\bar{\mathbf{c}}^i(\zeta)}{\mathrm{d}\zeta} \right\| \mathrm{d}\zeta. \tag{5.82}$$

Plots of the non-dimensional fold cross-section curve $\bar{\mathbf{c}}^i(\zeta_1)$ for different values of $\bar{\kappa}^*$ are shown in Fig. 5.18.

Step 3–5: Numerically Solve for the Fold Cross-section Kinematic Variables Once the assumptions on the curvature field are made, the non-dimensional variables $\hat{\theta}_i$, $\hat{w}_i/\hat{s}_i$, $\beta_{i1}/\hat{s}_i$ (and $\beta_{i2}/\hat{s}_i$ if smooth folds with G^2 continuity are considered) are numerically determined to satisfy such assumptions for various folded states. For a given value of $\bar{\kappa}^*$, the error in the non-dimensional arc-length $\bar{s}_i$ (which must be

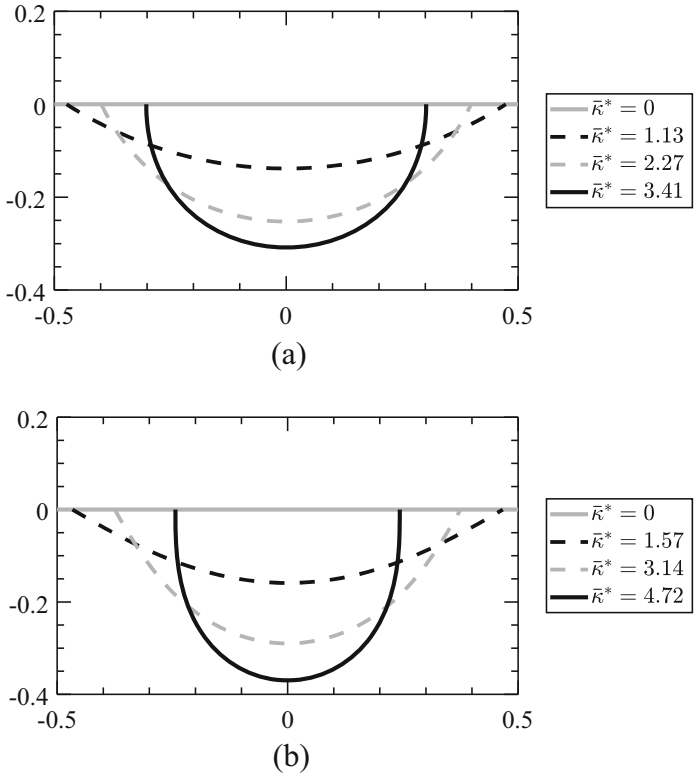


Fig. 5.18 Plots of the non-dimensional fold cross-section curve $\bar{\mathbf{c}}^i(\zeta_1)$ for different values of $\bar{\kappa}^*$: (**a**) Folds having G^1 continuity (curvature field (5.79)); (**b**) Folds having G^2 continuity (curvature field (5.80))

equal to 1, cf. (5.74)) and the error between the actual and goal curvature fields ($\bar{\kappa}_i$ and $\bar{\kappa}_i^G$, respectively) is minimized as follows:

$$\begin{aligned} &\text{Find} \qquad \hat{\theta}_i \in [0, \pi], \quad \frac{\hat{w}_i}{\hat{s}_i} \in (0, 1], \quad \frac{\beta_{i1}}{\hat{s}_i} \in (0, n], \quad \frac{\beta_{i2}}{\hat{s}_i} \in [0, n], \\ &\text{That minimize} \quad f\left(\hat{\theta}_i, \frac{\hat{w}_i}{\hat{s}_i}, \frac{\beta_{i1}}{\hat{s}_i}, \frac{\beta_{i2}}{\hat{s}_i}\right), \end{aligned} \tag{5.83}$$

where:

$$f\left(\hat{\theta}_i, \frac{\hat{w}_i}{\hat{s}_i}, \frac{\beta_{i1}}{\hat{s}_i}, \frac{\beta_{i2}}{\hat{s}_i}\right) = (1 - \bar{s}_i)^2 + \frac{\int_{-1}^{1} \left(\bar{\kappa}_i^G(\zeta_1) - \bar{\kappa}_i(\zeta_1)\right)^2 \mathrm{d}\zeta_1}{\int_{-1}^{1} \left(\bar{\kappa}_i^G(\zeta_1)\right)^2 \mathrm{d}\zeta_i}, \tag{5.84}$$

and the upper bound n for $\beta_{i1}/\hat{s}_i$ and $\beta_{i2}/\hat{s}_i$ is selected as 10, which is far from the actual values obtained for these parameters and thus it resulted in an inactive bound for the minimization problem (5.83).

The minimization problem (5.83) is repeated for different values of $\bar{\kappa}^*$ to obtain a list values of $\hat{\theta}_i$, $\hat{w}_i/\hat{s}_i$, $\beta_{i1}/\hat{s}_i$, and $\beta_{i2}/\hat{s}_i$ for various folded states (see Fig. 5.18).[8] We derive relations for the non-dimensional kinematic variables $\hat{w}_i/\hat{s}_i$, $\beta_{i1}/\hat{s}_i$, and $\beta_{i2}/\hat{s}_i$ as functions of fold angle $\hat{\theta}_i$ by interpolating among the obtained list of values. The relations $\hat{w}_i(\hat{\theta}_i, \hat{s}_i)$, $\beta_{i1}(\hat{\theta}_i, \hat{s}_i)$ and $\beta_{i2}(\hat{\theta}_i, \hat{s}_i)$, $i = 1, \ldots, N_{\mathcal{F}}$, can then be readily obtained.

5.6 Folding Map Formulation

In this section, we provide a method to determine the mapping of the position vector of any point in the sheet from the reference configuration $\mathcal{S}_0$ to a current configuration $\mathcal{S}_t$ (termed *folding map*). We follow the method outlined in Sect. 2.5 for origami with creased folds. A local folding map considering only the faces adjacent to an interior fold intersection was provided in (5.44). Such a formulation is extended here for the derivation of a folding map considering *all* the faces in the sheet. The method to determine the folding map requires the following steps:

1. A single arbitrary face in the sheet is assumed fixed in its reference configuration
2. Paths $\check{\boldsymbol{\gamma}}^j(\eta) : [0, 1] \to \mathcal{S}_0$, $j = 1, \ldots, N_{\mathcal{P}}$, connecting the fixed face to any other face with reference configuration $\mathcal{P}_0^j$, $j = 1, \ldots, N_{\mathcal{P}}$, are assumed (refer to Fig. 5.19 for examples). The paths $\check{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_{\mathcal{P}}$, may not cross any fold intersection (i.e., they travel only through faces and smooth folds of $\mathcal{S}_0$). Each path $\check{\boldsymbol{\gamma}}^j(\eta)$ crosses a number of $\check{n}_j$ smooth folds
3. The formulation of the transformation associated with folding each of the smooth folds crossed by the paths $\check{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_{\mathcal{P}}$ (termed *folding transformation* for simplicity) is determined
4. The folding map for any point in each face with reference configuration $\mathcal{P}_0^j$ is then obtained through the composition of folding transformations of the folds crossed by the corresponding path $\check{\boldsymbol{\gamma}}^j(\eta)$

Steps 1 and 2 from the previous list can be readily performed. Sections 5.6.1 and 5.6.2 provide the information required to perform Step 3. Specifically, Sect. 5.6.1 addresses the determination of the parameters required to define the folding transformation while Sect. 5.6.2 addresses the formulation of such a transformation. The formulation of the folding map for Step 4 is also addressed in Sect. 5.6.2.

[8] In our implementation, we solve the minimization problem (5.83) for each value of $\bar{\kappa}^*$ using the gradient-based optimization algorithm in MATLAB `fmincon` [55].

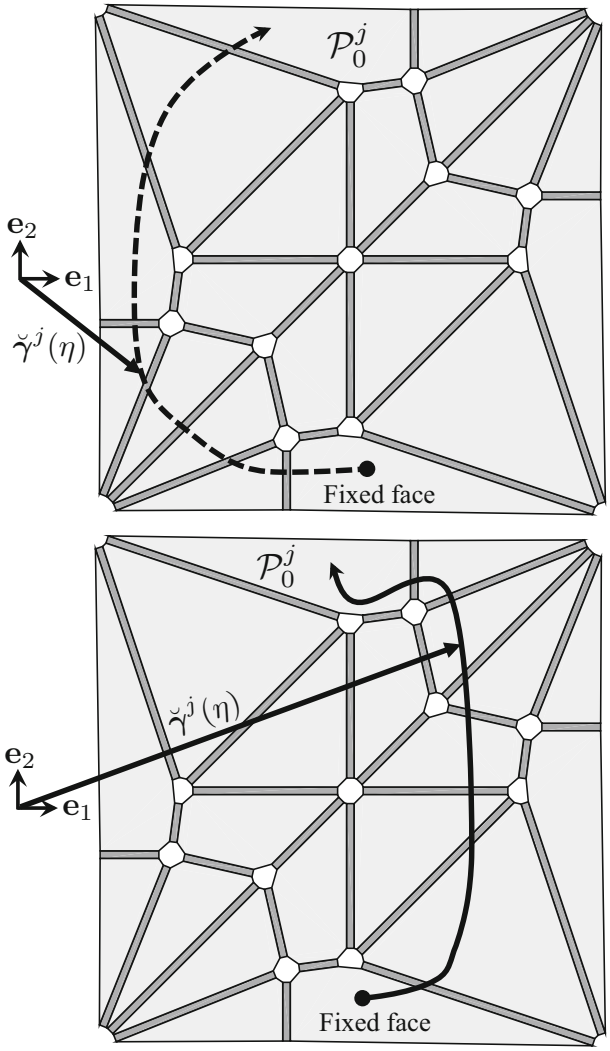


Fig. 5.19 Two equivalent paths $\breve{\boldsymbol{\gamma}}^j(\eta)$ connecting the fixed face to $\mathcal{P}_0^j$

5.6.1 Parameters Required to Derive the Folding Map

To derive the folding transformation associated with each fold crossed by a path $\breve{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_{\mathcal{P}}$, certain geometric parameters and kinematic variables associated with such folds must be determined. Our focus in this section is to provide equations for all such parameters. As shown in Fig. 5.20, a path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses the ith fold in the sheet *positively* if it crosses from the face adjacent to the fold in the direction $-\mathbf{e}_3 \times \hat{\mathbf{m}}^i$ to the face in the direction $\mathbf{e}_3 \times \hat{\mathbf{m}}^i$ (where $\hat{\mathbf{m}}^i$ is fold vector

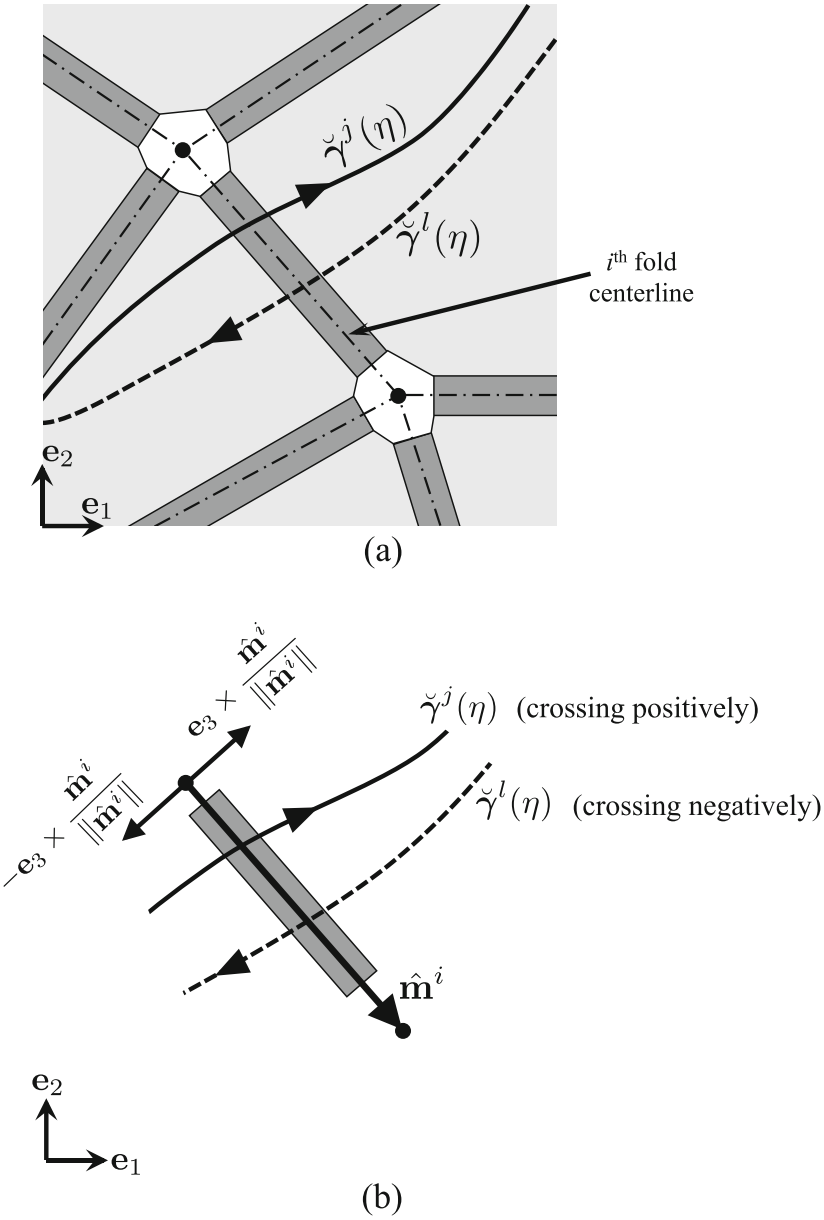


Fig. 5.20 (**a**) Two paths crossing the ith fold in $\mathcal{S}_0$. (**b**) Fold vector $\hat{\mathbf{m}}^i$ along the length of the ith fold centerline in $\mathcal{S}_0$. The path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses the fold *positively* (i.e., from the side in the direction $-\mathbf{e}_3 \times \hat{\mathbf{m}}^i$ to the side in the direction $\mathbf{e}_3 \times \hat{\mathbf{m}}^i$) and the path $\breve{\boldsymbol{\gamma}}^l(\eta)$ crosses the fold *negatively*

along the length of the ith fold centerline; see (5.27)). The path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses a fold *negatively* if it crosses such a fold in the opposite direction. The *face connectivity matrix* $\mathbf{C}^{\mathcal{P}} \in \mathbb{R}^{N_{\mathcal{P}} \times \max(\breve{n}_j)}$ with components $C^{\mathcal{P}}_{jk}$ is used for the identification and ordering of the folds crossed by each path $\breve{\boldsymbol{\gamma}}^j(\eta)$:

$$\begin{aligned} C^{\mathcal{P}}_{jk} &= \text{Index of the } k\text{th fold crossed by the path } \breve{\boldsymbol{\gamma}}^j(\eta) \\ &\quad \text{(multiplied by } -1 \text{ if the fold is } \textit{negatively} \text{ crossed by } \breve{\boldsymbol{\gamma}}^j(\eta)), \\ j &= 1, \ldots, N_{\mathcal{P}}, \quad k = 1, \ldots, \breve{n}_j. \end{aligned} \tag{5.85}$$

Let $\breve{\mathbf{m}}^{jk} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the vector along the centerline of the kth fold crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ (see Fig. 5.21b). This vector is defined such that the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses it positively (i.e., from the face in the direction $-\mathbf{e}_3 \times \breve{\mathbf{m}}^{jk}$ to the face in the direction $\mathbf{e}_3 \times \breve{\mathbf{m}}^{jk}$). The mapping from the fold vectors $\hat{\mathbf{m}}^1, \ldots, \hat{\mathbf{m}}^{N_{\mathcal{F}}}$ to the vectors $\breve{\mathbf{m}}^{j1}, \ldots, \breve{\mathbf{m}}^{j\breve{n}_j}$, $j = 1, \ldots, N_{\mathcal{P}}$, is given as follows:

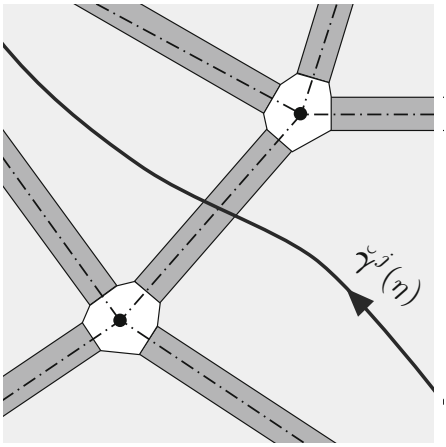
$$\begin{aligned} \breve{\mathbf{m}}^{jk} &= \begin{cases} \hat{\mathbf{m}}^{C^{\mathcal{P}}_{jk}}; & C^{\mathcal{P}}_{jk} > 0 \\ -\hat{\mathbf{m}}^{|C^{\mathcal{P}}_{jk}|}; & C^{\mathcal{P}}_{jk} < 0, \end{cases} \\ j &= 1, \ldots, N_{\mathcal{P}}, \quad k = 1, \ldots, \breve{n}_j. \end{aligned} \tag{5.86}$$

Note that $\breve{\mathbf{m}}^{jk}$ has the opposite direction of its associated fold vector $\hat{\mathbf{m}}^{|C^{\mathcal{P}}_{jk}|}$ if $C^{\mathcal{P}}_{jk} < 0$. This is applied such that the vectors $\breve{\mathbf{m}}^{jk}$ are always crossed positively by $\breve{\boldsymbol{\gamma}}^j(\eta)$.

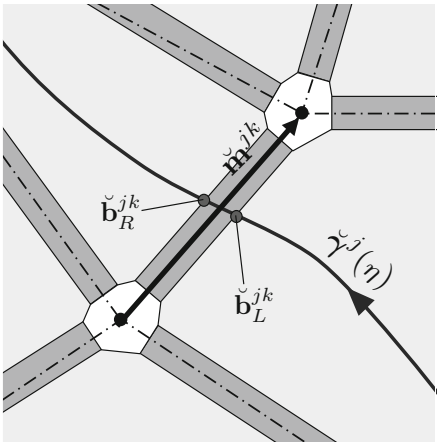
The position vector of the point where path $\breve{\boldsymbol{\gamma}}^j(\eta)$ *enters* the kth smooth fold is denoted $\breve{\mathbf{b}}^{jk}_L \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ and the position vector of the point where $\breve{\boldsymbol{\gamma}}^j(\eta)$ *exits* such a fold is denoted $\breve{\mathbf{b}}^{jk}_R \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ as illustrated in Fig. 5.21b. The selection of the vectors $\breve{\mathbf{b}}^{jk}_L, \breve{\mathbf{b}}^{jk}_R$, $k = 1, \ldots, \breve{n}_j$, is not unique and depends on the particular choice of the arbitrary path $\breve{\boldsymbol{\gamma}}^j(\eta)$. The corner points of the smooth folds defined in (5.28) provide a simple choice for the points where $\breve{\boldsymbol{\gamma}}^j(\eta)$ crosses each fold. Thus, they are used to define $\breve{\mathbf{b}}^{jk}_L, \breve{\mathbf{b}}^{jk}_R$, $k = 1, \ldots, \breve{n}_j$:

$$\begin{aligned} \breve{\mathbf{b}}^{jk}_L &= \begin{cases} \hat{\mathbf{p}}^{C^{\mathcal{P}}_{jk}1}; & C^{\mathcal{P}}_{jk} > 0 \\ \hat{\mathbf{p}}^{|C^{\mathcal{P}}_{jk}|4}; & C^{\mathcal{P}}_{jk} < 0 \end{cases}, \qquad \breve{\mathbf{b}}^{jk}_R = \begin{cases} \hat{\mathbf{p}}^{C^{\mathcal{P}}_{jk}4}; & C^{\mathcal{P}}_{jk} > 0 \\ \hat{\mathbf{p}}^{|C^{\mathcal{P}}_{jk}|1}; & C^{\mathcal{P}}_{jk} < 0 \end{cases} \\ j &= 1, \ldots, N_{\mathcal{P}}, \quad k = 1, \ldots, \breve{n}_j, \end{aligned} \tag{5.87}$$

cf. (5.38). Let $\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$ and $\breve{w}_{j1}, \ldots, \breve{w}_{j\breve{n}_j}$ be the kinematic variables associated with the ordered smooth folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$. The mapping from all the kinematic variables in the sheet ($\hat{\theta}_1, \ldots, \hat{\theta}_{N_{\mathcal{F}}}$ and $\hat{w}_1, \ldots, \hat{w}_{N_{\mathcal{F}}}$) to the



(a)



(b)

Fig. 5.21 An origami sheet illustrating parameters that define the folding transformation: (**a**) Path $\breve{\boldsymbol{\gamma}}^j(\eta)$; (**b**) Vector $\breve{\mathbf{m}}^{jk}$ along the centerline of the kth fold crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ and position vectors $\breve{\mathbf{b}}_L^{jk}$ and $\breve{\mathbf{b}}_R^{jk}$ of the points where $\breve{\boldsymbol{\gamma}}^j(\eta)$, respectively, enters and exits the fold

variables of only those crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$ and $\breve{w}_{j1}, \ldots, \breve{w}_{j\breve{n}_j}$) is given as follows:

$$\breve{\theta}_{jk} = \hat{\theta}_{|C_{jk}^{\mathcal{P}}|}, \quad \breve{w}_{jk} = \hat{w}_{|C_{jk}^{\mathcal{P}}|} \qquad j = 1, \ldots, N_{\mathcal{P}}, \quad k = 1, \ldots, \breve{n}_j. \tag{5.88}$$

The fold widths of the smooth folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ are denoted $\breve{w}_{j1}^0, \ldots, \breve{w}_{j\breve{n}_j}^0$ and are determined from the widths of all the folds in the sheet ($\hat{w}_1^0, \ldots, \hat{w}_{N_{\mathcal{F}}}^0$) as follows:

Table 5.3 Input parameters required to define the folding map of an origami sheet with smooth folds

Parameter	Definition
Fold angles $\hat{\theta}_1, \ldots, \hat{\theta}_{N_\mathcal{F}}$	
Face connectivity matrix $\mathbf{C}^\mathcal{P}$	(5.85)
Fold widths $\hat{w}_1^0, \ldots, \hat{w}_{N_\mathcal{F}}^0$	
Cross-section distance variables $\hat{w}_1, \ldots, \hat{w}_{N_\mathcal{F}}$	(5.2)

These parameters are required in addition to those associated with the fold pattern listed in Tables 5.1 and 5.2. The parameters that are not present in a sheet with creased folds are listed below the dashed line

Table 5.4 Calculated parameters required to define the folding map of an origami sheet with smooth folds

Parameter	Equation
Vectors along the centerlines of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\mathbf{m}}^{j1}, \ldots, \breve{\mathbf{m}}^{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$)	(5.86)
Fold angles of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{\theta}_{j1}, \ldots, \breve{\theta}_{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$)	(5.88)
Fold widths of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{w}_{j1}^0, \ldots, \breve{w}_{j\breve{n}_j}^0$, $j = 1, \ldots, N_\mathcal{P}$)	(5.89)
Cross-section distance variables of the folds crossed by $\breve{\boldsymbol{\gamma}}^j(\eta)$ ($\breve{w}_{j1}, \ldots, \breve{w}_{j\breve{n}_j}$, $j = 1, \ldots, N_\mathcal{P}$)	(5.88)
Position vectors of the points where $\breve{\boldsymbol{\gamma}}^j(\eta)$ enters and exits the smooth folds ($\breve{\mathbf{b}}_L^{j1}, \ldots, \breve{\mathbf{b}}_L^{j\breve{n}_j}$ and $\breve{\mathbf{b}}_R^{j1}, \ldots, \breve{\mathbf{b}}_R^{j\breve{n}_j}$, respectively; $j = 1, \ldots, N_\mathcal{P}$)	(5.87)

The parameters that are not present in a sheet with creased folds are listed below the dashed line

$$\breve{w}_{jk}^0 = \hat{w}_{|C_{jk}^\mathcal{P}|}^0 \qquad j = 1, \ldots, N_\mathcal{P}, \quad k = 1, \ldots, \breve{n}_j. \tag{5.89}$$

We summarize the input and calculated parameters required to define the folding map of an origami sheet with smooth folds in Tables 5.3 and 5.4.

5.6.2 *Folding Map Formulation*

In this section, we provide an expression for the folding transformation associated with each smooth fold crossed by the paths $\breve{\boldsymbol{\gamma}}^j(\eta)$, $j = 1, \ldots, N_\mathcal{P}$. We use (5.40) to provide a matrix formulation for such transformations. The *folding map* will then be obtained through the composition of the folding transformations associated with the folds crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$. The steps of the folding transformation are analogous to those illustrated in Fig. 5.16 and are ordered as follows:

1. Rotation by $\frac{\breve{\theta}_{jk}}{2}$ about an axis aligned to $\breve{\mathbf{m}}^{jk}$ and crossing a point with position vector $\breve{\mathbf{b}}_R^{jk}$. We represent this transformation via (5.40) as follows:

$$\mathbf{T}\left(\breve{\mathbf{b}}_R^{jk}\right)\mathbf{Q}_3(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{Q}_1\left(\frac{\breve{\theta}_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{T}^{-1}\left(\breve{\mathbf{b}}_R^{jk}\right). \tag{5.90}$$

2. Translation of magnitude $-\left(\breve{w}_{jk}^0 - \breve{w}_{jk}\right)$ in the direction of $\mathbf{e}_3 \times \frac{\breve{\mathbf{m}}^{jk}}{\|\breve{\mathbf{m}}^{jk}\|}$. This transformation is represented as follows:

$$\mathbf{T}^{-1}\left(\left(\breve{w}_{jk}^0 - \breve{w}_{jk}\right)\left(\mathbf{e}_3 \times \frac{\breve{\mathbf{m}}^{jk}}{\|\breve{\mathbf{m}}^{jk}\|}\right)\right). \tag{5.91}$$

3. Rotation by $\frac{\breve{\theta}_{jk}}{2}$ about an axis aligned to $\breve{\mathbf{m}}^{jk}$ and crossing a point with position vector $\breve{\mathbf{b}}_L^{jk}$. We represent this transformation via (5.40) as follows:

$$\mathbf{T}\left(\breve{\mathbf{b}}_L^{jk}\right)\mathbf{Q}_3(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{Q}_1\left(\frac{\breve{\theta}_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{T}^{-1}\left(\breve{\mathbf{b}}_L^{jk}\right). \tag{5.92}$$

We multiply the transformation matrices of (5.90)–(5.92) to determine the *folding transformation matrix* $\breve{\mathbf{H}}^{jk}$ associated with folding the kth smooth fold crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$:

$$\begin{aligned}\breve{\mathbf{H}}^{jk} = {} & \mathbf{T}\left(\breve{\mathbf{b}}_L^{jk}\right)\mathbf{Q}_3(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{Q}_1\left(\frac{\breve{\theta}_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{T}^{-1}\left(\breve{\mathbf{b}}_L^{jk}\right)\\ & \times \mathbf{T}^{-1}\left(\left(\breve{w}_{jk}^0 - \breve{w}_{jk}\right)\left(\mathbf{e}_3 \times \frac{\breve{\mathbf{m}}^{jk}}{\|\breve{\mathbf{m}}^{jk}\|}\right)\right)\\ & \times \mathbf{T}\left(\breve{\mathbf{b}}_R^{jk}\right)\mathbf{Q}_3(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{Q}_1\left(\frac{\breve{\theta}_{jk}}{2}\right)\mathbf{Q}_3^{-1}(\varphi(\breve{\mathbf{m}}^{jk}))\,\mathbf{T}^{-1}\left(\breve{\mathbf{b}}_R^{jk}\right).\end{aligned} \tag{5.93}$$

The *folding map* used to transform the position vector of any point in the faces of the sheet from the reference configuration $\mathcal{S}_0$ to a current configuration $\mathcal{S}_t$ is formulated here based on the fold transformation matrices $\breve{\mathbf{H}}^{jk}$ provided in (5.93).

Let $\mathbf{X} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the position vector of a point in a face with reference configuration $\mathcal{P}_0^j \subset \mathcal{S}_0$ within which the path $\breve{\boldsymbol{\gamma}}^j(\eta)$ terminates and $\mathbf{x} \in \mathbb{R}^3$ be the position vector of such a point in $\mathcal{P}_t^j \subset \mathcal{S}_t$. The mapping $\mathbf{X} \mapsto \mathbf{x}$ is determined as the multiplication of the ordered matrices $\breve{\mathbf{H}}^{jk}$ of the $\breve{n}_j$ smooth folds crossed by the path $\breve{\boldsymbol{\gamma}}^j(\eta)$:

$$\begin{bmatrix}\mathbf{x}\\ 1\end{bmatrix} = \left(\prod_{k=1}^{\breve{n}_j}\breve{\mathbf{H}}^{jk}\right)\begin{bmatrix}\mathbf{X}\\ 1\end{bmatrix}. \tag{5.94}$$

The reader is asked to show that the folding map provided in (5.94) is independent from the chosen path $\check{\boldsymbol{\gamma}}^j(\eta)$ in Problem 5.16.

5.7 Computational Implementation of the Model

Having introduced the details of the kinematic model for origami structures with smooth folds in Sects. 5.2–5.6, we continue by describing a procedure to implement the model in a computational environment. The complete set of MATLAB scripts used for the execution of such a procedure is included in the Supplemental Materials of this chapter. We describe such MATLAB scripts in Appendix B.4.

Our goal is to simulate the folding motion of an origami sheet. We start by providing the fold pattern data listed in Table 5.1. Additionally, we also have to provide information regarding the desired *history of folding motion* for the sheet. As stated in Sect. 5.5.2, the kinematic variables $\hat{w}_i$, $i = 1, \ldots, N_{\mathcal{F}}$ (see Fig. 5.7), are determined from the fold angles $\hat{\theta}_i$ and fold arc-lengths $\hat{s}_i$, $i = 1, \ldots, N_{\mathcal{F}}$, by establishing relations $\hat{w}_i(\hat{\theta}_i, \hat{s}_i)$ following certain assumptions on the curvature field of the smooth folds. For kinematic simulation, we further assume that the deformation exhibited by the smooth folds is *inextensible*:

$$\hat{s}_i = \hat{w}_i^0 \qquad \text{for } t \in [0, t_f] \qquad\qquad i = 1, \ldots, N_{\mathcal{F}}, \tag{5.95}$$

and hence the fold angles $\hat{\theta}_1, \ldots, \hat{\theta}_{N_{\mathcal{F}}}$ are the only kinematic variables considered during simulation.

For simulation in a numerical environment, the *continuous folding motion of the sheet is partitioned into increments*. Thus, the simulation of the folding motion is performed here by *incrementally* updating the values of the fold angles using input guess increments. As such, we need to provide the following data to simulate the folding motion of an origami sheet: (1) Fold pattern data (Table 5.1), and (2) Guess increments for the fold angles:

$$\Delta\hat{\theta}_1^l, \ldots, \Delta\hat{\theta}_{N_{\mathcal{F}}}^l \qquad\qquad l = 1, \ldots, N_{\text{inc}}, \tag{5.96}$$

where N_{inc} is the total number of guess increments.

We define $\Delta\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{N_{\mathcal{F}}}$ as the vector whose components correspond to the lth set of guess increments for the fold angles:

$$\Delta\hat{\boldsymbol{\theta}}^l = \begin{bmatrix} \Delta\hat{\theta}_1^l & \cdots & \Delta\hat{\theta}_{N_{\mathcal{F}}}^l \end{bmatrix}^\top \qquad\qquad l = 1, \ldots, N_{\text{inc}}. \tag{5.97}$$

Also, define $\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{N_{\mathcal{F}}}$ as the vector whose components correspond to the values of the fold angles at the lth increment:

$$\hat{\boldsymbol{\theta}}^l = \begin{bmatrix} \hat{\theta}_1^l & \cdots & \hat{\theta}_{N_{\mathcal{F}}}^l \end{bmatrix}^\top \qquad\qquad l = 1, \ldots, N_{\text{inc}}. \tag{5.98}$$

The fold angles $\hat{\boldsymbol{\theta}}^l$ must satisfy the kinematic constraints provided in (5.66) and (5.67) at every increment. Additionally, we also impose lower and upper bounds on the fold angles as in origami with creased folds (see Sect. 2.6). The lower bound and upper bound values for the ith fold angle are denoted $\hat{\theta}_i^L$ and $\hat{\theta}_i^U$, respectively. Conventional assignments for the lower and upper bounds of the fold angles are provided in Table 2.5. We define $\hat{\boldsymbol{\theta}}^L, \hat{\boldsymbol{\theta}}^U \in \mathbb{R}^{N_\mathcal{F}}$ as the vectors whose components, respectively, correspond to $\hat{\theta}_1^L, \ldots, \hat{\theta}_{N_\mathcal{F}}^L$ and $\hat{\theta}_1^U, \ldots, \hat{\theta}_{N_\mathcal{F}}^U$ as follows:

$$\hat{\boldsymbol{\theta}}^L = \begin{bmatrix} \hat{\theta}_1^L & \cdots & \hat{\theta}_{N_\mathcal{F}}^L \end{bmatrix}^\top, \qquad \hat{\boldsymbol{\theta}}^U = \begin{bmatrix} \hat{\theta}_1^U & \cdots & \hat{\theta}_{N_\mathcal{F}}^U \end{bmatrix}^\top. \tag{5.99}$$

The constraints to be satisfied at each increment are then summarized in Table 5.5.

In general, the configurations determined by the simple addition of the guess increments for the fold angles (i.e., those obtained as $\hat{\boldsymbol{\theta}}^l = \sum_{m=1}^{l} \Delta\hat{\boldsymbol{\theta}}^m$) may not satisfy the constraints listed in Table 5.5. In view of this, we use an *iterative* procedure to apply any necessary corrections such that the resulting configurations approximate the addition of the guess increments and also satisfy the constraints of Table 5.5.

The components of the vector $\hat{\boldsymbol{\theta}}^{l(k)} \in \mathbb{R}^{N_\mathcal{F}}$ correspond to the values of the fold angles at the kth *iteration* of the lth increment. The vector $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)}) \in \mathbb{R}^{6N_\mathcal{I}+2N_\mathcal{F}}$ with components $\mathfrak{R}_j(\hat{\boldsymbol{\theta}}^{l(k)})$ is the *residual vector* from constraints of Table 5.5 ($3N_\mathcal{I}$ from the rotational constraints (5.66), $3N_\mathcal{I}$ from the translational constraints (5.67), and $2N_\mathcal{F}$ from the upper and lower bounds of each fold angle).

The matrix-type constraint (5.66) provides the following three scalar constraints that are included in the components of the residual vector[9]:

Table 5.5 Constraints on the fold angles that must be satisfied at every increment (l) in the simulation of the folding motion

Rotational constraints (5.66)	$\mathbf{R}^j(\hat{\boldsymbol{\theta}}^l) = \mathbf{I}_3 \quad j = 1, \ldots, N_\mathcal{I}$
Translational constraints (5.67)	$\mathbf{d}^j(\hat{\boldsymbol{\theta}}^l) = \mathbf{0}_3 \quad j = 1, \ldots, N_\mathcal{I}$
Lower and upper bounds of the fold angles	$\hat{\boldsymbol{\theta}}^L \leq \hat{\boldsymbol{\theta}}^l \leq \hat{\boldsymbol{\theta}}^U$

[9] Since $\mathbf{R}^j$ in (5.66) is an *orthogonal* matrix, only three of its nine components are independent.

$$\begin{aligned}
\mathfrak{R}_{6j-5}(\hat{\boldsymbol{\theta}}^{l(k)}) &= \frac{1}{2}\lambda_R\left(R_{23}^j(\hat{\boldsymbol{\theta}}^{l(k)})\right)^2,\\
\mathfrak{R}_{6j-4}(\hat{\boldsymbol{\theta}}^{l(k)}) &= \frac{1}{2}\lambda_R\left(R_{31}^j(\hat{\boldsymbol{\theta}}^{l(k)})\right)^2,\\
\mathfrak{R}_{6j-3}(\hat{\boldsymbol{\theta}}^{l(k)}) &= \frac{1}{2}\lambda_R\left(R_{12}^j(\hat{\boldsymbol{\theta}}^{l(k)})\right)^2,
\end{aligned} \tag{5.100}$$

where $j = 1, \ldots, N_{\mathcal{I}}$ and λ_R is the scalar weight for residuals from kinematic constraints (5.66). This constant weight is an algorithmic parameter included in the residual vector to ensure that its components are scaled to a similar order of magnitude (i.e., the components of $\mathbf{R}^j(\hat{\boldsymbol{\theta}}^{l(k)})$ are dimensionless while those of $\mathbf{d}^j(\hat{\boldsymbol{\theta}}^{l(k)})$ have units of length). The three components of the vector $\mathbf{d}^j(\hat{\boldsymbol{\theta}}^{l(k)})$, $j = 1, \ldots, N_{\mathcal{I}}$, which must be zero for the constraint in (5.67) to be satisfied, provide the following components to the residual vector:

$$\begin{aligned}
\mathfrak{R}_{6j-2}(\hat{\boldsymbol{\theta}}^{l(k)}) &= \frac{1}{2}\lambda_d\left(d_1^j\left(\hat{\boldsymbol{\theta}}^{l(k)}\right)\right)^2,\\
\mathfrak{R}_{6j-1}(\hat{\boldsymbol{\theta}}^{l(k)}) &= \frac{1}{2}\lambda_d\left(d_2^j\left(\hat{\boldsymbol{\theta}}^{l(k)}\right)\right)^2,\\
\mathfrak{R}_{6j}(\hat{\boldsymbol{\theta}}^{l(k)}) &= \frac{1}{2}\lambda_d\left(d_3^j\left(\hat{\boldsymbol{\theta}}^{l(k)}\right)\right)^2,
\end{aligned} \tag{5.101}$$

where $j = 1, \ldots, N_{\mathcal{I}}$ and λ_d is the weight for residuals from (5.67).

The additional components of $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})$ required to enforce the lower bounds of the fold angles consist of penalties defined as follows:

$$\mathfrak{R}_{6N_{\mathcal{I}}+2i-1}(\hat{\boldsymbol{\theta}}^{l(k)}) = \frac{1}{2}\lambda_B \max\left(0,\ -\hat{\theta}_i^{l(k)} + \hat{\theta}_i^L\right)^2, \tag{5.102}$$

where $i = 1, \ldots, N_{\mathcal{F}}$ and λ_B is the scalar weight for fold angle bound constraints. Similar penalties are used to enforce the upper bounds of the fold angles and are defined as follows:

$$\mathfrak{R}_{6N_{\mathcal{I}}+2i}(\hat{\boldsymbol{\theta}}^{l(k)}) = \frac{1}{2}\lambda_B \max\left(0,\ \hat{\theta}_i^{l(k)} - \hat{\theta}_i^U\right)^2, \tag{5.103}$$

where $i = 1, \ldots, N_{\mathcal{F}}$.

For each increment, we seek to iteratively correct the fold angles such that the magnitude of the residual vector $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})$ is minimized. At the start of each increment, we first calculate the projection of the guess increments for the fold angles $\Delta\hat{\boldsymbol{\theta}}^l$ onto the null space of the residual vector derivatives of the previous

configuration (with fold angles $\hat{\boldsymbol{\theta}}^{l-1}$). This is performed to utilize the information of the previous configuration and reduce the number of iterations in the subsequent correction procedure [52]. Refer to Fig. 2.21a for an illustration of this concept. The resulting *projected* fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$ is calculated as follows (refer to (A.56)):

$$\Delta\hat{\boldsymbol{\theta}}^{l*} = \left(\mathbf{I}_{N_{\mathcal{F}}} - \left(\frac{\partial\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l-1})}{\partial\hat{\boldsymbol{\theta}}}\right)^{\dagger}\left(\frac{\partial\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l-1})}{\partial\hat{\boldsymbol{\theta}}}\right)\right)\Delta\hat{\boldsymbol{\theta}}^{l}. \tag{5.104}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse; see (A.37).

Given the fold angles of the previously determined configuration $\hat{\boldsymbol{\theta}}^{l-1}$ and the projected fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$, a configuration satisfying the kinematic constraints (5.66) and (5.67) and the fold angles bounds can be found via an iterative correction procedure. Such a procedure begins at iteration (0) where the first guess of fold angles is the addition of the previous fold angles and the projected fold angle increment:

$$\hat{\boldsymbol{\theta}}^{l(0)} = \hat{\boldsymbol{\theta}}^{l-1} + \Delta\hat{\boldsymbol{\theta}}^{l*}. \tag{5.105}$$

If $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(0)})\|/(6N_{\mathcal{I}} + 2N_{\mathcal{F}}) \geq \texttt{tol1}$ (where `tol1` is a numerical tolerance), the fold angles are corrected iteratively as follows (see Fig. 2.21b):

$$\hat{\boldsymbol{\theta}}^{l(k+1)} = \hat{\boldsymbol{\theta}}^{l(k)} + \Delta\hat{\boldsymbol{\theta}}^{l(k)}, \tag{5.106}$$

where $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ is the correction of fold angles at iteration (k). Following the generalized Newton's method [52], the first-order expansion of the residual vector $\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})$ is used to determine the correction $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ required to minimize the components of the residual vector:

$$\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)} + \Delta\hat{\boldsymbol{\theta}}^{l(k)}) = \mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)}) + \frac{\partial\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})}{\partial\hat{\boldsymbol{\theta}}}\Delta\hat{\boldsymbol{\theta}}^{l(k)} \simeq \mathbf{0}_{6N_{\mathcal{I}}+2N_{\mathcal{F}}}. \tag{5.107}$$

The following correction of fold angles $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ is obtained from the previous first-order expansion:

$$\Delta\hat{\boldsymbol{\theta}}^{l(k)} = -\left(\frac{\partial\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)})}{\partial\hat{\boldsymbol{\theta}}}\right)^{\dagger}\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k)}). \tag{5.108}$$

The iterative corrector procedure provided in (5.108) and (5.106) is repeated until $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k+1)})\|/(6N_{\mathcal{I}} + 2N_{\mathcal{F}}) < \texttt{tol1}$ or $\|\Delta\hat{\boldsymbol{\theta}}^{l(k)}\|/N_{\mathcal{F}} < \texttt{tol2}$, where `tol2`

Table 5.6 Numerical procedure used to determine valid configurations at each increment in the folding motion for origami sheets with smooth folds (lth fold angle increment)

1	Determine the projected fold angle increment $\Delta\hat{\boldsymbol{\theta}}^{l*}$ from the given $\Delta\hat{\boldsymbol{\theta}}^{l}$ using (5.104)
2	Calculate guess fold angles $\hat{\boldsymbol{\theta}}^{l(0)}$ using (5.105)
3	IF $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(0)})\|/(6N_{\mathcal{I}} + 2N_{\mathcal{F}}) <$ tol1 THEN set $\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{l(0)}$ and EXIT ELSE CONTINUE
4	Determine correction of fold angles $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ using (5.108)
5	Determine $\hat{\boldsymbol{\theta}}^{l(k+1)}$ using (5.106)
6	IF $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(k+1)})\|/(6N_{\mathcal{I}} + 2N_{\mathcal{F}}) <$ tol1 OR $\|\Delta\hat{\boldsymbol{\theta}}^{l(k)}\|/N_{\mathcal{F}} <$ tol2 THEN set $\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{l(k+1)}$ and EXIT ELSE set $k \leftarrow k + 1$ and GOTO 4

is another numerical tolerance. We summarize the numerical procedure used to determine valid configurations at each increment in the folding motion for origami sheets with smooth folds in Table 5.6.

5.8 Simulation Examples of the Kinematic Model

Here we provide representative examples of the implemented model for kinematic simulation of origami with smooth folds presented in Sects. 5.2–5.7. As stated in Sect. 5.7, the procedure for kinematic simulation of origami with smooth folds has been implemented in MATLAB. The associated MATLAB scripts are provided in the Supplemental Materials of this chapter and described in Appendix B.4. In MATLAB, we visualize the faces of origami sheets via filled three-dimensional polygons using the command `fill3` [56] and the smooth folds via three-dimensional surface plots using the command `surf` [57].

Smooth folds having G^2 continuous junctions with their adjacent faces are assumed for all the examples presented in this section. The formulation of the parametric curve $\hat{\mathbf{c}}^{i}(\zeta_1)$ that defines the cross-sectional shape of such smooth folds is provided in (5.23).

Figures 5.22, 5.23, and 5.24 illustrate examples of origami sheets that do not contain interior fold intersections. Therefore, the kinematic constraints provided in (5.66) and (5.67) do not need to be considered in these examples. The fold pattern for the so-called "self-similar wave" [58] addressed for origami with creased folds in Fig. 2.24 is shown for origami with smooth folds in Fig. 5.22. In this fold pattern, the lengths $L_1, \ldots, L_n$ (refer to Fig. 5.22) are recursively determined via (2.83). Two examples of fold patterns for self-similar waves considering $n = 2$ and $n = 3$ are shown in Fig. 5.22.

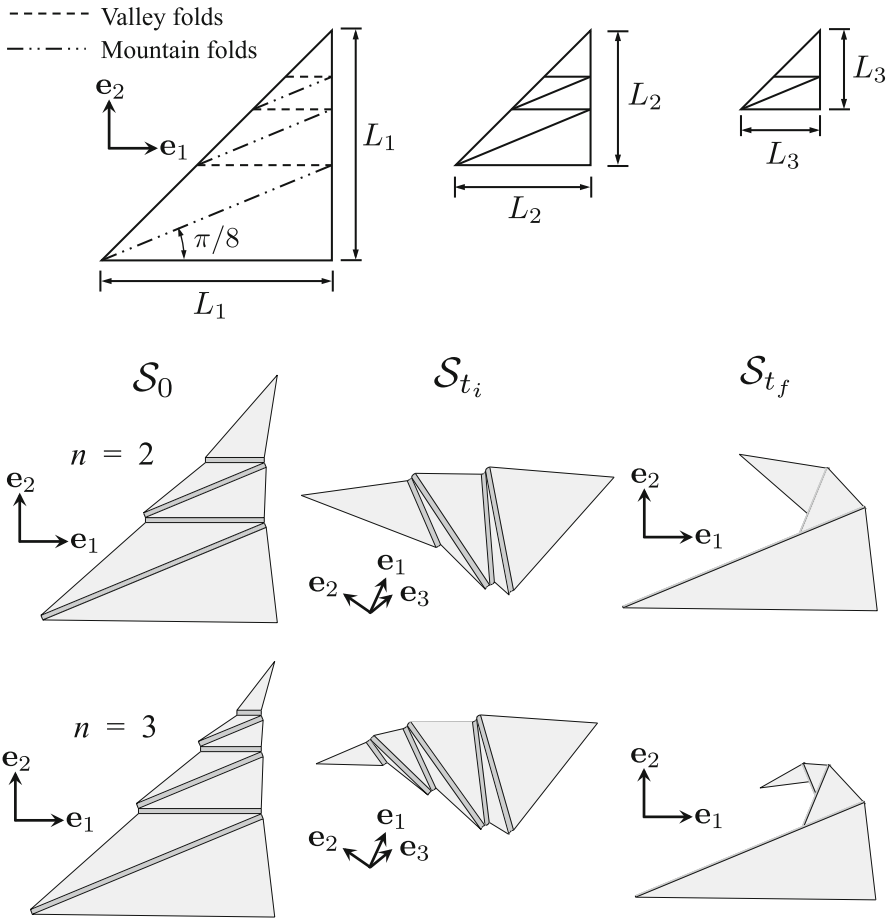


Fig. 5.22 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ for origami self-similar waves. The valley folds shown in the schematic reach a fold angle of π while the mountain folds reach a fold angle of $-\pi$

In Fig. 5.23, we show examples of origami "fans" where the fold pattern is parameterized by the radius R, the sector angle Γ, and number of sections n. Origami fans with smooth folds are shown in Fig. 5.23a and b and an origami fan with creased folds is shown in Fig. 5.23c. Note that all the faces in the fan occupy the same location when the fold angles reach $\pm\pi$ for creased folds while they are stacked at a non-zero distance from one another when considering smooth folds.

Figure 5.24 shows an origami fan illustrating the analogy between origami with smooth folds addressed herein and origami with creased folds and thick panels [59]. Note that both methods can address origami of finite thickness (because we study

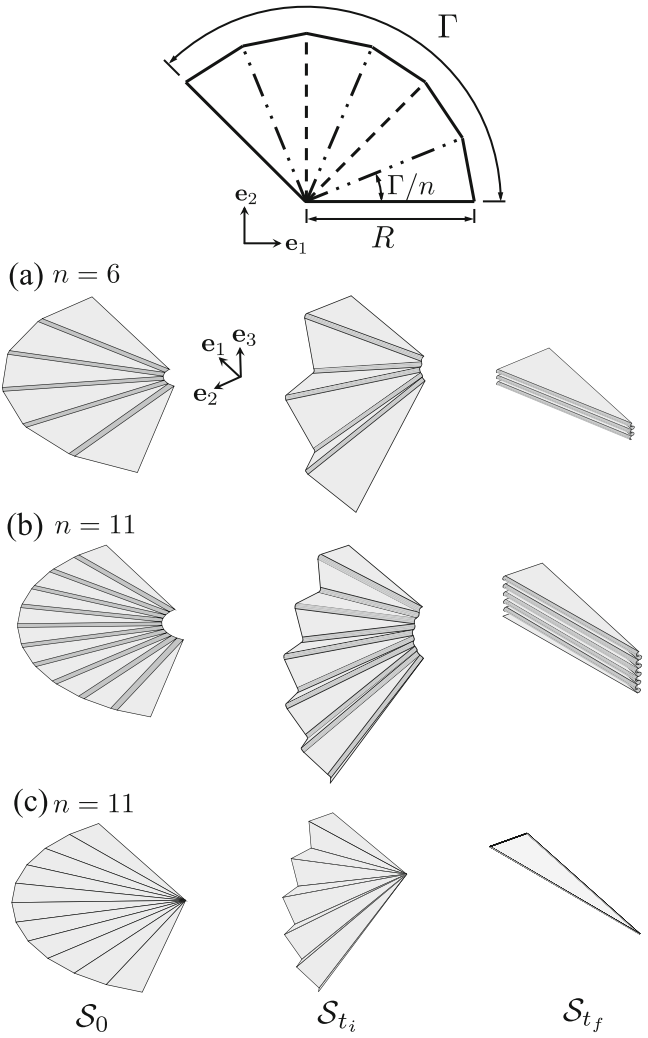


Fig. 5.23 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ for origami fans. The valley folds shown in the schematic reach a fold angle of π while the mountain folds reach a fold angle of $-\pi$. Values of $R = 0.5\,\text{m}$ and $\Gamma = 3\pi/4$ are used in all the examples. (**a**) and (**b**) Fans with smooth folds. (**c**) Fan with creased folds

the *mid-surface* of an origami structure of finite thickness in the present model). Nonetheless, it is noted that the model for origami with smooth folds presented here *also* accounts for folds having non-negligible thickness or maximum curvature constraints based on material or structural limitations.

In Fig. 5.25, we show examples of origami "umbrellas." The radius of the umbrellas is denoted R and the number of sections is given as $2n$. Origami umbrellas

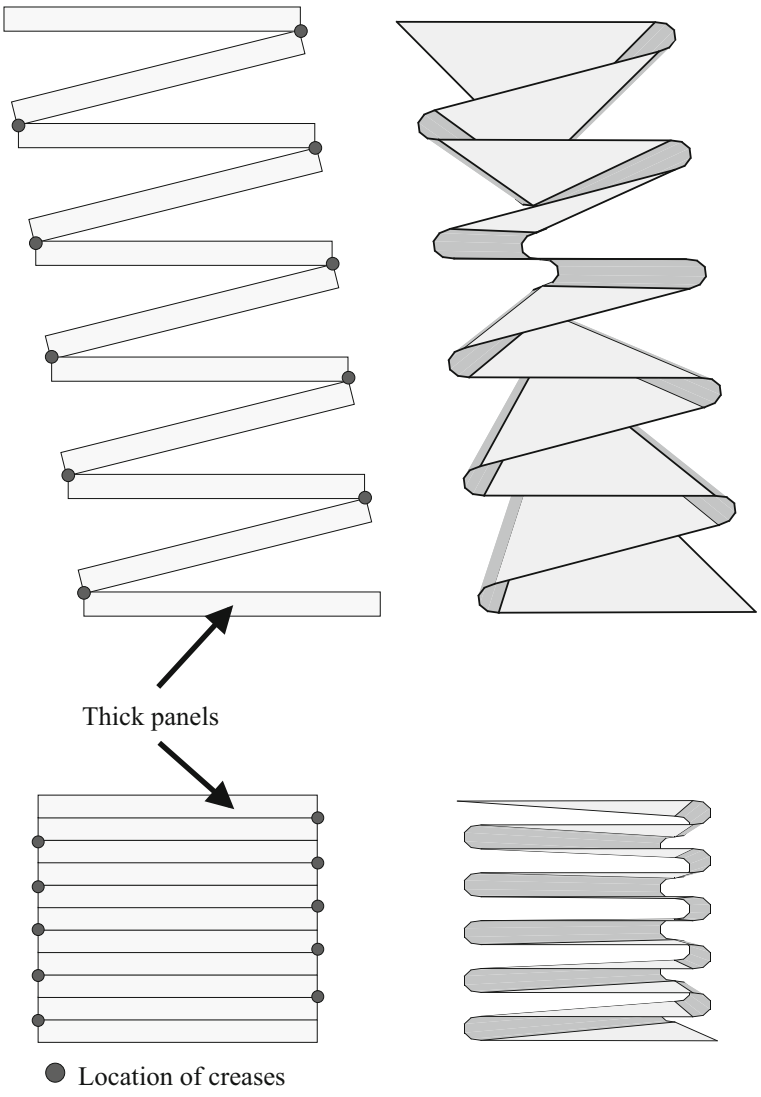


Fig. 5.24 Similarities between origami with smooth folds presented in this chapter and origami of thick panels [59]

with smooth folds are shown in Fig. 5.25a and b and an origami umbrella with creased folds is shown in Fig. 5.25c.

Figures 5.26 and 5.27 show examples of origami sheets where all the faces in the sheets are parallel in the final configuration $\mathcal{S}_{t_f}$ (i.e., *flat foldable* patterns). Specifically, Fig. 5.26 shows a sheet with a single interior fold intersection and four

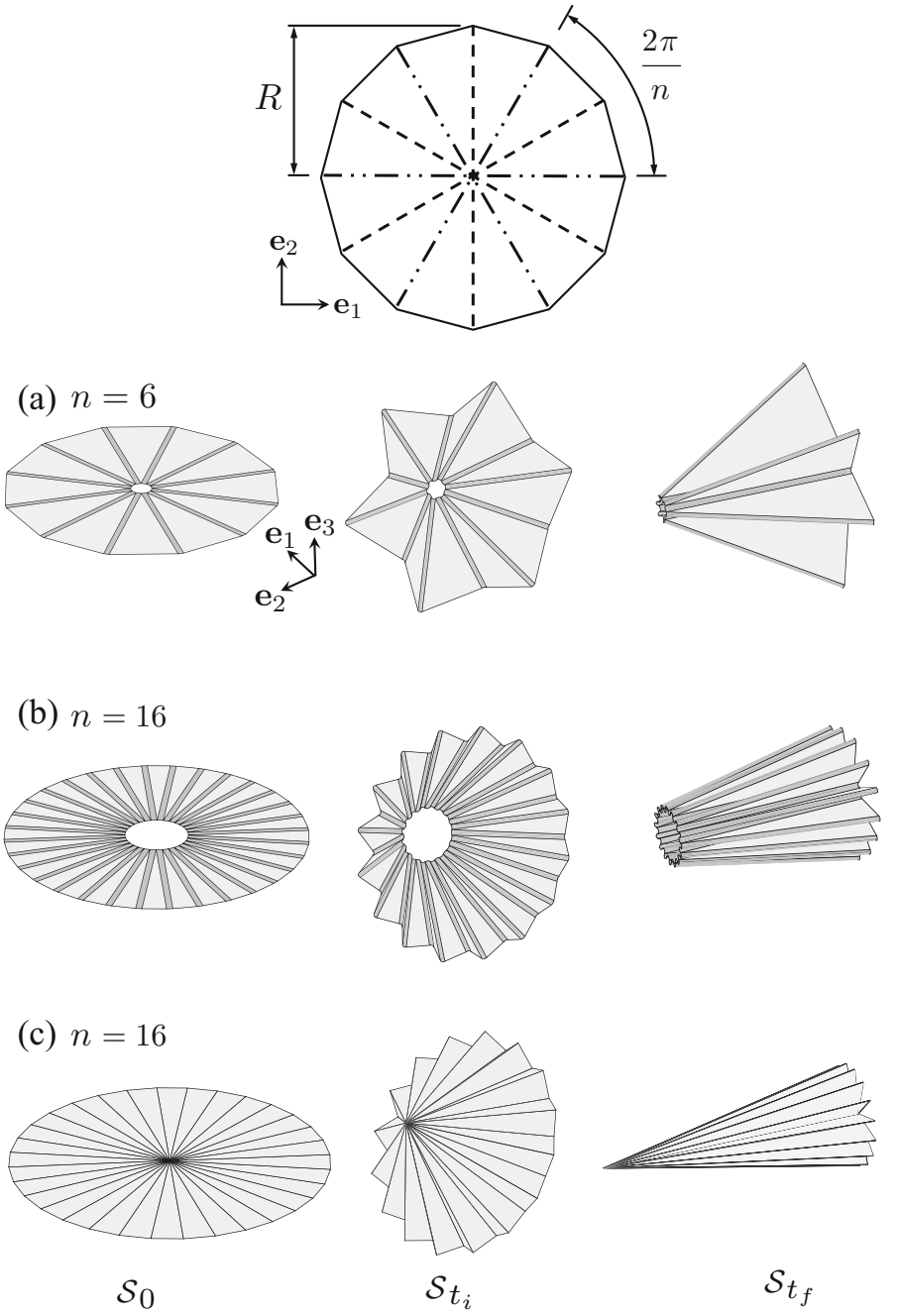


Fig. 5.25 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ ($0 < t_i < t_f$), and final configuration $\mathcal{S}_{t_f}$ for origami "umbrellas." A value of $R = 0.5$ m is used in all the examples. (**a**) and (**b**) Umbrellas with smooth folds. (**c**) Umbrella with creased folds

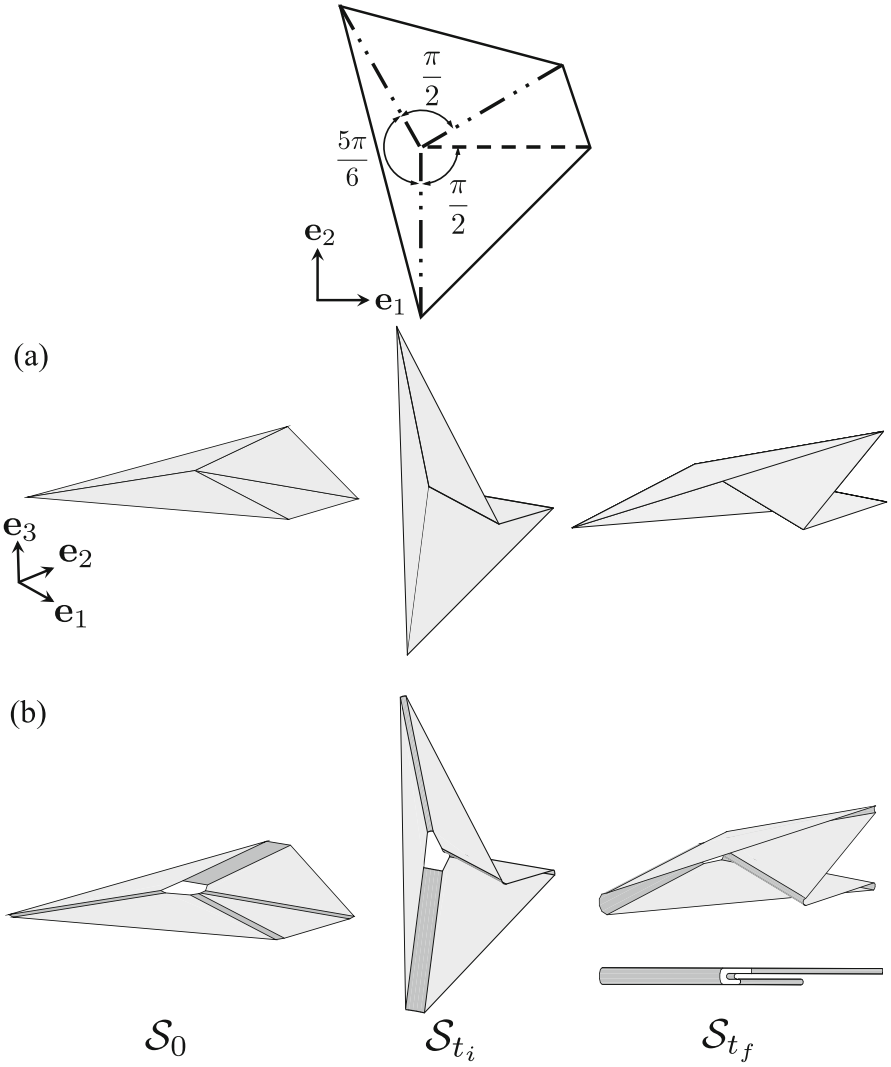


Fig. 5.26 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ of an origami sheet with four folds adjacent to a single interior fold intersection: (**a**) Sheet with creased folds; (**b**) Sheet with smooth folds

folds that have different fold widths while Fig. 5.27 shows a diamond fold pattern where all the fold widths in the sheet are equal.

A sheet having eight smooth folds meeting at one interior fold intersection is shown in Fig. 5.28a. The folds are enumerated in counterclockwise order. Two different sets of guess increments for the fold angles are considered. The

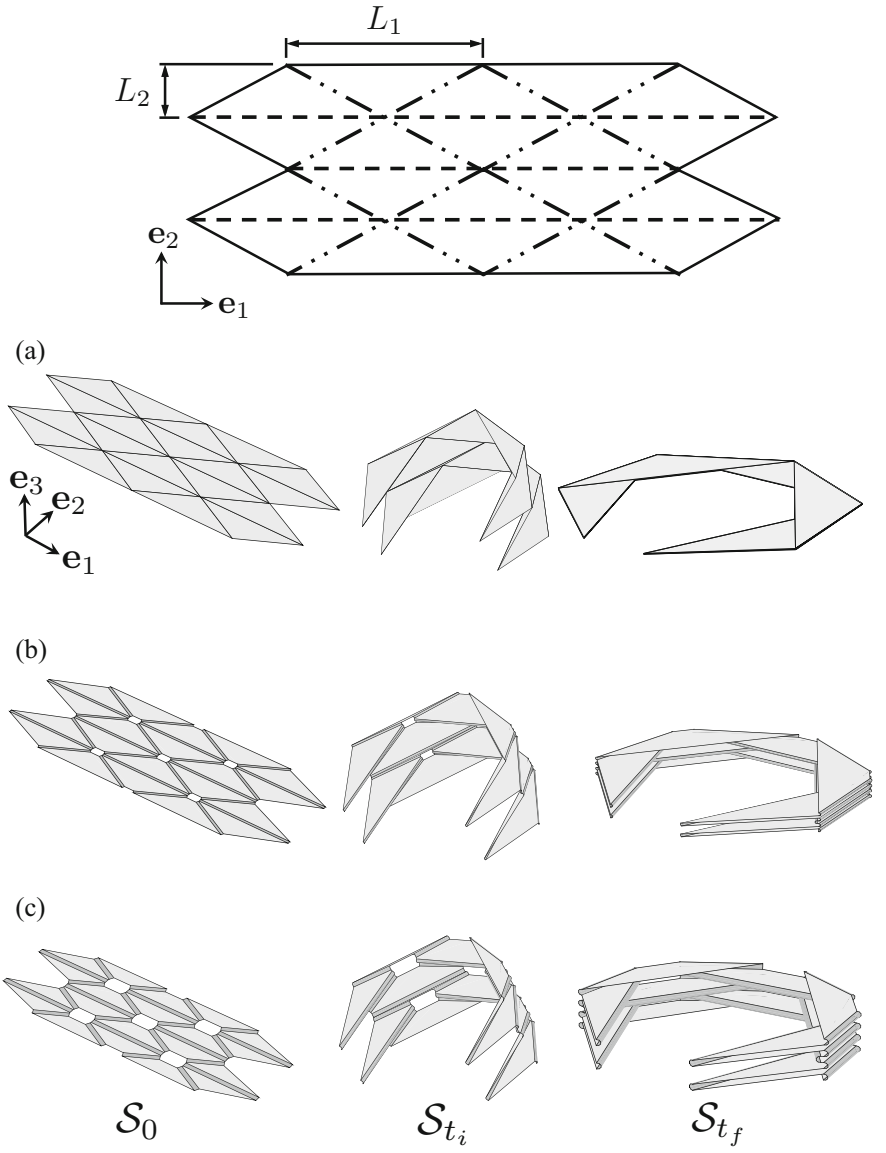


Fig. 5.27 Schematics showing the fold pattern, reference configuration $\mathcal{S}_0$, an intermediate current configuration $\mathcal{S}_{t_i}$ $(0 < t_i < t_f)$, and final configuration $\mathcal{S}_{t_f}$ for sheets with the diamond fold pattern. Values of $L_1 = 1.9$ m and $L_2 = 0.5$ m are used in all the examples. (**a**) Sheet with creased folds (i.e., zero fold width). (**b**) Sheet with smooth folds having a width of 0.045 m. (**c**) Sheet with smooth folds having a width of 0.09 m

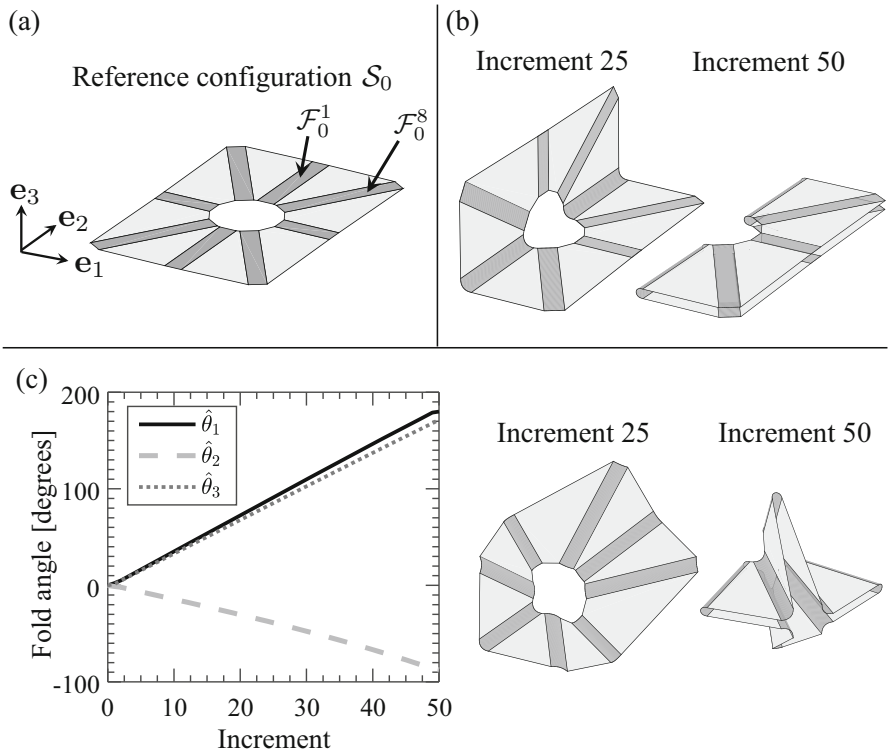


Fig. 5.28 (**a**) Reference configuration of a sheet having a single interior fold intersection. (**b**) Configurations of the sheet obtained via the guess increments for the fold angles provided in (5.109). (**c**) Fold angles vs. increment number and configurations obtained through the guess increments for the fold angles provided in (5.110)

folded configurations shown in Fig. 5.28b are obtained through the following guess increments for the fold angles:

$$\Delta\hat{\boldsymbol{\theta}}^l = \frac{\pi}{50}\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^\top \qquad l = 1, \ldots, 50. \tag{5.109}$$

The previous guess increments for the fold angles represent a simple example where the fold angle correction procedure (refer to Table 5.6) converged prior to performing an initial corrective iteration (i.e., $\|\mathfrak{R}(\hat{\boldsymbol{\theta}}^{l(0)})\|/(6N_{\mathcal{I}} + 2N_{\mathcal{F}}) < \texttt{tol1}$ for $l = 1, \ldots, 50$). Alternatively, an example of a more complex folding motion resulting from guess increments for the fold angles that required iterative corrections is shown in Fig. 5.28c. For this example, the guess increments for the fold angles are as follows:

$$\Delta\hat{\boldsymbol{\theta}}^l = \frac{\pi}{50}\begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}^\top \qquad l = 1, \ldots, 50. \tag{5.110}$$

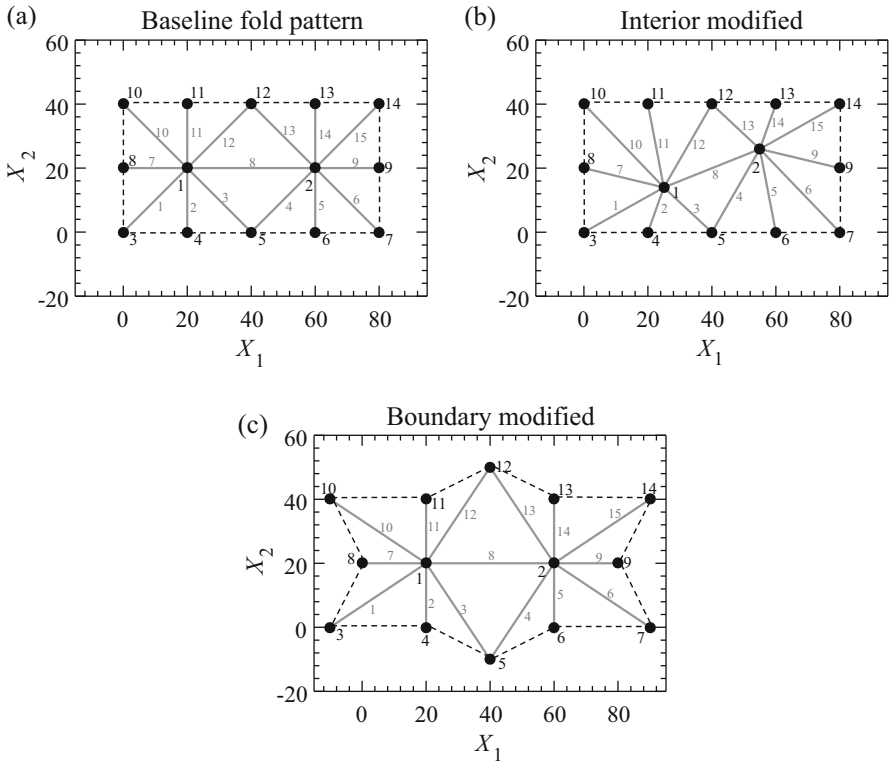


Fig. 5.29 Graphs showing the vertex coordinates and fold centerlines for the sheets shown in Fig. 5.30

As shown in the fold angle vs. increment plot in Fig. 5.28c, the fold angles obtained from the simulation procedure differ from the simple addition of the guess fold angle increments. As observable in the configurations shown in Fig. 5.28c, all the surface subdomains comprising the sheet remain joined throughout the folding motion.

More complex examples of origami sheets having two interior fold intersections are shown in Fig. 5.30. The graphs of Fig. 5.29 show the vertices and fold centerlines associated with such sheets. Since the two interior fold intersections for these sheets share a common adjacent fold, their associated constraints (5.66) and (5.67) are coupled. For all the three sheets, the guess increments for the fold angles are as follows (see Fig. 5.29 for the numbering of the folds):

$$\Delta\hat{\boldsymbol{\theta}}^l = \frac{\pi}{120}\begin{bmatrix} 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 \end{bmatrix}^\top$$
$$l = 1, \ldots, 100. \tag{5.111}$$

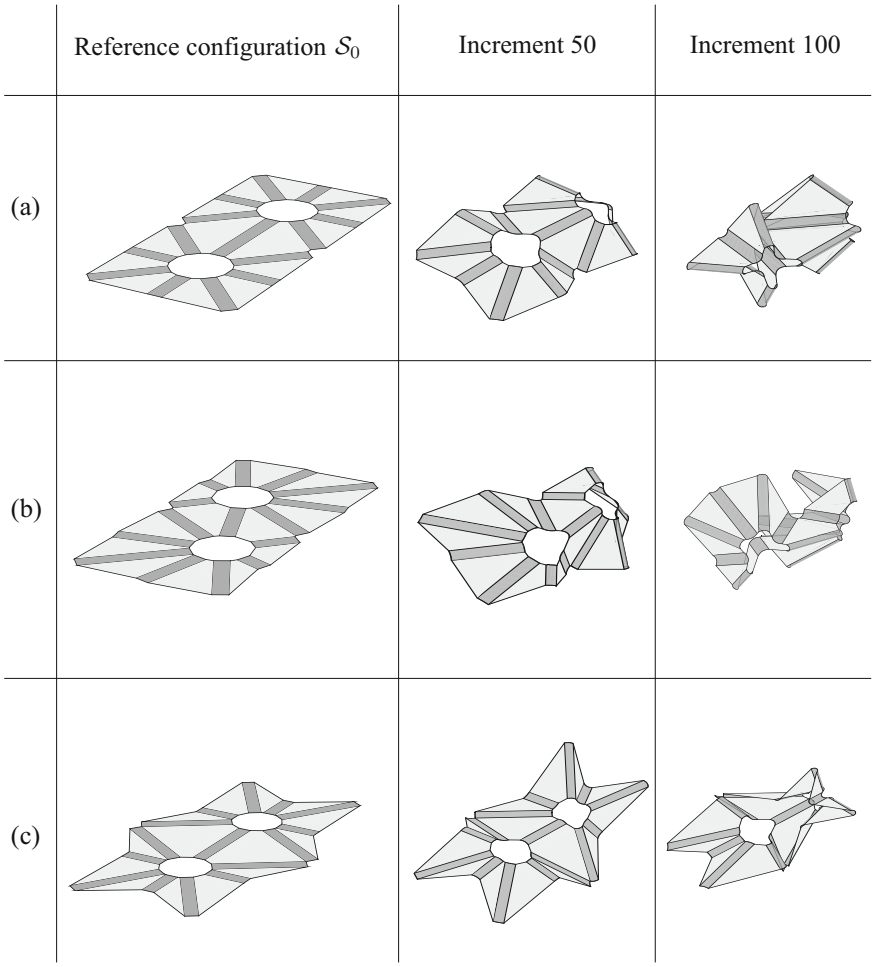


Fig. 5.30 Configurations for origami sheets with vertex coordinates and fold centerlines defined in Fig. 5.29: (**a**) Sheet with the baseline fold pattern; (**b**) Sheet with a fold pattern generated by modifying the interior vertex coordinates of the baseline fold pattern; (**c**) Sheet with a fold pattern generated by modifying the boundary vertex coordinates of the baseline fold pattern

The sheet having the baseline fold pattern shown in Fig. 5.30a exhibits a symmetric behavior, which is expected from the symmetry of the fold pattern and the guess increments for the fold angles. Sheets having fold patterns obtained by modifying the coordinates of the interior vertices and the coordinates of the boundary vertices of the baseline fold pattern are, respectively, shown in Fig. 5.30b and c. It is observed that both sheets undergo dissimilar fold angle histories compared to the sheet having the baseline fold pattern as observed from both the folded configurations in Fig. 5.30 and the fold angle vs. increment number plots in Fig. 5.31. These examples show the versatility of the present model in allowing for simulation of sheets having arbitrary fold patterns and boundary shapes.

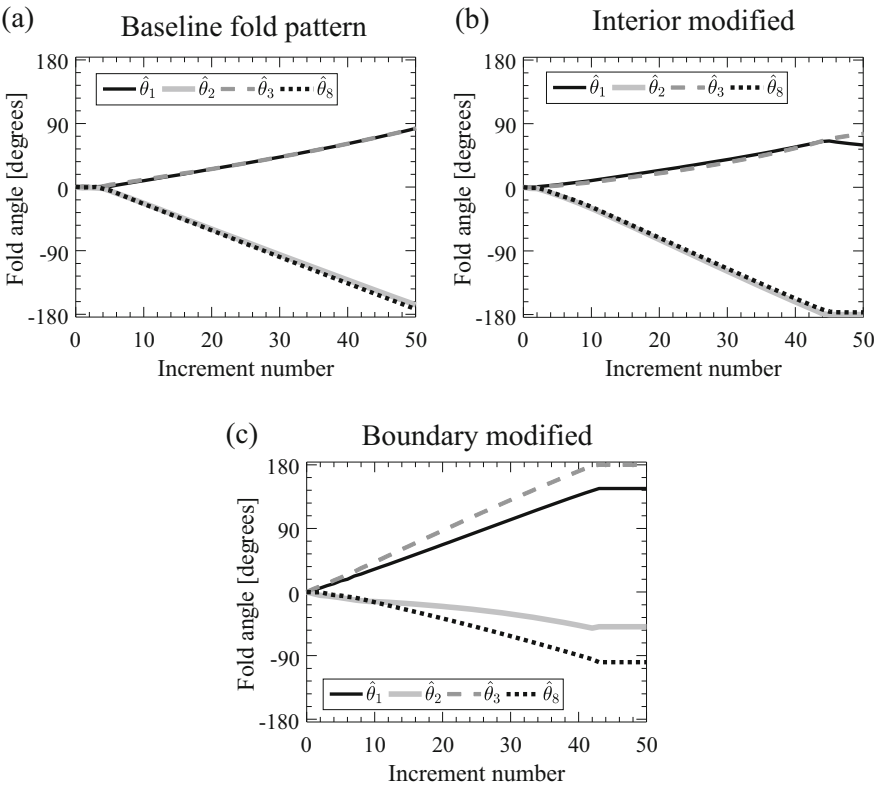


Fig. 5.31 Evolution of fold angles with increment number for the sheets shown in Fig. 5.30

The present model is also applicable for the simulation of sheets having arbitrary topology. To illustrate this, sheets having four and five interior fold intersections are shown in Figs. 5.32 and 5.33, respectively. Various configurations are shown for these examples. As observed from these schematics, the present model captures well the behavior of the folded sheets during their full range of motion (fold angles vary from 0 to $\pm\pi$ for various folds in these sheets).

Chapter Summary

A model for the kinematic response of origami structures with smooth folds having non-zero sheet surface area and arbitrary order of geometric continuity was presented in this chapter. The geometrical description of smooth folds was presented in Sect. 5.3. The fold pattern description (Sect. 5.4), the constraints on the sheet deformation for origami with smooth folds analogous to those for

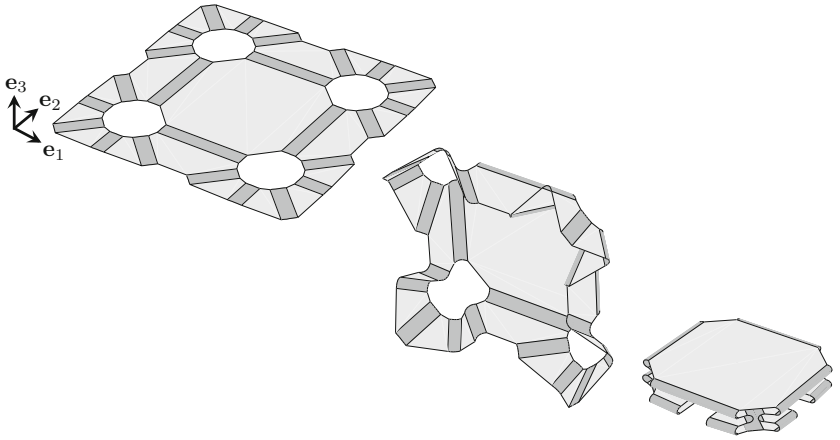


Fig. 5.32 Demonstration of constrained folding motion associated with an origami sheet having four interior fold intersections

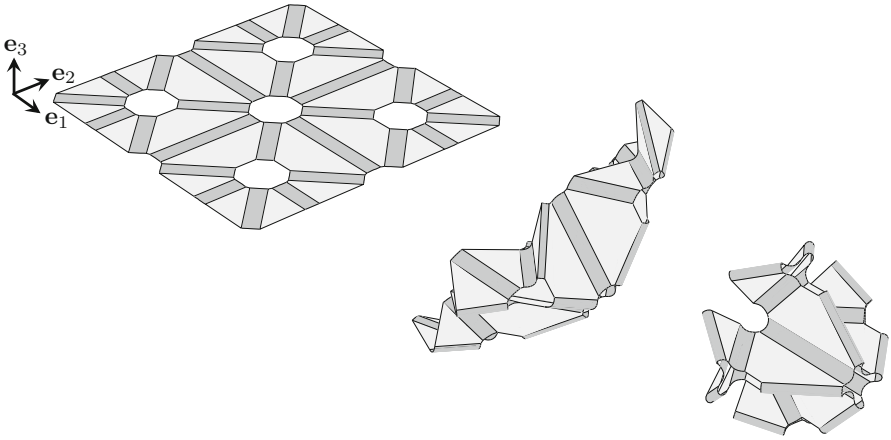


Fig. 5.33 Demonstration of constrained folding motion associated with an origami sheet having five interior fold intersections

origami with creased folds (Sect. 5.5), and the mapping from reference to current configurations (Sect. 5.6) were also presented. The numerical implementation of the model allowing for simulation of the motion of sheets with arbitrary fold patterns was described in Sect. 5.7 and representative implementation examples were provided in Sect. 5.8.

The proposed model successfully allows for the mathematical representation of origami with folds that have non-zero sheet surface area and arbitrary order of geometric continuity (in terms of fold shape geometry, constraints on the fold kinematic variables, and mapping from reference to current configurations). The

conventional origami with creased folds of G^0 continuity addressed in Chap. 2 represents a special case of this more general model and can be captured as well. Furthermore, the arbitrary order of geometric continuity in the sheet considered in this chapter allows for the physical modeling of origami structures having significant thickness using plate or shell representations, which is the focus of Chap. 8.

Problems

5.1 The reference configuration of the smooth folds is simplified in this chapter to a rectangular shape. However, the model presented here can be extended for the consideration of folds having trapezoidal reference configurations. Propose a formulation for the parametric surface $\boldsymbol{\mathcal{F}}^i(\zeta_1, \zeta_2)$ analogous to that of (5.1) for smooth folds having a trapezoidal reference configurations.

5.2 Based on Fig. 5.6, state the conditions on the second derivative of $\hat{\mathbf{c}}^i(\zeta_1)$ required for G^2 continuity of a general fold for which $\hat{a}_i$ is not necessarily equal to $\frac{1}{2}$.

5.3 Derive expressions for the unit tangent vector $\mathbf{t}^i(\zeta_1)$, curvature $\hat{\kappa}_i(\zeta_1)$, and signed curvature $\kappa_i(\zeta_1)$ for the parameterization of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ provided in (5.21).

5.4 Repeat Problem 5.3 considering the parameterization of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ provided in (5.23).

5.5 Show that the parameterization of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ in (5.21) satisfies the conditions for G^1 continuity presented in Sect. 5.3.1.

5.6 Show that the parameterization of the curve $\hat{\mathbf{c}}^i(\zeta_1)$ in (5.23) satisfies the conditions for G^2 continuity presented in Sect. 5.3.1.

5.7 Propose a constraint on the parameters $\hat{r}_{i1}$ and $\hat{r}_{i2}$ (see (5.28)) that ensures that the length of the smooth folds along the direction of the fold vector is positive.

5.8 Determine the position vectors $\hat{\mathbf{p}}^{i1}$, $\hat{\mathbf{p}}^{i3}$, and $\hat{\mathbf{p}}^{i4}$ of the corners of the fold shown in Fig. 5.11.

5.9 In Sect. 5.5.2, the constraints on the kinematic variables for the folds adjacent to an interior fold intersection were derived using a path $\boldsymbol{\gamma}^j(\eta)$ oriented counterclockwise. Derive such constraints using the process presented Sect. 5.5.2 but using a path $\boldsymbol{\gamma}^j(\eta)$ oriented clockwise (i.e., crossing the folds with associated fold vectors in the following order: $\mathbf{m}^{jn_j}, \ldots, \mathbf{m}^{j2}, \mathbf{m}^{j1}$). Are the resulting constraints equivalent to those provided in (5.66) and (5.67)?

5.10 Derive the mapping (5.44) considering general smooth folds for which $\hat{a}_i$ is not necessarily equal to $\frac{1}{2}$.

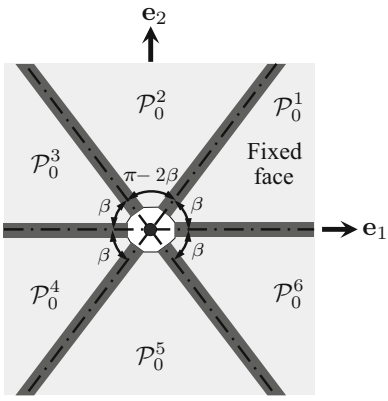


Fig. 5.34 Schematic for Problem 5.12

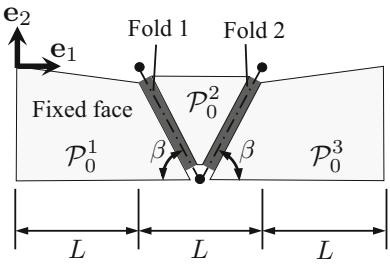


Fig. 5.35 Schematic for Problem 5.13

5.11 Using the results from Problem 5.10, derive the kinematic constraints (5.66) and (5.67) considering general smooth folds for which $\hat{a}_i$ is not necessarily equal to $\frac{1}{2}$.

5.12 Considering the fold pattern presented in Fig. 5.34, determine the mapping between reference and current configurations in the form of (5.49) for each face. Denote the width of each fold as w^* and the distance between the interior vertex and all the fold boundaries as r^*.

5.13 Consider the fold pattern presented in Fig. 5.35. (a) Determine the folding map in the form of (5.94) for each face. Denote the width of each fold as w^* and the distance between the fold centerline end points and the fold boundaries as r^*. (b) Determine the position vector $\mathbf{x}$ in a current configuration with $\hat{\theta}_1 = \pi/2$, $\hat{\theta}_2 = -\pi/2$, and $\hat{w}_1 = \hat{w}_2 = w^*/2$ of the point with reference position vector $\mathbf{X} = [3L \quad 0 \quad 0]^\top$.

5.14 Repeat Problem 2.8 considering a sheet having smooth folds.

5.15 Repeat Problem 2.9 considering a sheet having smooth folds.

5.16 Repeat Problem 2.10 considering a sheet having smooth folds.

5.17 Implement the algorithm for kinematic simulation of origami sheets with smooth folds presented in this chapter. Then, create the fold patterns and simulate

the folding motion of: (a) An origami self-similar wave having the parameterization shown in Fig. 5.22 and parameter $n = 12$; (b) Origami fans having the parameterization shown in Fig. 5.23 and parameters $\Gamma = \pi$, $R = 0.5$ m, and $n = 5, 10, 20$; (c) Origami umbrellas having the parameterization shown in Fig. 5.25 and parameters $R = 0.5$ m, $n = 8$, and three distinct values for the all the fold widths in the sheet.

References

1. T. Tachi, Simulation of rigid origami, in *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education*, pp. 175–187 (2009)
2. T.A. Evans, R.J. Lang, S.P. Magleby, L.L. Howell, Rigidly foldable origami gadgets and tessellations. R. Soc. Open Sci. **2**(9) (2015)
3. Z. Abel, J. Cantarella, E.D. Demaine, D. Eppstein, T.C. Hull, J.S. Ku, R.J. Lang, T. Tachi, Rigid origami vertices: conditions and forcing sets. J. Comput. Geom. **7**(1), 171–194 (2016)
4. T. Tachi, Geometric considerations for the design of rigid origami structures, in *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*, vol. 12, pp. 458–460 (2010)
5. T. Tachi, Designing freeform origami tessellations by generalizing Resch's patterns, J. Mech. Des. **135**(11), 111006 (2013)
6. T. Tachi, Freeform origami tessellations by generalizing Resch's patterns, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2013–12326, pp. V06BT07A025 (American Society of Mechanical Engineers, New York, 2013)
7. J.M. Gattas, W. Wu, Z. You, Miura-base rigid origami: parameterizations of first-level derivative and piecewise geometries. J. Mech. Des. **135**(11), 111011 (2013)
8. K. Wang, Y. Chen, Folding a patterned cylinder by rigid origami, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education*, pp. 265–276 (2011)
9. X.-Y. Li, T. Ju, Y. Gu, S.-M. Hu, A geometric study of V-style pop-ups: theories and algorithms, in *ACM SIGGRAPH 2011 Papers*, SIGGRAPH '11, pp. 98:1–98:10 (ACM, New York, 2011). ISBN: 978-1-4503-0943-1
10. S.N. Le, S.-J. Leow, T.-V. Le-Nguyen, C. Ruiz, K.-L. Low, Surface and contour-preserving origamic architecture paper pop-ups. IEEE Trans. Vis. Comput. Graph. **20**(2), 276–288 (2014)
11. P.M. Reis, F.L. Jiménez, J. Marthelot, Transforming architectures inspired by origami. Proc. Natl. Acad. Sci. **112**(40), 12234–12235 (2015)
12. R.J. Lang, *Origami In Action: Paper Toys That Fly, Flag, Gobble and Inflate!* (St. Martin's Griffin, New York, 1997)
13. Y. Qin, J. Dai, Four motion branches of an origami based eight bar spatial mechanism, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, Portland, No. DETC2013–12584, pp. V06BT07A030 (2013)
14. L.A. Bowen, C.L. Grames, S.P. Magleby, L.L. Howell, R.J. Lang, A classification of action origami as systems of spherical mechanisms. J. Mech. Des. **135**(11), 111008 (2013)
15. L.A. Bowen, C.L. Grames, S.P. Magleby, R.J. Lang, L.L. Howell, An approach for understanding action origami as kinematic mechanisms, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, Portland, No. DETC2013–13407, pp. V06BT07A044 (American Society of Mechanical Engineers, New York, 2013)
16. L.A. Bowen, W.L. Baxter, S.P. Magleby, L.L. Howell, A position analysis of coupled spherical mechanisms found in action origami. Mech. Mach. Theory **77**, 13–24 (2014)

17. E.R. Leal, J.S. Dai, From origami to a new class of centralized 3-DOF parallel mechanisms, in *Proceedings of the ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, pp. 1183–1193 (American Society of Mechanical Engineers, New York, 2007)
18. G. Wei, J.S. Dai, Origami-inspired integrated planar-spherical overconstrained mechanisms. J. Mech. Des. **136**(5), 051003 (2014)
19. I.L. Delimont, S.P. Magleby, L.L. Howell, Evaluating compliant hinge geometries for origami-inspired mechanisms. J. Mech. Robot. **7**(1), 011009 (2015)
20. B.G. Winder, S.P. Magleby, L.L. Howell, Kinematic representations of pop-up paper mechanisms. J. Mech. Robot. **1**(2), 021009 (2009)
21. G. Wei, J.S. Dai, Geometry and kinematic analysis of an origami-evolved mechanism based on artmimetics, in *Proceedings of the 2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots ReMAR*, pp. 450–455 (IEEE, Piscataway, 2009)
22. B.H. Hanna, S.P. Magleby, R.J. Lang, L.L. Howell, Force–deflection modeling for generalized origami waterbomb-base mechanisms. J. Appl. Mech. **82**(8), 081001 (2015)
23. A. Yellowhorse, L.L. Howell, Creating rigid foldability to enable mobility of origami-inspired mechanisms. J. Mech. Robot. **8**(1), 011011 (2016)
24. W. Gao, K. Ramani, R.J. Cipra, Reconfigurable foldable spatial mechanisms and robotic forms inspired by kinetogami, in *Proceedings of the ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/-CIE*, No. DETC2012–71403, pp. 1161–1168 (American Society of Mechanical Engineers, New York, 2012)
25. T. Evans, Deployable and foldable arrays of spatial mechanisms. Master's thesis, Brigham Young University-Provo, 2015
26. Y.L. Kergosien, H. Gotoda, T.L. Kunii, Bending and creasing virtual paper. IEEE Comput. Graph. Appl. **14**(1), 40–48 (1994)
27. J. Solomon, E. Vouga, M. Wardetzky, E. Grinspun, Flexible developable surfaces. Comput. Graph. Forum **31**(5), 1567–1576 (2012)
28. H.-D. Hwang, S.-H. Yoon, Constructing developable surfaces by wrapping cones and cylinders. Comput. Aided Des. **58**, 230–235 (2015)
29. C. Schreck, D. Rohmer, S. Hahmann, M.-P. Cani, S. Jin, C.C.L. Wang, J.-F. Bloch, Nonsmooth developable geometry for interactively animating paper crumpling. ACM Trans. Graph. (TOG) **35**(1), 10 (2015)
30. L. Zhu, T. Igarashi, J. Mitani, Soft folding. Comput. Graph. Forum **32**(7), 167–176 (2013)
31. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr., D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
32. K. Fuchi, T.H. Ware, P.R. Buskohl, G.W. Reich, R.A. Vaia, T.J. White, J.J. Joo, Topology optimization for the design of folding liquid crystal elastomer actuators. Soft Matter **11**(37), 7288–7295 (2015)
33. K. Fuchi, P.R. Buskohl, T. Ware, R.A. Vaia, T.J. White, G.W. Reich, J.J. Joo, Inverse design of LCN films for origami applications using topology optimization, in *Proceedings of the ASME 2014 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, No. SMASIS2014–7497, pp. V001T01A011 (American Society of Mechanical Engineers, New York, 2014)
34. Y. Liu, J.K. Boyles, J. Genzer, M.D. Dickey, Self-folding of polymer sheets using local light absorption. Soft Matter **8**(6), 1764–1769 (2012)
35. E.A. Peraza Hernandez, S. Hu, H.W. Kung, D. Hartl, E. Akleman, Towards building smart self-folding structures. Comput. Graph. **37**(6), 730–742 (2013)
36. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr., Design and numerical analysis of an SMA mesh-based self-folding sheet. Smart Mater. Struct. **22**(9), 094008 (2013)
37. E. Peraza Hernandez, D. Hartl, E. Galvan, R. Malak, Design and optimization of a shape memory alloy-based self-folding sheet. J. Mech. Des. **135**(11), 111007 (2013)

38. E. Peraza Hernandez, D. Hartl, R. Malak, D. Lagoudas, Analysis and optimization of a shape memory alloy-based self-folding sheet considering material uncertainties, in *Proceedings of the ASME 2015 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, No. SMASIS2015–9001, pp. V001T01A013 (American Society of Mechanical Engineers, New York, 2015)
39. S. Ahmed, Z. Ounaies, M. Frecker, Investigating the performance and properties of dielectric elastomer actuators as a potential means to actuate origami structures. Smart Mater. Struct. **23**(9), 094003 (2014)
40. K. McGough, S. Ahmed, M. Frecker, Z. Ounaies, Finite element analysis and validation of dielectric elastomer actuators used for active origami. Smart Mater. Struct. **23**(9), 094002 (2014)
41. L. Ionov, Nature-inspired stimuli-responsive self-folding materials, in *Intelligent Stimuli-Responsive Materials: From Well-Defined Nanostructures to Applications*, pp. 1–16 (2013)
42. C. Danielson, A. Mehrnezhad, A. YekrangSafakar, K. Park, Fabrication and characterization of self-folding thermoplastic sheets using unbalanced thermal shrinkage. Soft Matter **13**, 4224–4230 (2017)
43. Y. Liu, B. Shaw, M.D. Dickey, J. Genzer, Sequential self-folding of polymer sheets. Sci. Adv. **3**(3), e1602417 (2017)
44. A.N. Pressley, *Elementary Differential Geometry* (Springer, London, 2010)
45. N.J. Pagano, Exact solutions for composite laminates in cylindrical bending, J. Compos. Mater. **3**(3), 398–411 (1969)
46. P. Heyliger, S. Brooks, Exact solutions for laminated piezoelectric plates in cylindrical bending. J. Appl. Mech. **63**(4), 903–910 (1996)
47. P.V. Nimbolkar, I.M. Jain, Cylindrical bending of elastic plates. Procedia Mater. Sci. **10**, 793–802 (2015)
48. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Kinematics of origami structures with smooth folds. J. Mech. Robot. **8**(6), 061019 (2016)
49. B.A. Barsky, T.D. DeRose, *Geometric Continuity of Parametric Curves* (Computer Science Division, University of California, 1984)
50. B.A. Barsky, R.H. Bartels, J.C. Beatty, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling* (M. Kaufmann Publishers, Los Altos, 1987)
51. W. Cheney, D. Kincaid, *Numerical Analysis. Mathematics of Scientific Computing* (Brooks & Cole Publishing Company, Pacific Grove, 1996)
52. T. Tachi, Freeform variations of origami. J. Geom. Graph. **14**(2), 203–215 (2010)
53. S.-M. Belcastro, T.C. Hull, A mathematical model for non-flat origami, in *Origami 3: Third International Meeting of Origami Mathematics, Science, and Education*, pp. 39–51 (2002)
54. S.-M. Belcastro, T.C. Hull, Modelling the folding of paper into three dimensions using affine transformations. Linear Algebra Appl. **348**(1–3), 273–282 (2002)
55. MathWorks, MATLAB's `fmincon`. http://www.mathworks.com/help/optim/ug/fmincon.html
56. MathWorks, Filled 3-D polygons in MATLAB (`fill3`). http://www.mathworks.com/help/matlab/ref/fill3.html
57. MathWorks, Surface plot in MATLAB (`surf`). http://www.mathworks.com/help/matlab/ref/surf.html
58. T. Hull, *Project Origami: Activities for Exploring Mathematics* (CRC Press, Boca Raton, 2012)
59. Y. Chen, R. Peng, Z. You, Origami of thick panels. Science **349**(6246), 396–400 (2015)

Chapter 6
Unfolding Polyhedra Method for the Design of Origami Structures with Smooth Folds

Abstract In this chapter, we address the method of *unfolding polyhedra* for origami structures with smooth folds. We develop this method based on the theory of unfolding polyhedra for origami with creased folds studied in Chap. 3. Accordingly, the goal shape is represented as a three-dimensional goal mesh. The objective is to determine the geometry of a planar sheet with smooth folds that can be folded towards a configuration that approximates the goal mesh. We also examine the computational implementation aspects of unfolding polyhedra for origami with smooth folds and provide representative examples.

6.1 Introduction

Origami structures posses several characteristics that are advantageous in deployable systems and morphing structures. These characteristics include storage/deployment capabilities [1–3], reconfigurability, and reduction in manufacturing complexity [4–18]. As stated in Sect. 3.1, the key challenge in the development of origami applications is *origami design*, which is the process of creating an origami structure with targeted properties such as a desired shape. Current methods for origami design generally assume that the folds in the origami structures are creases [19–21], such as in the unfolding polyhedra method considered in Chap. 3 and throughout the literature [19, 22–34].

The assumption of creased folds is not suitable for the design of origami structures having non-negligible fold thickness or that are comprised of materials incapable of the strain magnitudes required to generate the high curvatures compatible with a creased idealization. For these structures, the folded regions are not accurately represented as creases but rather as bent sheet regions having higher-order geometric continuity, which we call *smooth folds*. In view of this, we develop the *unfolding polyhedra method for origami with smooth folds* in this chapter. This

Electronic Supplementary Material The online version of this article (https://doi.org/10.1007/978-3-319-91866-2_6) contains supplementary material, which is available to authorized users.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_6

method is developed based on the theory of unfolding polyhedra for origami with creased folds studied in Chap. 3.

The unfolding polyhedra method for origami with smooth folds presented in this chapter is particularly useful in the design of *active origami structures*, where the achievable curvature at the active material-based folds is limited by the maximum strain magnitude provided by such materials [11, 35–40]. For example, active origami structures with smooth folds designed via the unfolding polyhedra method are shown in Figs. 6.1 [41] and 6.2 [42]. This chapter is organized as follows: Sect. 6.2 presents the problem description and solution approach of the unfolding polyhedra method considering smooth folds, and examples of the implemented method considering several goal shapes are provided in Sect. 6.3.

6.2 Unfolding Polyhedra Method Considering Smooth Folds

Here we describe the various aspects of the unfolding polyhedra method for the design of origami structures with smooth folds. Section 6.2.1 provides the design problem definition and the solution approach, and Sect. 6.2.2 describes the face trimming step.

6.2.1 Problem Definition

The unfolding polyhedra method for origami with smooth folds aims to solve the following problem in origami design:

- *Given*: A three-dimensional goal shape represented as a polygonal mesh (termed as the *goal mesh* $\mathcal{M}$),
- *Find*: The geometry of a planar sheet with smooth folds that can be folded to approximate $\mathcal{M}$, and a history of folding motion from the planar configuration of the sheet ($\mathcal{S}_0$) to the goal configuration ($\mathcal{S}_\star$) that matches $\mathcal{M}$

As explained in Chap. 3, the planar sheet obtained using the unfolding polyhedra method corresponds to an *unfolding* of the goal mesh $\mathcal{M}$, which is the flattening of $\mathcal{M}$ onto a plane and has boundary segments associated with *cuts* made on $\mathcal{M}$. As in conventional approaches for unfolding polyhedra [19, 24], we require the unfolding to be a single sheet that does not have any overlaps and the cuts are required to be made exclusively on edges of the goal mesh. An unfolding that satisfies these characteristics is called a *net* [19]. In the examples of active origami structures shown in Figs. 6.1 and 6.2, we note that the sheets do not have overlaps and were obtained by making cuts solely on edges of the goal shapes; therefore, they are nets.

The steps followed in the unfolding polyhedra method for origami with smooth folds to determine a net for a given goal mesh $\mathcal{M}$ are illustrated in Fig. 6.3 and listed as follows:

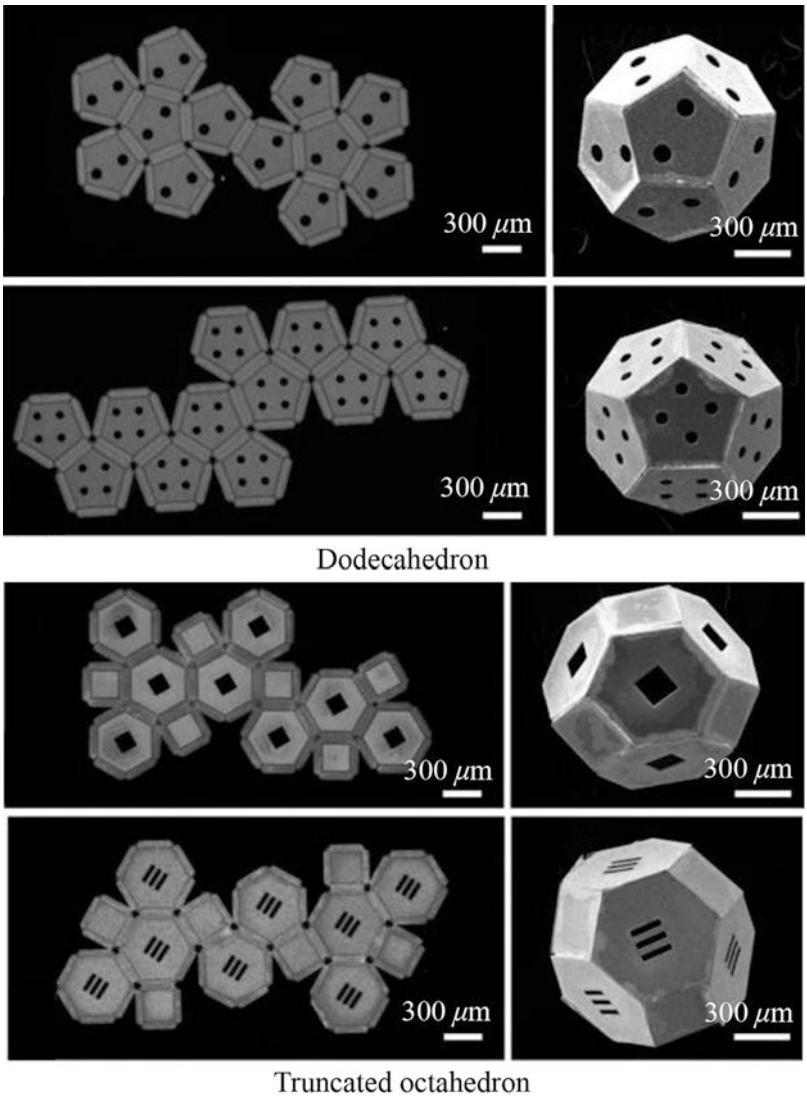


Fig. 6.1 Motivation for the current design method based on active origami structures. The images show active origami structures realized using the unfolding polyhedra method [41]. Self-folding occurs after the sheets are released from the silicon substrate on which they are fabricated and then heated above the melting point of the material at the folds, where face rotation is subsequently driven by the minimization of surface tension at the liquid fold hinges. Adapted from [41] with permission of the Proceedings of the National Academy of Sciences (PNAS)

1. Determination of a *spanning tree* on the goal mesh $\mathcal{M}$ (Fig. 6.3b). A spanning tree is a line graph on $\mathcal{M}$ that contains a reference point in each face of $\mathcal{M}$. The spanning tree cannot contain any node of $\mathcal{M}$ and cannot cross any edge of $\mathcal{M}$

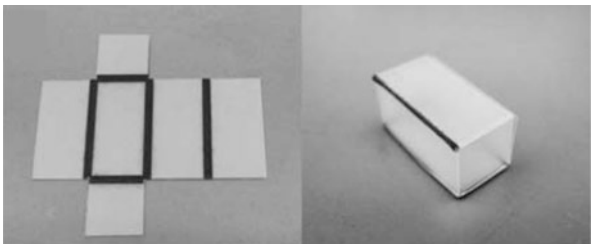


Fig. 6.2 Photographs of a square prism created via thermally activated folding of a shape memory polymer sheet. Note that the folds are not creases. Adapted from [42] with permission of The Royal Society of Chemistry

more than once. The importance of the spanning tree is not its particular shape, but is rather the set of edges of $\mathcal{M}$ that it crosses

2. Assignment of every edge that is *not* crossed by the spanning tree as a *boundary edge* (Fig. 6.3c). Since only the edges crossed by spanning tree are not assigned as boundary edges, the resulting mesh corresponds to a tree of faces that has the same topology of the spanning tree
3. The faces of $\mathcal{M}$ are adjusted (i.e., "trimmed") at the locations where the smooth folds will be introduced. These locations correspond to the edges of $\mathcal{M}$ that are *not* assigned as boundary edges in Step 2. This step is referred to as *face trimming*. The resulting *trimmed mesh* is denoted $\mathcal{M}_\sharp$ as shown in Fig. 6.3d
4. Mapping the trimmed faces obtained in Step 3 onto a plane. This step produces a *net* for the goal mesh, which corresponds to the reference configuration $\mathcal{S}_0$ of an origami sheet with smooth folds as shown in Fig. 6.3e. This sheet contains the faces of $\mathcal{M}_\sharp$ and the introduced smooth folds
5. Determination of a history of folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that matches $\mathcal{M}$ (Fig. 6.3f)

Here, we utilize the same data employed in Chap. 3 to define the goal mesh $\mathcal{M}$, which is described in detail in Sect. 3.2.2. Likewise, the procedure employed to determine a spanning tree for $\mathcal{M}$ in Chap. 3 is also utilized here. Such a procedure is presented in Sect. 3.2.3. The *face trimming* step that is introduced in the unfolding polyhedra method for origami with smooth folds is addressed in Sect. 6.2.2. The procedure to determine a history of folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ follows the one outlined in Sect. 3.2.5 for unfolding polyhedra with creased folds. We provide a set of MATLAB® scripts that perform the aforementioned steps in the Supplemental Material of this chapter. These MATLAB scripts are described in Appendix B.5.

The *number of folds* $N_\mathcal{F}$ in a sheet designed using the unfolding polyhedra method is equal to $N_\mathcal{M} - 1$ as determined in (3.23), where $N_\mathcal{M}$ is the *number of mesh faces* of $\mathcal{M}$. For the goal mesh $\mathcal{M}$ shown in Fig. 6.3, $N_\mathcal{M} = 6$ and hence $N_\mathcal{F} = N_\mathcal{M} - 1 = 5$. In this chapter, the *fold widths* $\hat{w}_i^0$, $i = 1, \ldots, N_\mathcal{F}$, of the introduced smooth folds are assumed *given*[1] (see Fig. 6.3f). In practice, the fold widths are determined such that the smooth folds having a specified thickness and

[1] Consult Sect. 5.3 for a detailed description of the geometry of smooth folds.

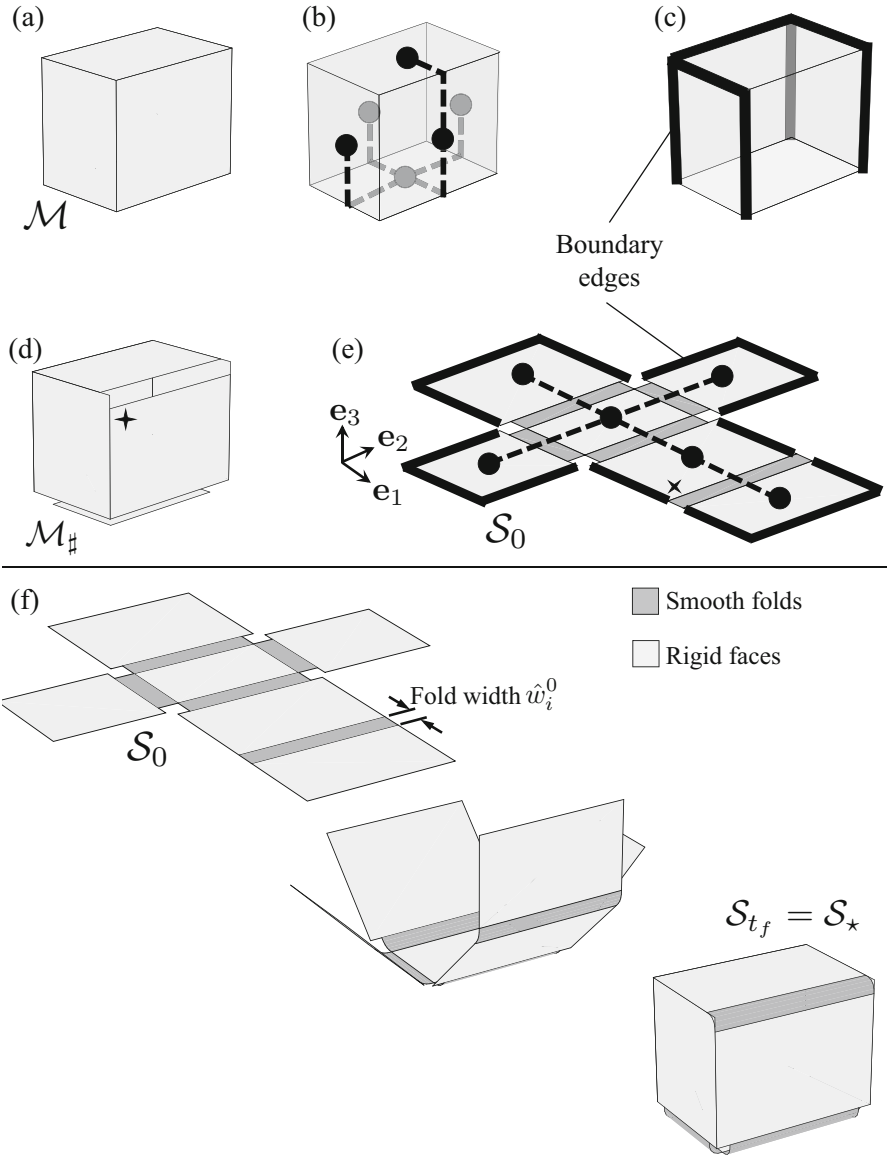


Fig. 6.3 Summary of the unfolding polyhedra method for origami with smooth folds: (**a**) Given goal mesh $\mathcal{M}$; (**b**) Spanning tree including all the faces of $\mathcal{M}$; (**c**) The edges that are not crossed by the spanning tree are assigned as boundary edges; (**d**) Trimmed mesh $\mathcal{M}_\sharp$ computed in order to accommodate smooth folds; (**e**) Determined net corresponding to the reference configuration $\mathcal{S}_0$ of an origami sheet. This sheet contains the faces of $\mathcal{M}_\sharp$ and the introduced smooth folds; (**f**) Continuous folding motion from $t = 0$ to $t = t_f$ such that the final configuration $\mathcal{S}_{t_f}$ corresponds to the goal configuration $\mathcal{S}_\star$ that approximates $\mathcal{M}$

comprised of a specified material are able to achieve their required fold angles in the goal configuration $\mathcal{S}_\star$. Such a physically based determination of the fold widths is a topic of Chap. 8.

6.2.2 *Face Trimming Step*

As stated in Sect. 6.2.1, once a spanning tree has been determined for a given goal mesh $\mathcal{M}$ and a cut boundary has been obtained based on that spanning tree, we proceed with the *face trimming* step. In this chapter, we use the procedure described in Sect. 3.2.3 to determine the spanning tree. As illustrated in Fig. 6.3d, the face trimming step is performed to remove regions of the mesh faces at the locations where the smooth folds will be placed in the designed origami sheet. It is remarked that the smooth folds are placed at each edge of the goal mesh $\mathcal{M}$ that is *not* assigned as a boundary edge based on the spanning tree.

Figure 6.4a shows the side view of two faces in $\mathcal{M}$ sharing a common edge that is not assigned as boundary edge. Figure 6.4b shows the side view of the associated faces connected by the introduced smooth fold in the goal configuration $\mathcal{S}_\star$ of the designed origami sheet. It is observed in this figure that due to the bending deformation of the smooth fold (as opposed to creasing), the two faces connected by the smooth fold do not share a common edge in $\mathcal{S}_\star$. To account for this, the faces of the goal mesh $\mathcal{M}$ are recomputed to generate the *trimmed mesh* $\mathcal{M}_\sharp$ (see Fig. 6.3d). As shown in Fig. 6.4b, the regions removed at the faces compensate for the space occupied by the introduced smooth fold and have associated *trim lengths* $\hat{l}_i$, $i = 1, \ldots, N_\mathcal{F}$.

To determine the trim length $\hat{l}_i$, we first compute the *dihedral angle* $\hat{\Theta}_i$ between the two faces in $\mathcal{M}$ connected by the edge associated with the ith smooth fold (see Fig. 6.4):

$$\hat{\Theta}_i = \begin{cases} \pi + \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for convex edges} \\ \pi - \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for concave edges,} \end{cases} \qquad (6.1)$$
$$i = 1, \ldots, N_\mathcal{F},$$

where $\hat{\mathbf{n}}^{i1}, \hat{\mathbf{n}}^{i2} \in \mathbb{R}^3$ are the unit normal vectors in $\mathcal{M}$ of the two faces connected by the edge associated with the ith smooth fold. We define $\hat{w}_i^\star$ as the distance between the end points of the cross-section of the ith smooth fold in $\mathcal{S}_\star$, as shown in Fig. 6.4b. The reader is referred to Sect. 5.3 for a description of the geometry of smooth folds. Using the schematic shown in Fig. 6.4c, the trim length $\hat{l}_i$ is determined as follows:

$$\boxed{\hat{l}_i = \frac{\hat{w}_i^\star}{2} \csc\left(\frac{\hat{\Theta}_i}{2}\right).} \qquad (6.2)$$

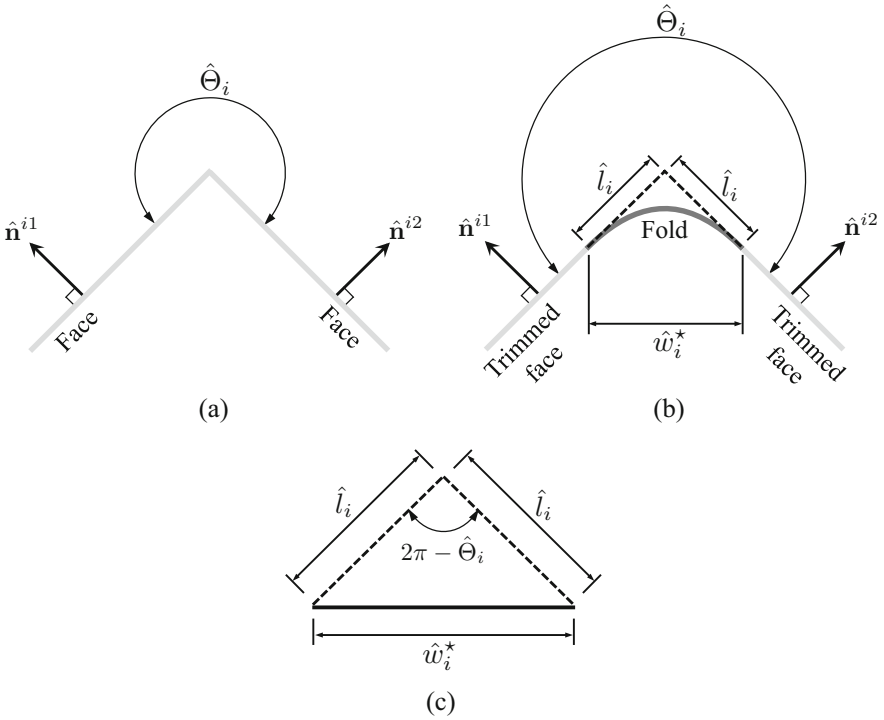


Fig. 6.4 (**a**) Side view of two faces in the goal mesh $\mathcal{M}$ sharing a common edge that is not assigned as boundary edge based on the spanning tree. (**b**) Side view of the introduced smooth fold and its adjacent trimmed faces at the goal configuration $\mathcal{S}_\star$. The geometric parameters defining the trim length $\hat{l}_i$ are shown. (**c**) Schematic used to derive (6.2)

Employing the notation of Sect. 3.2.2, the $N_\mathcal{M}$ mesh faces of $\mathcal{M}$ are denoted $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}}$ and the position vectors of the nodes of $\mathcal{M}^j$ are denoted $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^\mathcal{M}} \in \mathbb{R}^3$, where $n_j^\mathcal{M}$ is the number of nodes of $\mathcal{M}^j$. Here we denote the $N_\mathcal{M}$ *trimmed faces* of $\mathcal{M}_\sharp$ as $\mathcal{M}_\sharp^1, \ldots, \mathcal{M}_\sharp^{N_\mathcal{M}}$ and the position vectors of the nodes of $\mathcal{M}_\sharp^j$ as $\tilde{\mathbf{y}}_\sharp^{j1}, \ldots, \tilde{\mathbf{y}}_\sharp^{jn_j^\mathcal{M}} \in \mathbb{R}^3$. We now proceed to derive an expression for $\tilde{\mathbf{y}}_\sharp^{j1}, \ldots, \tilde{\mathbf{y}}_\sharp^{jn_j^\mathcal{M}}$ as functions of the node position vectors of the original mesh faces $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^\mathcal{M}}$ and the trim lengths $\hat{l}_1, \ldots, \hat{l}_{N_\mathcal{F}}$ computed using (6.2).

Figure 6.5a shows a schematic of the boundary of a mesh face $\mathcal{M}^j$ (denoted $\partial\mathcal{M}^j$) and the boundary of its associated trimmed face $\mathcal{M}_\sharp^j$ (denoted $\partial\mathcal{M}_\sharp^j$). The trimmed lengths associated with each edge of $\mathcal{M}^j$ are denoted $\tilde{l}_{jk}$, $j = 1, \ldots, N_\mathcal{M}$, $k = 1, \ldots, n_j^\mathcal{M}$, as shown in Fig. 6.5a. To determine $\tilde{l}_{jk}$ from the trimmed lengths $\hat{l}_i$ calculated in (6.2), the *trim connectivity matrix* $\mathbf{C}^\sharp$ with components $C_{jk}^\sharp$ is introduced:

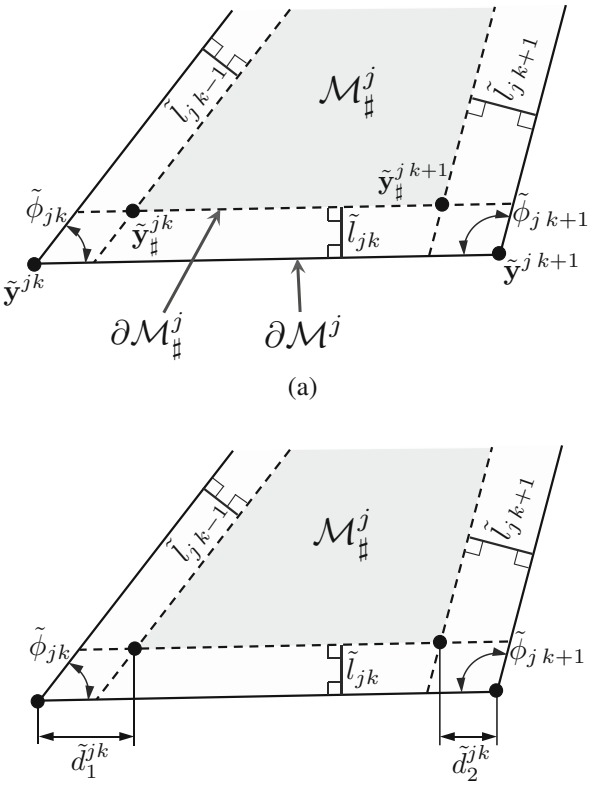


Fig. 6.5 (**a**) Boundary of a mesh face $\mathcal{M}^j$ (denoted $\partial\mathcal{M}^j$) and its associated trimmed face $\mathcal{M}^j_\sharp$ (denoted $\partial\mathcal{M}^j_\sharp$). (**b**) Edge trim lengths $\tilde{d}_1^{jk}$ and $\tilde{d}_2^{jk}$

$$\begin{aligned} C^\sharp_{jk} &= \text{Index of the edge in } \mathcal{M} \text{ corresponding to the } k^{\text{th}}\text{edge (in} \\ &\qquad \text{counterclockwise order) of } \mathcal{M}^j \\ j &= 1, \ldots, N_\mathcal{M}, \qquad k = 1, \ldots, n^\mathcal{M}_j. \end{aligned} \tag{6.3}$$

The components of $\mathbf{C}^\sharp$ can be determined from the given goal mesh data listed in Table 3.1 (see Problem 6.3).

The mapping from $\hat{l}_1, \ldots, \hat{l}_{N_\mathcal{F}}$ computed using (6.2) to the trim lengths of the edges of each mesh face $\mathcal{M}^j$ $(\tilde{l}_{j1}, \ldots, \tilde{l}_{jn^\mathcal{M}_j})$ is given as follows:

$$\tilde{l}_{jk} = \hat{l}_{C^\sharp_{jk}} \qquad j = 1, \ldots, N_\mathcal{M}, \quad k = 1, \ldots, n^\mathcal{M}_j. \tag{6.4}$$

The position vectors of the nodes of the trimmed face $\mathcal{M}^j_\sharp$ are then determined as follows (refer to Fig. 6.5a):

$$\tilde{\mathbf{y}}_{\sharp}^{jk} = \tilde{\mathbf{y}}^{jk} + \tilde{l}_{j\,k-1}\csc(\tilde{\phi}_{jk})\frac{\tilde{\mathbf{y}}^{j\,k+1} - \tilde{\mathbf{y}}^{jk}}{\|\tilde{\mathbf{y}}^{j\,k+1} - \tilde{\mathbf{y}}^{jk}\|} + \tilde{l}_{jk}\csc(\tilde{\phi}_{jk})\frac{\tilde{\mathbf{y}}^{j\,k-1} - \tilde{\mathbf{y}}^{jk}}{\|\tilde{\mathbf{y}}^{j\,k-1} - \tilde{\mathbf{y}}^{jk}\|}, \tag{6.5}$$

where $j = 1, \ldots, N_{\mathcal{M}}$, $k = 1, \ldots, n_j^{\mathcal{M}}$. The *interior corner angles* of $\mathcal{M}^j$ denoted $\tilde{\phi}_{jk}$ are calculated as follows:

$$\tilde{\phi}_{jk} = \cos^{-1}\left(\frac{(\tilde{\mathbf{y}}^{j\,k-1} - \tilde{\mathbf{y}}^{jk}) \cdot (\tilde{\mathbf{y}}^{j\,k+1} - \tilde{\mathbf{y}}^{jk})}{\|\tilde{\mathbf{y}}^{j\,k-1} - \tilde{\mathbf{y}}^{jk}\|\,\|\tilde{\mathbf{y}}^{j\,k+1} - \tilde{\mathbf{y}}^{jk}\|}\right). \tag{6.6}$$

Once the node position vectors $\tilde{\mathbf{y}}_{\sharp}^{jk}$, $j = 1, \ldots, N_{\mathcal{M}}$, $k = 1, \ldots, n_j^{\mathcal{M}}$, have been computed using (6.5), we can fully generate the trimmed mesh $\mathcal{M}_{\sharp}$.

The *edge trim lengths* $\tilde{d}_1^{jk}$ and $\tilde{d}_2^{jk}$, $j = 1, \ldots, N_{\mathcal{M}}$, $k = 1, \ldots, n_j^{\mathcal{M}}$, quantify the reduction in length of the edges of $\mathcal{M}^j$ due to face trimming and are determined as follows (refer to Fig. 6.5b):

$$\begin{aligned} \tilde{d}_1^{jk} &= \tilde{l}_{j\,k-1}\csc(\tilde{\phi}_{jk}) + \tilde{l}_{jk}\cot(\tilde{\phi}_{jk}), \\ \tilde{d}_2^{jk} &= \tilde{l}_{j\,k+1}\csc(\tilde{\phi}_{j\,k+1}) + \tilde{l}_{jk}\cot(\tilde{\phi}_{j\,k+1}). \end{aligned} \tag{6.7}$$

As indicated in Fig. 6.5b, the reduction in length of the k^{th} edge of $\mathcal{M}^j$ is equal to $\tilde{d}_1^{jk} + \tilde{d}_2^{jk}$.

As illustrated in Fig. 6.6, as the width of the smooth folds is increased, the size of the trimmed regions also increases. In general, the *face trimming step should not reduce the length of any edge in the goal mesh faces to zero*. This requirement yields the following inequality:

$$\|\tilde{\mathbf{y}}^{j\,k+1} - \tilde{\mathbf{y}}^{jk}\| > \tilde{d}_1^{jk} + \tilde{d}_2^{jk} \qquad j = 1, \ldots, N_{\mathcal{M}}, \quad k = 1, \ldots, n_j^{\mathcal{M}}, \tag{6.8}$$

where $\|\tilde{\mathbf{y}}^{j\,k+1} - \tilde{\mathbf{y}}^{jk}\|$ is the original length of the k^{th} edge of $\mathcal{M}^j$. From (6.4) and (6.7), it is noted that $\tilde{d}_1^{jk}$ and $\tilde{d}_2^{jk}$ are functions of the trim lengths $\hat{l}_i$, $i = 1, \ldots, N_{\mathcal{F}}$. Likewise, it is noted from (6.2) that the trim lengths $\hat{l}_i, i = 1, \ldots, N_{\mathcal{F}}$, are functions of the cross-section distances $\hat{w}_i^{\star}$, $i = 1, \ldots, N_{\mathcal{F}}$, which are proportional to the *fold widths* $\hat{w}_i^0, i = 1, \ldots, N_{\mathcal{F}}$.[2] Therefore, the inequality (6.8) is an implicit constraint on the fold widths of the introduced smooth folds.

[2]Consult Sect. 5.5.2 for details on the determination of the kinematic variables $\hat{w}_i$ for the cross-section of smooth folds.

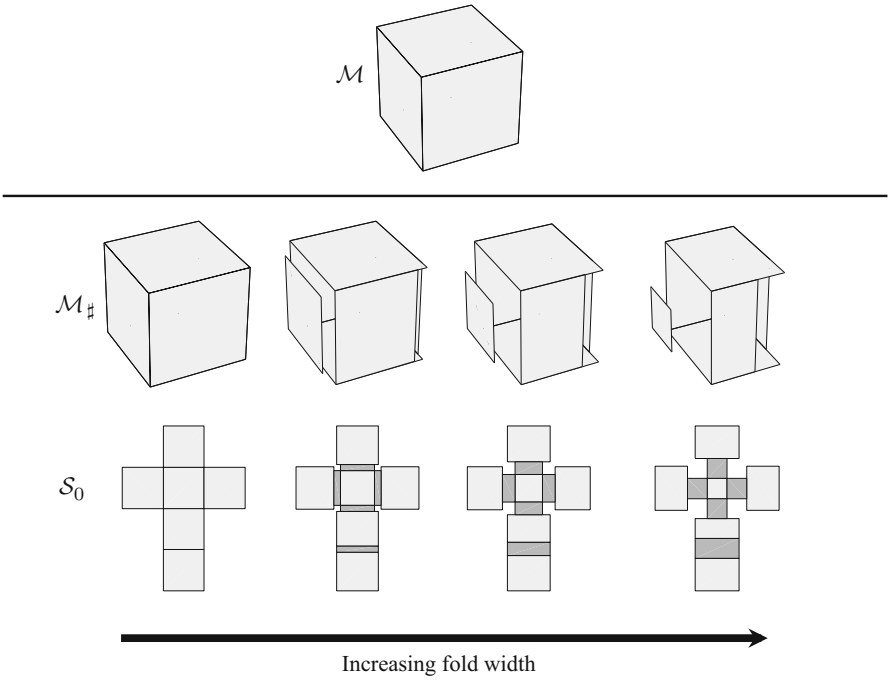


Fig. 6.6 Nets of a cube having different fold widths. The leftmost case corresponds to a net with creased (i.e., zero fold width) folds for which the face trimming step is not required

6.3 Examples of the Unfolding Polyhedra Method

This section provides representative examples of the unfolding polyhedra method for origami with smooth folds described in Sect. 6.2. We implemented such a method in MATLAB. The faces of the goal meshes, trimmed meshes, and nets are visualized as filled three-dimensional polygons using the MATLAB command `fill3` and the smooth folds are visualized as three-dimensional surfaces using the command `surf`. The complete set of MATLAB scripts used to generate the examples is included in the Supplemental Materials for this chapter and described in Appendix B.5. The MATLAB implementation of the kinematic model for origami with smooth folds presented in Chap. 5 is used to simulate the folding motion of the origami sheets corresponding to the nets. As stated in Sect. 6.2.1, the procedure used to determine a history of folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ follows the one outlined in Sect. 3.2.5.

Figure 6.7 shows the goal meshes for the surfaces of the *Platonic solids*, the computed trimmed meshes, and the corresponding nets having smooth folds. Their counterparts with creased folds are shown in Fig. 3.23. The Platonic solids are convex polyhedra constructed by congruent, regular, polygonal faces with the same number of faces meeting at each node. These include the regular tetrahedron

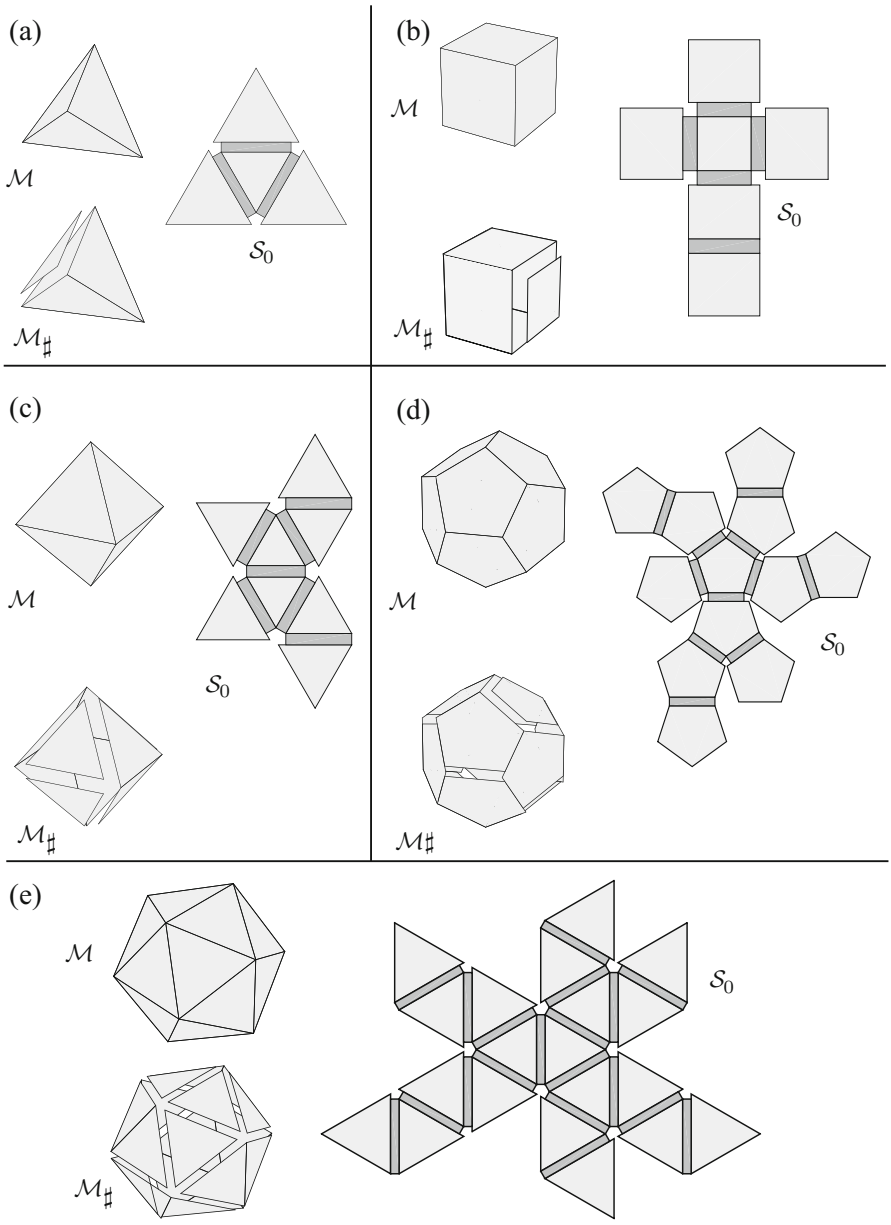


Fig. 6.7 Goal meshes for the surfaces of the Platonic solids ($\mathcal{M}$), trimmed meshes ($\mathcal{M}_\sharp$), and associated nets having smooth folds ($\mathcal{S}_0$). (**a**) Tetrahedron. (**b**) Cube. (**c**) Octahedron. (**d**) Dodecahedron. (**e**) Icosahedron

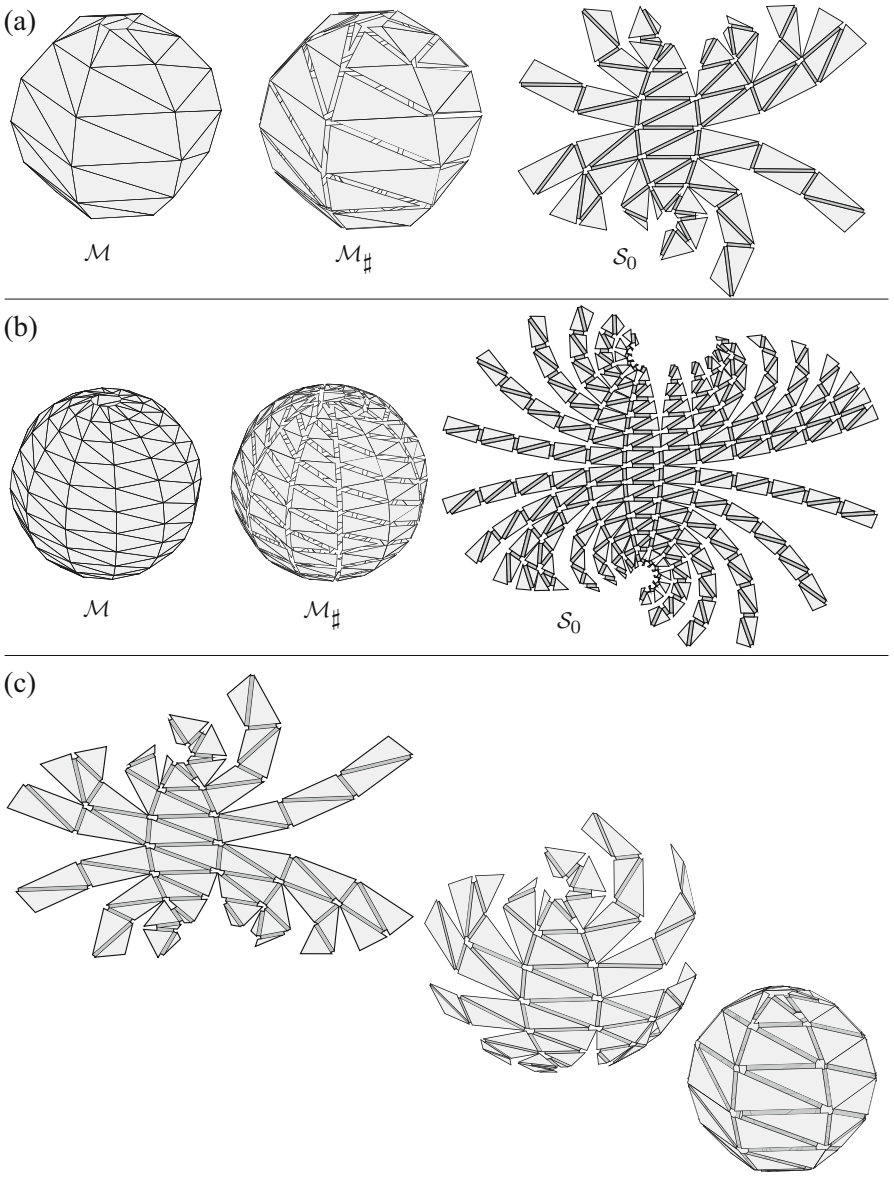


Fig. 6.8 (**a**)–(**b**) Two different mesh discretizations of a sphere and their associated nets having smooth folds. The mesh shown in (**a**) has 6 triangle pairs along the azimuthal direction and 6 triangle pairs along the inclination direction (i.e., 6×6 discretization). (**b**) 12×12 discretization. (**c**) Folding motion of the net in (**a**) associated with the 6×6 discretization

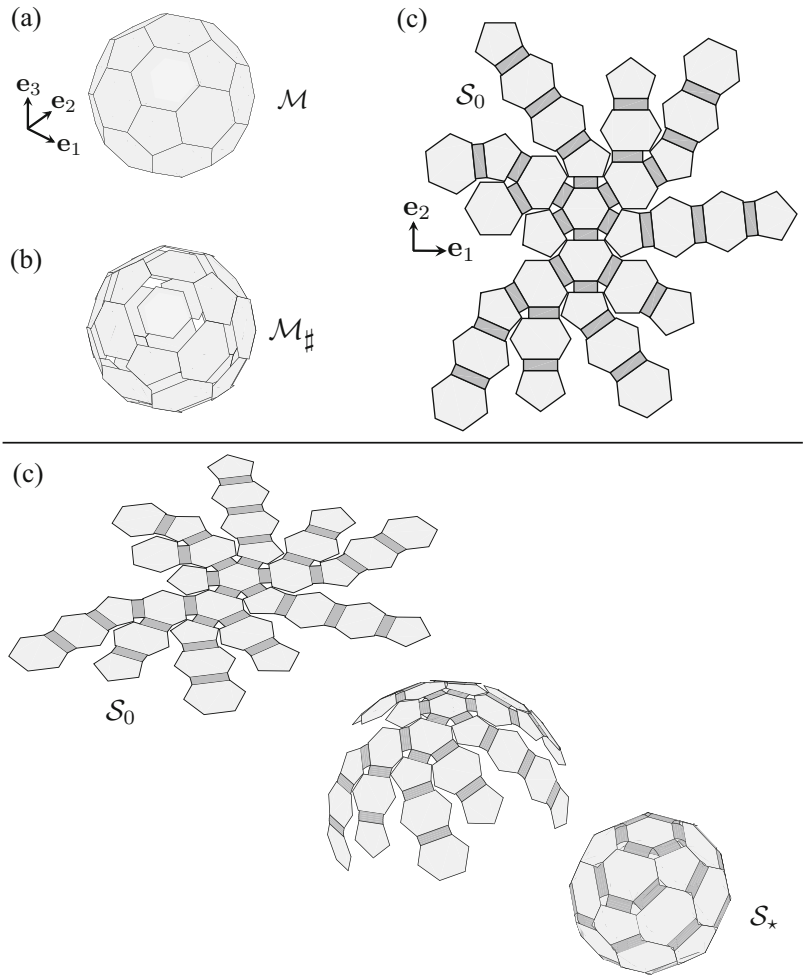


Fig. 6.9 (**a**) Goal mesh corresponding to a truncated icosahedron. (**b**) Trimmed mesh. (**c**) Net of the truncated icosahedron. (**d**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that approximates $\mathcal{M}$

(Fig. 6.7a), the cube (Fig. 6.7b), the regular octahedron (Fig. 6.7c), the regular dodecahedron (Fig. 6.7d), and the regular icosahedron (Fig. 6.7e).

Figure 6.8 shows two polygonal mesh discretizations for a sphere goal shape and their corresponding nets with smooth folds determined using the unfolding polyhedra method presented in this chapter. The trimmed meshes associated with each goal mesh are also shown.

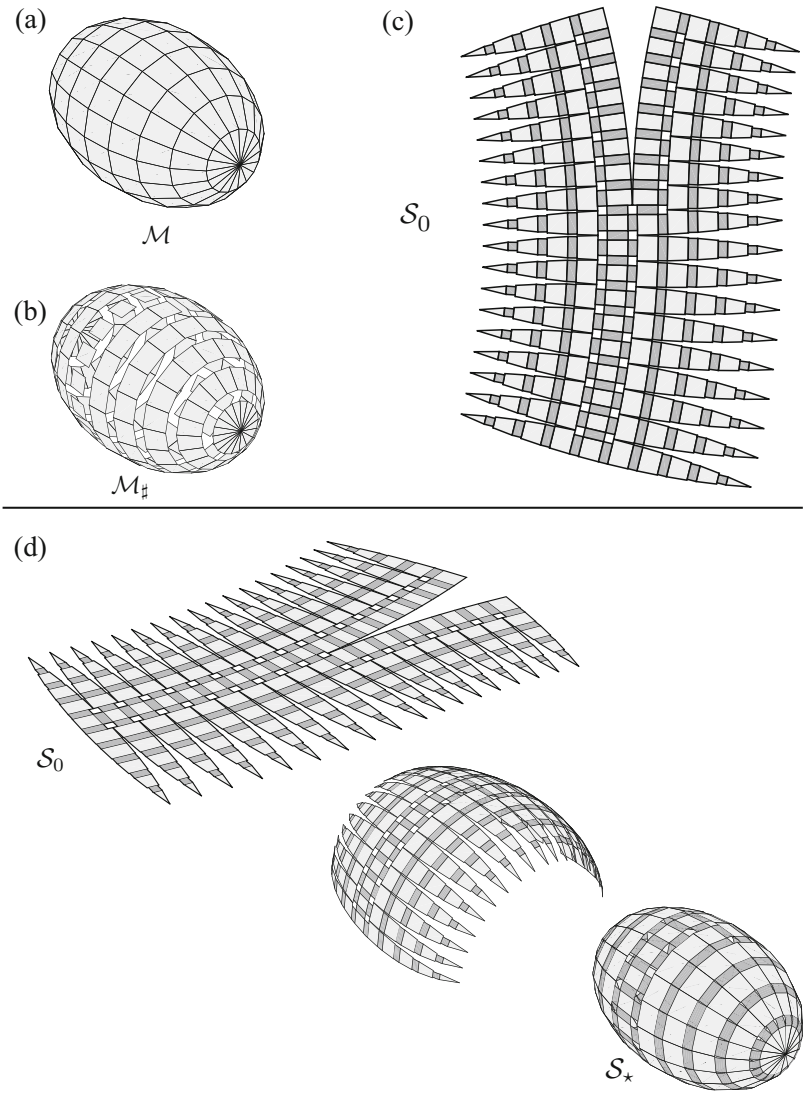


Fig. 6.10 (**a**) Goal mesh $\mathcal{M}$ for an ellipsoidal shape comprised of quadrilaterals and triangles. (**b**) Trimmed mesh. (**c**) Net of the goal mesh. (**d**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that approximates $\mathcal{M}$

Finally, Figs. 6.9, 6.10, 6.11, 6.12 show the determined nets with smooth folds and simulated folding motion associated with goal meshes for a truncated icosahedron, an ellipsoid, a shelter, and a parabolic disk. The trimmed meshes associated with each goal mesh are also shown. As observed from these figures, the unfolding polyhedra method presented in this chapter is able to successfully generate a net

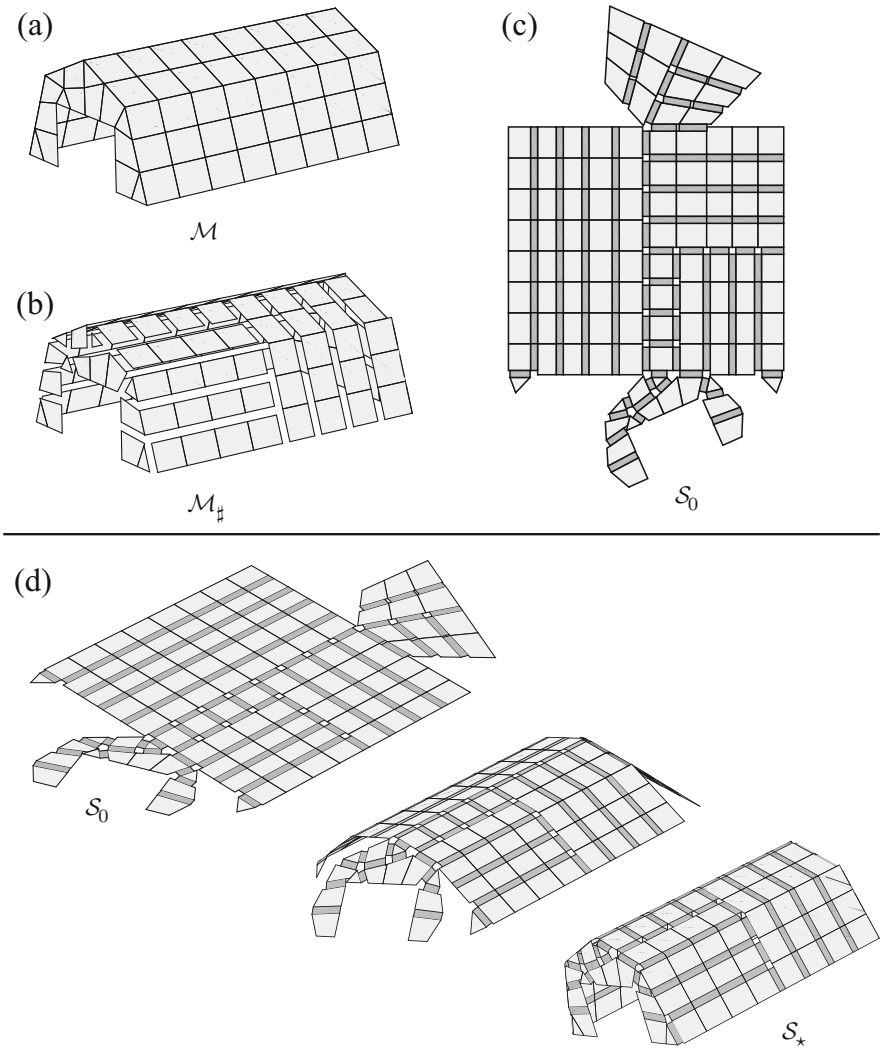


Fig. 6.11 (**a**) Goal mesh $\mathcal{M}$ for a shelter comprised of quadrilaterals and triangles. (**b**) Trimmed mesh. (**c**) Net of the goal mesh. (**d**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that approximates $\mathcal{M}$

with smooth folds that can be folded to approximate each of the considered goal meshes. Their counterparts with creased folds are shown in Figs. 3.26, 3.27, 3.28, 3.29.

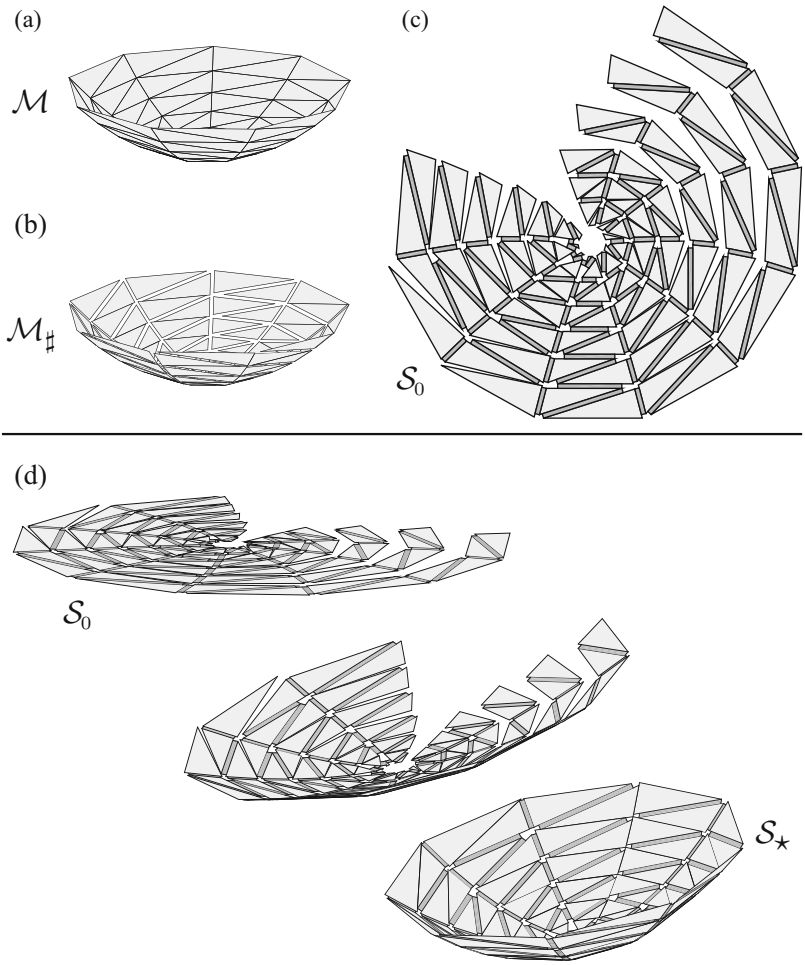


Fig. 6.12 (**a**) Goal mesh $\mathcal{M}$ for a parabolic dish. (**b**) Trimmed mesh $\mathcal{M}_\sharp$. (**c**) Net of the goal mesh. (**d**) Continuous folding motion from $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$ that approximates $\mathcal{M}$

Chapter Summary

In this chapter, we presented a method known as *unfolding polyhedra* for the design of origami structures with smooth folds that achieve targeted three-dimensional shapes. We develop this method based on the theory of unfolding polyhedra for origami with creased folds studied in Chap. 3. As such, the goal shape is represented in this method as a three-dimensional polygonal mesh termed as the *goal mesh*. The objective in unfolding polyhedra is to determine a *net* for the goal mesh, which is a connected, non-overlapping sheet corresponding to a flattening of the goal mesh onto a plane and has a boundary associated with cuts made exclusively on edges

of the goal mesh. The faces of the goal mesh are adjusted (i.e., "trimmed") at the locations where the smooth folds are placed in order to compensate for the space occupied by such folds. Thus, the net contains the trimmed faces of the goal mesh and the introduced smooth folds.

We presented the problem description and solution approach of the unfolding polyhedra method in Sect. 6.2.1. The *face trimming* step that is introduced in unfolding polyhedra for origami with smooth folds was addressed in Sect. 6.2.2. Representative examples of the implemented method considering various goal meshes were provided in Sect. 6.3. The limitations of the unfolding polyhedra method considering creased folds discussed in Sect. 3.2.6 are also applicable to the more general unfolding polyhedra method presented in this chapter.

The proposed method for origami design successfully allows for the realization of nets with folds having non-zero surface area and arbitrary order of geometric continuity. The conventional nets with creased folds studied in Chap. 3 represent a special case for this more general design method and can be obtained as well. In Chap. 8, we provide a framework for the design of active origami structures that morph towards arbitrary shapes under non-mechanical stimuli (e.g., thermal, chemical, electromagnetic) based on the unfolding polyhedra method presented in this chapter.

Problems

6.1 Derive (6.2) using the schematic shown in Fig. 6.4c.

6.2 Create a schematic such as that provided in Fig. 6.4 considering a concave edge (i.e., an edge having dihedral angle $\hat{\Theta}_i < \pi$). Then, show that (6.2) is also applicable for concave edges.

6.3 Propose an algorithm to determine the components of the trim connectivity matrix $\mathbf{C}^\sharp$, which is defined in (6.3), from the given goal mesh data listed in Table 3.1.

6.4 Derive (6.5) using the schematic shown in Fig. 6.5a.

6.5 Derive (6.7) using the schematic shown in Fig. 6.5b.

6.6 Since the resulting goal configuration $\mathcal{S}_\star$ in unfolding polyhedra for origami with smooth folds contains the trimmed mesh $\mathcal{M}_\sharp$ as a subset instead of the given goal mesh $\mathcal{M}$ (see Fig. 6.3), the approximation error between $\mathcal{S}_\star$ and $\mathcal{M}$ must be quantified. Propose parameter(s) that can be used to quantify such an error.

6.7 Following the approach presented in Sect. 3.2.4, formulate the mapping of the trimmed faces from their position in the trimmed mesh $\mathcal{M}_\sharp$ to their corresponding position in the reference configuration $\mathcal{S}_0$ of an origami sheet designed using the unfolding polyhedra method.

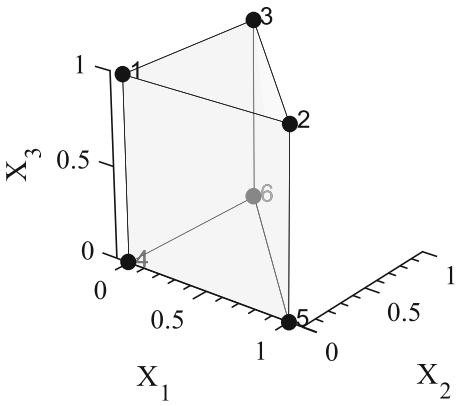


Fig. 6.13 Goal mesh $\mathcal{M}$ for Problem 6.8

6.8 Implement the unfolding polyhedra method for origami with smooth folds in a computational environment. Consider the mesh of a triangular prism shown in Fig. 6.13. The position vectors of the nodes in such a mesh are given as follows:

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\right]^\top = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{6.9}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 2 & 3 & 0 \\ 6 & 5 & 4 & 0 \\ 4 & 5 & 2 & 1 \\ 6 & 4 & 1 & 3 \\ 5 & 6 & 3 & 2 \end{bmatrix}. \tag{6.10}$$

Using the procedure to determine a spanning tree outlined in Sect. 3.2.3, find three different sheet designs for this mesh by correspondingly selecting faces 1, 3, and 5 as the reference face for the spanning tree. Assume fold widths of 0.1 for all the smooth folds.

6.9 Consider the mesh shown in Fig. 6.14. The position vectors of the nodes in such a mesh are given as follows:

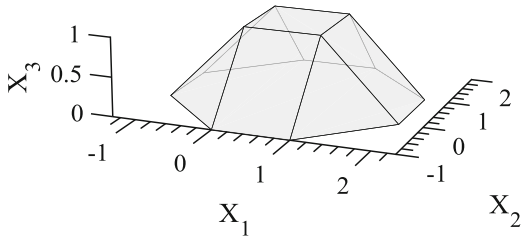


Fig. 6.14 Goal mesh $\mathcal{M}$ for Problem 6.9

$$\left[\hat{\mathbf{y}}^1 \ \cdots \ \hat{\mathbf{y}}^{12}\right]^\top = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 2 & 0 & 0 \\ 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 2 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & -1 & 0 \end{bmatrix}, \tag{6.11}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 2 & 3 & 4 & 0 & 0 & 0 & 0 \\ 1 & 5 & 6 & 2 & 0 & 0 & 0 & 0 \\ 2 & 6 & 7 & 0 & 0 & 0 & 0 & 0 \\ 3 & 2 & 7 & 8 & 0 & 0 & 0 & 0 \\ 3 & 8 & 9 & 0 & 0 & 0 & 0 & 0 \\ 4 & 3 & 9 & 10 & 0 & 0 & 0 & 0 \\ 4 & 10 & 11 & 0 & 0 & 0 & 0 & 0 \\ 1 & 4 & 11 & 12 & 0 & 0 & 0 & 0 \\ 1 & 12 & 5 & 0 & 0 & 0 & 0 & 0 \\ 12 & 11 & 10 & 9 & 8 & 7 & 6 & 5 \end{bmatrix}. \tag{6.12}$$

Using the procedure to determine a spanning tree outlined in Sect. 3.2.3, find three different sheet designs for this mesh by correspondingly selecting a triangular face, a quadrilateral face, and the octagonal face as the reference face for the spanning tree. Assume fold widths of 0.1 for all the smooth folds.

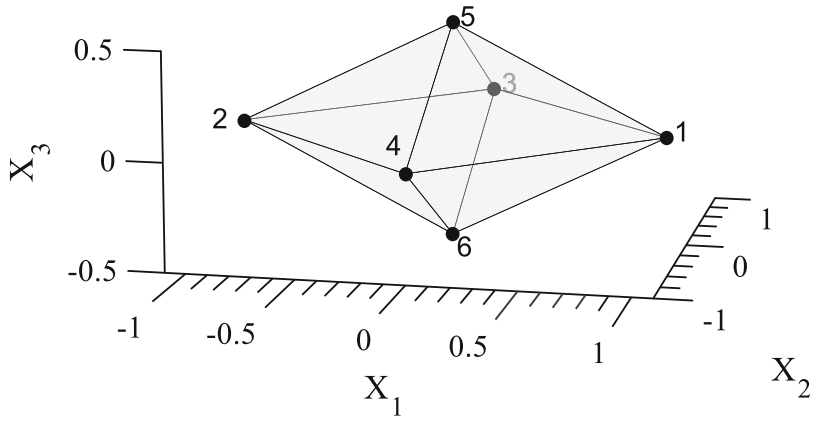


Fig. 6.15 Goal mesh $\mathcal{M}$ of an octahedron for Problem 6.10

6.10 Consider the mesh of an octahedron shown in Fig. 6.15. The position vectors of the nodes in such a mesh are given as follows:

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\right]^\top = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0.5 \\ 0 & 0 & -0.5 \end{bmatrix}, \tag{6.13}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 3 & 5 \\ 1 & 5 & 4 \\ 2 & 4 & 5 \\ 2 & 5 & 3 \\ 4 & 2 & 6 \\ 6 & 2 & 3 \\ 6 & 3 & 1 \\ 1 & 4 & 6 \end{bmatrix}. \tag{6.14}$$

Determine four different sheet designs for this mesh by correspondingly selecting fold widths of 0.1, 0.2, 0.3, and 0.4 for all the smooth folds.

6.11 Consider the mesh of an icosahedron shown in Fig. 6.16. The position vectors of the nodes in such a mesh are given as follows:

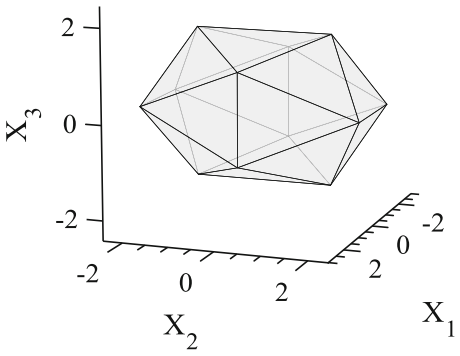


Fig. 6.16 Goal mesh $\mathcal{M}$ of an icosahedron for Problem 6.11

$$\left[\hat{\mathbf{y}}^1 \ \cdots \ \hat{\mathbf{y}}^{12}\right]^\top = \begin{bmatrix} 0 & 1.5 & 1.618 \\ 0 & 1.5 & -1.618 \\ 0 & -1.5 & 1.618 \\ 0 & -1.5 & -1.618 \\ 1 & 2.4271 & 0 \\ 1 & -2.4271 & 0 \\ -1 & 2.4271 & 0 \\ -1 & -2.4271 & 0 \\ 1.618 & 0 & 1 \\ 1.618 & 0 & -1 \\ -1.618 & 0 & 1 \\ -1.618 & 0 & -1 \end{bmatrix}, \tag{6.15}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 3 & 9 \\ 5 & 1 & 9 \\ 7 & 1 & 5 \\ 11 & 1 & 7 \\ 11 & 3 & 1 \\ 5 & 9 & 10 \\ 9 & 3 & 6 \\ 8 & 3 & 11 \\ 12 & 8 & 11 \\ 12 & 11 & 7 \\ 10 & 9 & 6 \\ 2 & 5 & 10 \\ 2 & 7 & 5 \\ 2 & 12 & 7 \\ 4 & 8 & 12 \\ 6 & 3 & 8 \\ 4 & 6 & 8 \\ 4 & 10 & 6 \\ 2 & 4 & 12 \\ 2 & 10 & 4 \end{bmatrix}. \tag{6.16}$$

Determine four different sheet designs for this mesh by correspondingly selecting fold widths of 0.1, 0.2, 0.3, and 0.4 for all the smooth folds.

6.12 Implement the unfolding polyhedra method for origami with smooth folds in a computational environment and determine nets for the 13 Archimedean polyhedra.

References

1. M. Schenk, A.D. Viquerat, K.A. Seffen, S.D. Guest, Review of inflatable booms for deployable space structures: packing and rigidization. J. Spacecr. Rocket. **51**(3), 762–778 (2014)
2. Y. Shi, F. Zhang, K. Nan, X. Wang, J. Wang, Y. Zhang, Y. Zhang, H. Luan, K.-C. Hwang, Y. Huang, J.A. Rogers, Y. Zhang, Plasticity-induced origami for assembly of three dimensional metallic structures guided by compressive buckling. Extreme Mech. Lett. **11**, 105–110 (2017)
3. Z. Yan, F. Zhang, J. Wang, F. Liu, X. Guo, K. Nan, Q. Lin, M. Gao, D. Xiao, Y. Shi, Y. Qiu, H. Luan, J.H. Kim, Y. Wang, H. Luo, M. Han, Y. Huang, Y. Zhang, J.A. Rogers, Controlled mechanical buckling for origami-inspired construction of 3D microstructures in advanced materials. Adv. Funct. Mater. **26**(16), 2629–2639 (2016)
4. N. Turner, B. Goodwine, M. Sen, A review of origami applications in mechanical engineering. Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci. (2015). https://doi.org/10.1177/0954406215597713
5. L.J. Fei, D. Sujan, Origami theory and its applications: a literature review. Int. J. Soc. Hum. Sci. Eng. **7**(1), 113–117 (2013)

6. M. Johnson, Y. Chen, S. Hovet, S. Xu, B. Wood, H. Ren, J. Tokuda, Z. Tse, Fabricating biomedical origami: a state-of-the-art review. Int. J. Comput. Assist. Radiol. Surg. **12**(11), 2023–2032 (2017)
7. Q. Ge, C.K. Dunn, H.J. Qi, M.L. Dunn, Active origami by 4D printing. Smart Mater. Struct. **23**(9), 094007 (2014)
8. H.J. In, S. Kumar, Y. Shao-Horn, G. Barbastathis, Origami fabrication of nanostructured, three-dimensional devices: electrochemical capacitors with carbon electrodes. Appl. Phys. Lett. **88**(8), 083104 (2006)
9. D. Han, S. Pal, J. Nangreave, Z. Deng, Y. Liu, H. Yan, DNA origami with complex curvatures in three-dimensional space. Science **332**(6027), 342–346 (2011)
10. C.D. Santangelo, Extreme mechanics: self-folding origami. Annu. Rev. Condens. Matter Phys. **8**, 165–183 (2017)
11. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr, D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
12. M.B. Pinson, M. Stern, A.C. Ferrero, T.A. Witten, E. Chen, A. Murugan, Self-folding origami at any energy scale. Nat. Commun. **8**, 15477 (2017)
13. Z. Song, T. Ma, R. Tang, Q. Cheng, X. Wang, D. Krishnaraju, R. Panat, C.K. Chan, H. Yu, H. Jiang, Origami lithium-ion batteries. Nat. Commun. **5**, 3140 (2014)
14. I. Shitanda, S. Kato, S. Tsujimura, Y. Hoshi, M. Itagaki, Screen-printed, paper-based, array-type, origami biofuel cell. Chem. Lett. **46**, 726–728 (2017)
15. J.P. Rojas, D. Conchouso, A. Arevalo, D. Singh, I.G. Foulds, M.M. Hussain, Paper-based origami flexible and foldable thermoelectric nanogenerator. Nano Energy **31**, 296–301 (2017)
16. J.T. Early, R. Hyde, R.L. Baron, Twenty-meter space telescope based on diffractive Fresnel lens, in *Proceedings of the SPIE's 48th Annual Meeting, Optical Science and Technology* (International Society for Optics and Photonics, San Diego, 2004), pp. 148–156
17. L. Xu, T.C. Shyu, N.A. Kotov, Origami and kirigami nanocomposites. ACS Nano **11**(8), 7587–7599 (2017)
18. G. Stoychev, M.J. Razavi, X. Wang, L. Ionov, 4D origami by smart embroidery. Macromol. Rapid Commun. **38**(18), 1700213 (2017)
19. E.D. Demaine, J. O'Rourke, *Geometric Folding Algorithms* (Cambridge University Press, Cambridge, 2007)
20. T.A. Evans, R.J. Lang, S.P. Magleby, L.L. Howell, Rigidly foldable origami gadgets and tessellations. R. Soc. Open Sci. **2**(9), 150067 (2015)
21. X. Zhou, H. Wang, Z. You, Design of three-dimensional origami structures based on a vertex approach. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. **471**(2181), 20150407 (2015)
22. W. Liu, K. Tai, Optimal design of flat patterns for 3D folded structures by unfolding with topological validation. Comput Aided Des. **39**(10), 898–913 (2007)
23. S. Bouzette, F. Buekenhout, E. Dony, A. Gottcheiner, A theory of nets for polyhedra and polytopes related to incidence geometries. Des. Codes Cryptogr. **10**(2), 115–136 (1997)
24. W. Schlickenrieder, Nets of polyhedra. Master's thesis, Technische Universität Berlin, 1997
25. M. Bern, E.D. Demaine, D. Eppstein, E. Kuo, A. Mantler, J. Snoeyink, Ununfoldable polyhedra with convex faces. Comput. Geom. **24**(2), 51–62 (2003)
26. B. Grünbaum, Nets of polyhedra II. Geombinatorics **1**(3), 5–10 (1991)
27. M. Damian, R. Flatland, J. O'Rourke, Epsilon-unfolding orthogonal polyhedra. Graph. Comb. **23**(1), 179–194 (2007)
28. S. Takahashi, H.-Y. Wu, S.H. Saw, C.-C. Lin, H.-C. Yen, Optimized topological surgery for unfolding 3D meshes. Comput. Graph. Forum **30**(7), 2077–2086 (2011)
29. E.D. Demaine, A. Lubiw, A generalization of the source unfolding of convex polyhedra, in *Spanish Meeting on Computational Geometry* (Springer, Berlin, 2011), pp. 185–199
30. Z. Xi, Y. Kim, Y.J. Kim, J.-M. Lien, Learning to segment and unfold polyhedral mesh from failures. Comput. Graph. **58**, 139–149 (2016)
31. M.A. Alam, I. Streinu, Star-unfolding polygons, in *International Workshop on Automated Deduction in Geometry* (Springer, Berlin, 2014), pp. 1–20

32. V. Chandru, R. Hariharan, N.M. Krishnakumar, Short-cuts on star, source and planar unfoldings, in *International Conference on Foundations of Software Technology and Theoretical Computer Science* (Springer, Berlin, 2004), pp. 174–185
33. S. Kiazyk, A. Lubiw, Star unfolding from a geodesic curve. Discret. Comput. Geom. **56**(4), 1018–1036 (2016)
34. J. Itoh, J. O'Rourke, C. Vîlcu, Star unfolding convex polyhedra via quasigeodesic loops. Discret. Comput. Geom. **44**(1), 35–54 (2010)
35. Y. Lee, M. Cho, Self-folding structure using light-absorption of polystyrene sheet, in *Proceedings of SPIE Smart Structures and Materials+Nondestructive Evaluation and Health Monitoring* (International Society for Optics and Photonics, San Diego, 2017), p. 101650L
36. H. He, J. Guan, J.L. Lee, An oral delivery device based on self-folding hydrogels. J. Control. Release **110**(2), 339–346 (2006)
37. J. Guan, H. He, D.J. Hansford, L.J. Lee, Self-folding of three-dimensional hydrogel microstructures. J. Phys. Chem. B **109**(49), 23134–23137 (2005)
38. G. Stoychev, N. Puretskiy, L. Ionov, Self-folding all-polymer thermoresponsive microcapsules. Soft Matter **7**(7), 3277–3279 (2011)
39. C. Yoon, R. Xiao, J. Park, J. Cha, T.D. Nguyen, D.H. Gracias, Functional stimuli responsive hydrogel devices by self-folding. Smart Mater. Struct. **23**(9), 094008 (2014)
40. S. Chen, J. Li, L. Fang, Z. Zhu, S.H. Kang, Simple triple-state polymer actuators with controllable folding characteristics. Appl. Phys. Lett. **110**(13), 133506 (2017)
41. S. Pandey, M. Ewing, A. Kunas, N. Nguyen, D.H. Gracias, G. Menon, Algorithmic design of self-folding polyhedra. Proc. Natl. Acad. Sci. **108**(50), 19885–19890 (2011)
42. Y. Liu, J.K. Boyles, J. Genzer, M.D. Dickey, Self-folding of polymer sheets using local light absorption. Soft Matter **8**(6), 1764–1769 (2012)

Chapter 7
Tuck-Folding Method for the Design of Origami Structures with Smooth Folds

Abstract Having addressed the unfolding polyhedra method for the design of origami structures with smooth folds in the previous chapter, here we consider a design method applicable to a much larger spectrum of three-dimensional goal shapes. This chapter presents the *tuck-folding method* for origami with smooth folds based on the same method presented in Chap. 4 for origami with creased folds. The present method generates a sheet comprised of the faces of the goal mesh in addition to introduced regions having two rigid faces and three smooth folds. The smooth folds are used to *tuck*-fold the added regions to form the shape of the goal mesh. Representative examples considering goal shapes having different geometries are provided.

7.1 Introduction

As reviewed in Sects. 1.3 and 6.1, current methods for origami design are generally limited to the idealization of the folds as creases of zeroth-order geometric continuity. Such an idealization is not proper for origami structures having non-negligible fold thickness or with maximum curvature at the folds restricted by material limitations [1–3] (e.g., *active origami structures*). For such structures, the folds are not properly represented as creases but instead as bent regions of higher-order geometric continuity (i.e., *smooth folds*; refer to Sect. 5.2).

This chapter presents a novel method for origami design having its inspiration from the *tuck-folding method* presented by Tachi in [4, 5] and revisited in Chap. 4 but considering smooth folds as opposed to creased folds. The outline of this chapter is as follows: Sect. 7.2 presents the problem description and solution approach of the tuck-folding method for origami with smooth folds, examples of the computationally implemented method considering various goal shapes are provided

Electronic Supplementary Material The online version of this article (https://doi.org/10.1007/978-3-319-91866-2_7) contains supplementary material, which is available to authorized users.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_7

in Sect. 7.3, and a study on the design and fabrication of shape memory polymer (SMP) self-folding sheets using the tuck-folding method is presented in Sect. 7.3.1.

7.2 Tuck-Folding Method Considering Smooth Folds

In this section, we describe the various aspects of the tuck-folding method for the design of origami structures with smooth folds. Section 7.2.1 provides the design problem definition and the solution approach, Sect. 7.2.2 describes the face trimming step, and the edge module parameterization and associated design constraints are addressed in Sect. 7.2.3.

7.2.1 Problem Definition

The tuck-folding method for origami with smooth folds aims to solve the following origami design problem:

- *Given*: A three-dimensional goal shape represented as a polygonal mesh (i.e., the *goal mesh*[1] $\mathcal{M}$)
- *Find*: The shape and fold pattern of a planar sheet with smooth folds that can be folded to match $\mathcal{M}$, and a history of folding motion from the planar configuration of the sheet ($\mathcal{S}_0$) to the goal configuration ($\mathcal{S}_\star$) that matches $\mathcal{M}$

The steps followed in the tuck-folding method for origami with smooth folds to solve this design problem are listed as follows [6]:

1. If the given goal mesh is not topologically equivalent to a disk [4],[2] interior edges of the mesh are assigned as boundary edges to introduce an additional boundary $\partial\mathcal{M}^C$ such that the obtained mesh satisfies the aforementioned topological property. A goal mesh $\mathcal{M}$ with boundary $\partial\mathcal{M} = \partial\mathcal{M}^C \cup \partial\mathcal{M}^O$ and mesh faces $\mathcal{M}^1, \ldots, \mathcal{M}^{N_\mathcal{M}} \subset \mathcal{M}$ (where $N_\mathcal{M}$ is the total number of faces in $\mathcal{M}$) is then obtained (the boundary $\partial\mathcal{M}^O$ consists of the boundary edges of the originally given mesh). We illustrate these concepts in Fig. 4.4
2. The faces of $\mathcal{M}$ are adjusted (i.e., "trimmed") at the locations where the *edge modules* will be introduced in Step 3. These locations correspond to the edges of $\mathcal{M}$ that are *not* assigned as boundary edges in Step 1. This step is referred to as *face trimming*. The resulting *trimmed mesh* is denoted $\mathcal{M}_\sharp$ as shown in Fig. 7.1 and its *trimmed faces* are denoted $\mathcal{M}_\sharp^1, \ldots, \mathcal{M}_\sharp^{N_\mathcal{M}} \subset \mathcal{M}_\sharp$

[1] The goal mesh $\mathcal{M}$ is a *connected*, *orientable*, *2-manifold* polygonal mesh.

[2] Consult Sects. 4.2.1 and 4.2.2 for a discussion on the topology required for the goal mesh $\mathcal{M}$ to allow for the application of the tuck-folding method.

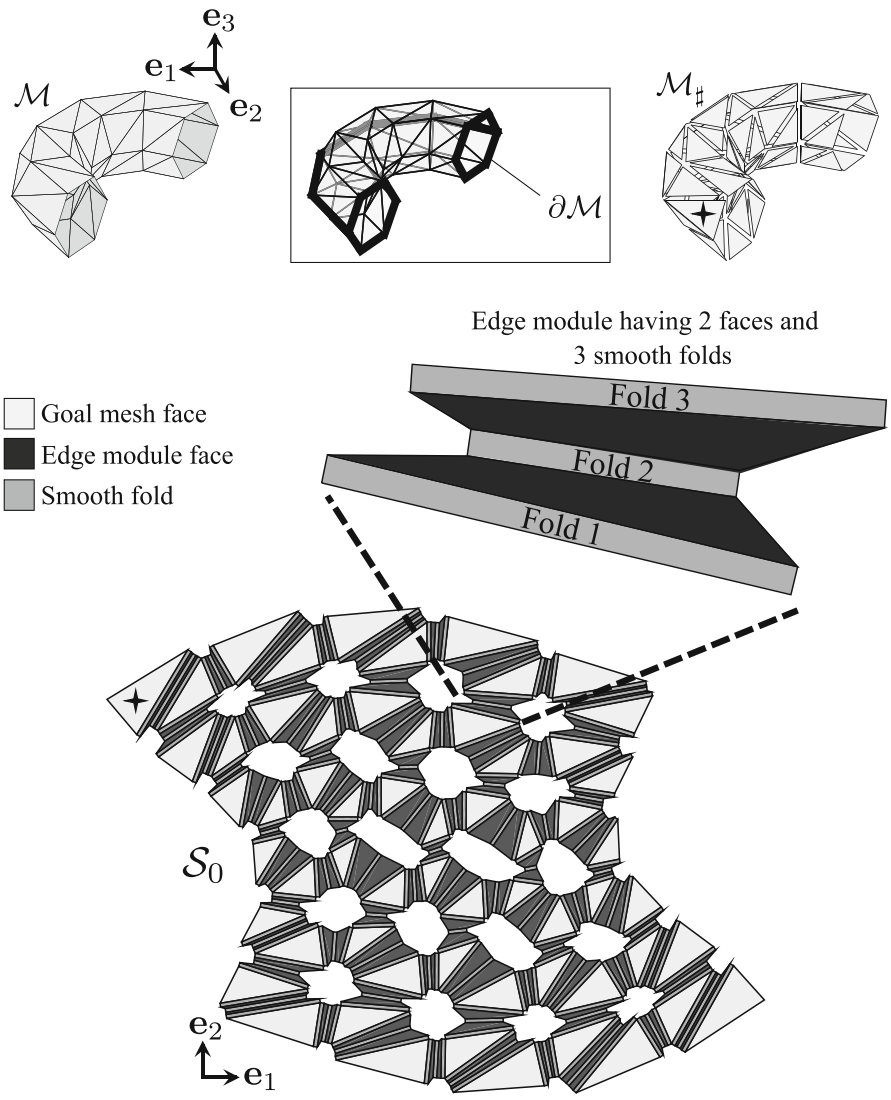


Fig. 7.1 Schematic illustrating Steps 2 and 3 of the tuck-folding method for origami with smooth folds presented in this chapter: Given a goal mesh $\mathcal{M}$ with boundary $\partial\mathcal{M}$, compute the *trimmed mesh* $\mathcal{M}_\sharp$ and then find the reference configuration $\mathcal{S}_0$ of an origami sheet that approximates $\mathcal{M}$ in a known folded configuration. The reference configuration $\mathcal{S}_0$ is comprised of the faces of $\mathcal{M}_\sharp$ mapped onto a plane and *edge modules* having two rigid faces and three smooth folds introduced at all the interior edges of $\mathcal{M}$

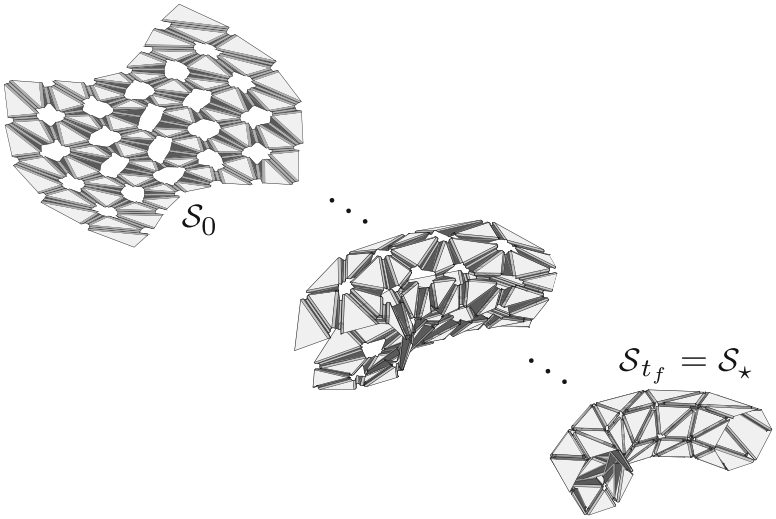


Fig. 7.2 Schematic illustrating Step 4 of the tuck-folding method for origami with smooth folds: Folding motion of an origami sheet determined in Step 3 from the reference configuration $\mathcal{S}_0$ towards the goal configuration $\mathcal{S}_\star$ that approximates the goal mesh $\mathcal{M}$ (note that $\mathcal{M}_\sharp \subset \mathcal{S}_\star$; refer to Fig. 7.1)

3. An origami sheet with reference configuration $\mathcal{S}_0$ comprised of the trimmed faces $\mathcal{M}_\sharp^1, \ldots, \mathcal{M}_\sharp^{N_\mathcal{M}}$ mapped onto a plane and $N_\mathcal{E}^I$ introduced *edge modules* (where $N_\mathcal{E}^I$ is identically the *number of interior edges* of $\mathcal{M}$) is then determined as shown in Fig. 7.1. The edge modules consist of *two rigid faces and three smooth folds* and are placed between every pair of faces of $\mathcal{M}$ connected by an interior edge, hence their name. Accordingly, $\mathcal{S}_0$ has $3N_\mathcal{E}^I$ smooth folds and $2N_\mathcal{E}^I + N_\mathcal{M}$ rigid faces:

$$N_\mathcal{F} = 3N_\mathcal{E}^I, \qquad N_\mathcal{P} = 2N_\mathcal{E}^I + N_\mathcal{M}, \tag{7.1}$$

   where $N_\mathcal{F}$ and $N_\mathcal{P}$ are, respectively, the number of folds and faces of the designed origami sheet. The challenge in this step is to determine the geometry of the edge modules such that a valid reference configuration $\mathcal{S}_0$ is obtained (refer to Sect. 5.2) and such a sheet approximates $\mathcal{M}$ in a known *goal configuration* $\mathcal{S}_\star$ (Fig. 7.2)
4. The final step entails the determination of a history of folding motion from the reference configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$. During folding motion, each edge module is *tuck-folded* to form the shape of the goal mesh $\mathcal{M}$. Figure 7.2 shows configurations attained by the sheet of Fig. 7.1 during a determined history of folding motion from $\mathcal{S}_0$ to $\mathcal{S}_\star$

Table 7.1 Values for the fold width and the kinematic variables of the smooth folds in the goal configuration $\mathcal{S}_\star$

	Fold width	Fold angle at $\mathcal{S}_\star$	Distance between cross-section end points at $\mathcal{S}_\star$
Fold 1	$\hat{w}_i^{\mathsf{E}0}$	$-\hat{\Theta}_i/2$	$\hat{w}_i^{\mathsf{E}\star}$
Fold 2	$\hat{w}_i^{\mathsf{I}0}$	π	$\hat{w}_i^{\mathsf{I}\star}$
Fold 3	$\hat{w}_i^{\mathsf{E}0}$	$-\hat{\Theta}_i/2$	$\hat{w}_i^{\mathsf{E}\star}$

Refer to Fig. 7.4a for the numbering of the folds

To define the goal mesh $\mathcal{M}$, we employ the same data used in the tuck-folding method for origami with creased folds. Such data used to define the goal mesh and any required mesh modifications are described in Sect. 4.2.2.

7.2.2 *Face Trimming Step*

As stated in Sect. 7.2.1, once a goal mesh $\mathcal{M}$ topologically equivalent to a disk has been determined, we proceed to the *face trimming* step. As illustrated in Fig. 7.1, the face trimming step is performed to remove regions of the mesh faces at the locations where the edge modules will be placed in the designed origami sheet (Fig. 7.3b). It is remarked that the edge modules are placed at each interior edge of the goal mesh $\mathcal{M}$. The regions removed at the mesh faces compensate for the space occupied by the tuck-folded edge modules in the goal configuration $\mathcal{S}_\star$ (Fig. 7.3d).

A side view of a tuck-folded edge module is shown in Fig. 7.4. It is assumed that each edge module is tuck-folded in a *symmetric* manner and therefore the exterior folds for each edge module (Folds 1 and 3 in Fig. 7.4a) have the same values for their kinematic variables at $\mathcal{S}_\star$. Such kinematic variables are provided in Table 7.1. The interior fold (Fold 2 in Fig. 7.4a) has a fold angle of π radians when the edge module is tuck-folded at $\mathcal{S}_\star$. It is observed in Fig. 7.4a that due to the bending deformation of the smooth folds (as opposed to creasing), a gap results between two mesh faces connected by the edge module in the goal configuration $\mathcal{S}_\star$. To account for this gap, the faces of the goal mesh are recomputed to generate the *trimmed* mesh as shown in Fig. 7.1. The removed regions compensate for the gaps and have associated *trim lengths* $\hat{l}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, as shown in Fig. 7.4a. To define these removed regions, first the dihedral angle $\hat{\Theta}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, between the two faces connected by the ith interior edge is calculated as follows (see Fig. 7.4a):

$$\hat{\Theta}_i = \begin{cases} \pi + \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for convex edges} \\ \\ \pi - \cos^{-1}(\hat{\mathbf{n}}^{i1} \cdot \hat{\mathbf{n}}^{i2}); & \text{for concave edges,} \end{cases} \tag{7.2}$$

$$i = 1, \ldots, N_{\mathcal{E}}^I,$$

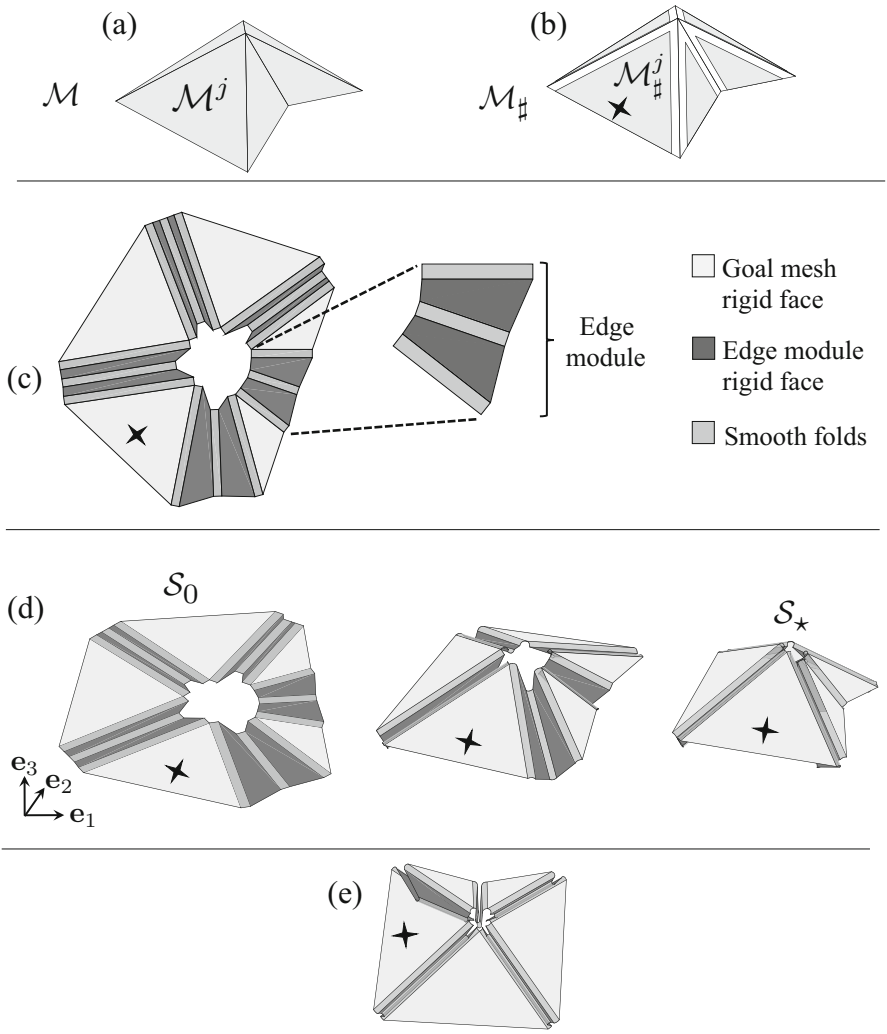


Fig. 7.3 (**a**) Example goal mesh $\mathcal{M}$ having one interior node; (**b**) Trimmed mesh $\mathcal{M}_\sharp$; (**c**) Determined sheet geometry and fold pattern associated with the goal mesh shown in (**a**); (**d**) Folding motion from the reference planar configuration $\mathcal{S}_0$ to the goal configuration $\mathcal{S}_\star$. Note that $\mathcal{M}_\sharp \subset \mathcal{S}_\star$ under appropriate rigid transformations; (**e**) View of the non-intersecting tuck-folded edge modules in the goal configuration $\mathcal{S}_\star$

where $\hat{\mathbf{n}}^{i1}$ and $\hat{\mathbf{n}}^{i2}$ are the unit normal vectors of the faces in $\mathcal{M}$ connected by the ith interior edge as shown in Fig. 7.4a.

Let $\hat{w}_i^{\downarrow\star}$ be the distance between the end points of the cross-section of the interior fold in the ith edge module at $\mathcal{S}_\star$ (refer to Table 7.1). Also, let $w_i^{\mathsf{E}\star}$ be the distance

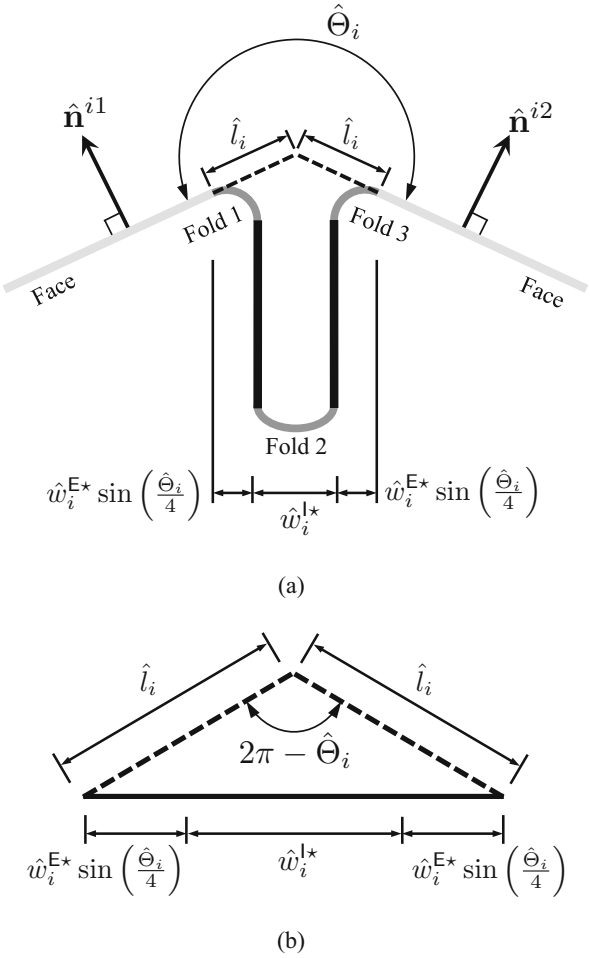


Fig. 7.4 (**a**) Side view of a tuck-folded edge module and its adjacent mesh faces in the goal configuration $\mathcal{S}_\star$. The geometric parameters defining the trim length $\hat{l}_i$ are shown. (**b**) Schematic used to derive (7.3)

between the cross-section end points of the exterior folds. Using the schematic shown in Fig. 7.4b, we calculate the trim lengths $\hat{l}_i$ as follows:

$$\hat{l}_i = \left(\frac{\hat{w}_i^{\mathsf{I}\star}}{2} + \hat{w}_i^{\mathsf{E}\star} \sin\left(\frac{\hat{\Theta}_i}{4}\right) \right) \csc\left(\frac{\hat{\Theta}_i}{2}\right) \qquad i = 1, \ldots, N_{\mathcal{E}}^I. \tag{7.3}$$

Employing the notation of Sect. 6.2.2, the position vectors of the nodes of the trimmed faces $\mathcal{M}_\sharp^j$ are denoted as $\tilde{\mathbf{y}}_\sharp^{j1}, \ldots, \tilde{\mathbf{y}}_\sharp^{jn_j^{\mathcal{M}}} \in \mathbb{R}^3$, $j = 1, \ldots, N_{\mathcal{M}}$. The expressions for $\tilde{\mathbf{y}}_\sharp^{j1}, \ldots, \tilde{\mathbf{y}}_\sharp^{jn_j^{\mathcal{M}}}$ as functions of the node position vectors of the

original mesh faces $\tilde{\mathbf{y}}^{j1}, \ldots, \tilde{\mathbf{y}}^{jn_j^{\mathcal{M}}}$ and the trim lengths $\hat{l}_1, \ldots, \hat{l}_{N_{\mathcal{E}}^I}$ are provided in (6.5).

Once a goal mesh $\mathcal{M}$ topologically equivalent to a disk has been determined in Step 1 and the associated trimmed mesh $\mathcal{M}_\sharp$ has been computed in Step 2, Step 3 of the tuck-folding method for origami with smooth folds addresses the design of a planar sheet that can be folded to approximate $\mathcal{M}$ (refer to Sect. 7.2.1). The designed origami sheet is comprised of the trimmed faces $\mathcal{M}_\sharp^1, \ldots, \mathcal{M}_\sharp^{N_{\mathcal{M}}}$ mapped onto a plane and $N_{\mathcal{E}}^I$ introduced *edge modules*, each consisting of *two rigid faces and three smooth folds* (Fig. 7.3c). The edge modules are placed between every pair of faces of $\mathcal{M}$ connected by an interior edge. The challenge in this step is to determine the geometry of the edge modules such that a valid reference configuration $\mathcal{S}_0$ is obtained (refer to Sect. 5.2) and the designed sheet approximates $\mathcal{M}$ in a known *goal configuration* $\mathcal{S}_\star$ (Fig. 7.3d–e). To address this challenge, we first study the parameterization of the edge modules and derive the associated constraints on the design variables. We then use the numerical procedure outlined in Sect. 4.2.5 to solve for the design variables that satisfy the derived constraints.

7.2.3 Edge Module Parameterization and Constraints

In this chapter, the *fold widths* of the smooth folds in each edge module are assumed *given*.[3] In practice, the fold widths are determined such that the smooth folds having a specified thickness and comprised of a specified material are able to achieve their required fold angles in the goal configuration $\mathcal{S}_\star$ (refer to Table 7.1). Such a physically-based determination of the fold widths is a topic of Chap. 8.

Following the approach of Chap. 4, we parameterize each edge module by a *length variable* denoted $\hat{W}_i$ and an *angular variable* denoted $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$. These variables are schematically shown in Fig. 7.5a. The modified edge module of Fig. 7.5a accounting for the change in edge lengths due to face trimming (refer to Sect. 7.2.2) is shown in Fig. 7.5b. The *edge module trim lengths* $\hat{d}_{nm}^i$, $n = 1, 2$, $m = 1, 2$, shown therein correspond to the edge trim lengths $\tilde{d}_m^{jk}$, $m = 1, 2$ (see (6.7)), of the two faces connected to the ith interior edge.

7.2.3.1 Loop Closure Constraints

The variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, must be determined such that the faces of $\mathcal{M}$ and the edge modules associated with each interior node of $\mathcal{M}$ *form a closed loop in* $\mathcal{S}_0$ (Fig. 7.3c). Let W_{jk} and ψ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, be the variables $\hat{W}_i$ and $\hat{\psi}_i$, respectively, of the edge module associated with the kth

[3] Refer to Sect. 5.3 for a detailed description of the geometry of smooth folds.

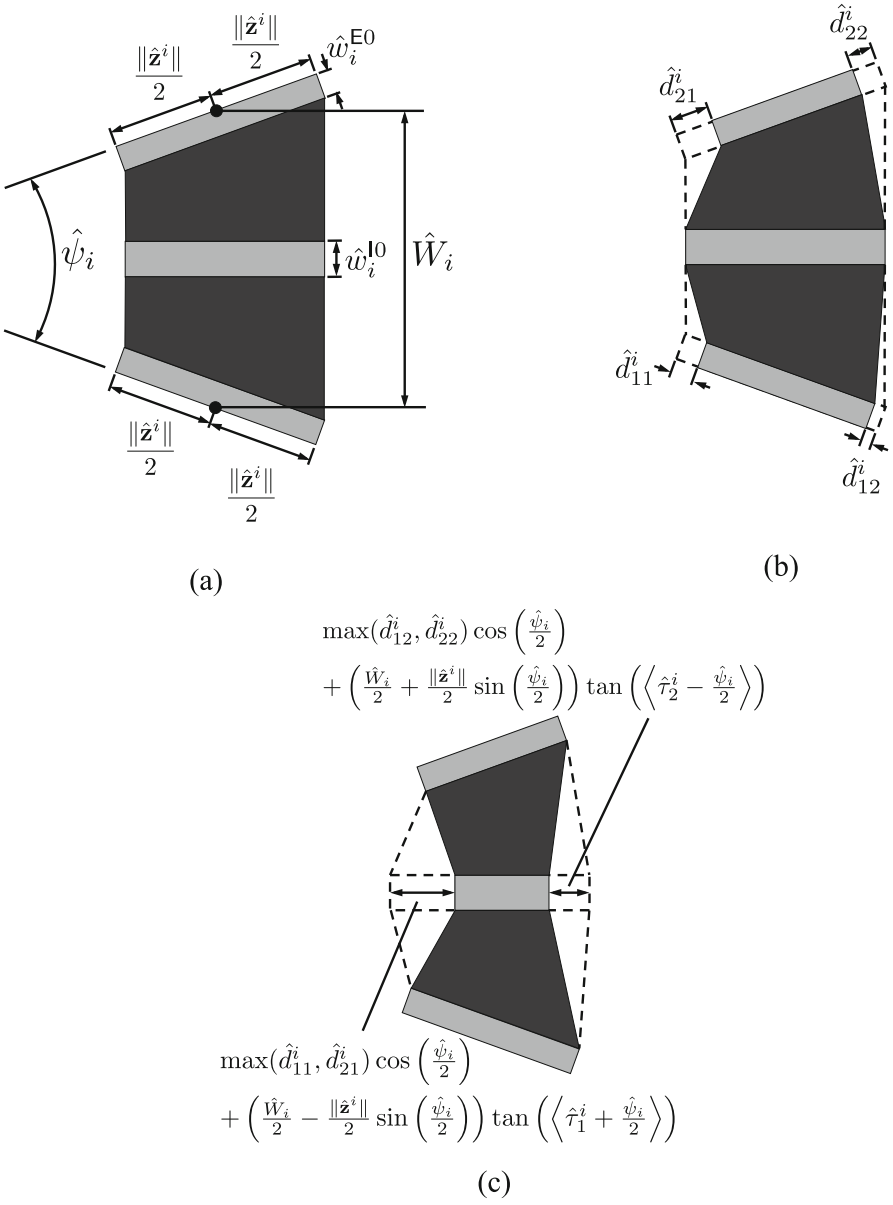


Fig. 7.5 (**a**) Edge module and associated geometric parameters; (**b**) Modified edge module accounting for the change in edge lengths due to face trimming; (**c**) Edge module trimmed accounting for self-intersection avoidance at the goal configuration $\mathcal{S}_\star$

interior edge incident to the jth interior node of $\mathcal{M}$. It is recalled that $n_j^{\mathcal{N}}$ is the number of edges incident to the jth interior node of $\mathcal{M}$. The mapping from $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, to W_{jk} and ψ_{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, is given as follows:

$$W_{jk} = \hat{W}_{|C_{jk}^{\mathcal{N}_I}|}, \qquad \psi_{jk} = \begin{cases} \hat{\psi}_{C_{jk}^{\mathcal{N}_I}}; & C_{jk}^{\mathcal{N}_I} > 0 \\ -\hat{\psi}_{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} < 0, \end{cases} \tag{7.4}$$

$$j = 1, \ldots, N_{\mathcal{N}}^I, \qquad k = 1, \ldots, n_j^{\mathcal{N}}.$$

where $\mathbf{C}^{\mathcal{N}_I}$ is the *node connectivity matrix* defined in (3.16). Also, let d_{11}^{jk} and d_{21}^{jk} be the edge module trim lengths $\hat{d}_{nm}^i$ of the kth edge module associated with the jth interior node of $\mathcal{M}$. The mapping from $\hat{d}_{nm}^i$, $n = 1, 2$, $m = 1, 2$, to d_{11}^{jk} and d_{21}^{jk}, $j = 1, \ldots, N_{\mathcal{N}}^I$, $k = 1, \ldots, n_j^{\mathcal{N}}$, is given as follows:

$$d_{11}^{jk} = \begin{cases} \hat{d}_{11}^{C_{jk}^{\mathcal{N}_I}}; & C_{jk}^{\mathcal{N}_I} > 0 \\ \hat{d}_{22}^{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} < 0, \end{cases} \qquad d_{21}^{jk} = \begin{cases} \hat{d}_{21}^{C_{jk}^{\mathcal{N}_I}}; & C_{jk}^{\mathcal{N}_I} > 0 \\ \hat{d}_{12}^{|C_{jk}^{\mathcal{N}_I}|}; & C_{jk}^{\mathcal{N}_I} < 0, \end{cases} \tag{7.5}$$

$$j = 1, \ldots, N_{\mathcal{N}}^I, \qquad k = 1, \ldots, n_j^{\mathcal{N}}.$$

To derive constraints on W_{jk} and ψ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, allowing the faces of $\mathcal{M}_\sharp$ and the edge modules associated with the jth interior node of $\mathcal{M}$ to form a closed loop in $\mathcal{S}_0$, first let $\mathbf{b}^{j0}, \ldots, \mathbf{b}^{jn_j^{\mathcal{N}}} \in \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ be the position vectors of the inner corners of the edge modules as shown in Fig. 7.6a (numbered in counterclockwise order). The vectors $\mathbf{b}^{j0}, \ldots, \mathbf{b}^{jn_j^{\mathcal{N}}}$ can be determined recursively as follows:

$$\mathbf{b}^{jk} = \mathbf{b}^{j\,k-1} + \mathbf{R}_3 \left(\sum_{l=1}^{k-1} \left(\frac{\psi_{jl}}{2} + \phi_{jl} + \frac{\psi_{j\,l+1}}{2} \right) \right)$$

$$\times \begin{bmatrix} (d_{21}^{jk} - d_{11}^{jk}) \cos\left(\frac{\psi_{jk}}{2}\right) \\ W_{jk} + (d_{11}^{jk} + d_{21}^{jk} - \|\mathbf{z}^{jk}\|) \sin\left(\frac{\psi_{jk}}{2}\right) \\ 0 \end{bmatrix},$$

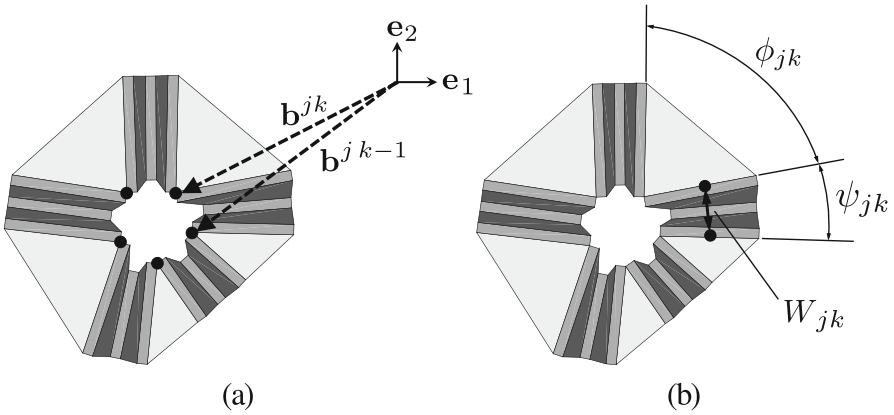


Fig. 7.6 Trimmed faces and edge modules in $\mathcal{S}_0$ associated with the jth interior node of $\mathcal{M}$: (**a**) Position vectors of the inner corners of the edge modules; (**b**) Face corner angles ϕ_{jk} and edge module variables ψ_{jk} and W_{jk}

$$= \sum_{l=1}^{k} \left(\mathbf{R}_3 \left(\sum_{m=1}^{l-1} \left(\frac{\psi_{jm}}{2} + \phi_{jm} + \frac{\psi_{j\,m+1}}{2} \right) \right) \times \begin{bmatrix} (d_{21}^{jl} - d_{11}^{jl}) \cos\left(\frac{\psi_{jl}}{2}\right) \\ W_{jl} + (d_{11}^{jl} + d_{21}^{jl} - \|\mathbf{z}^{jl}\|) \sin\left(\frac{\psi_{jl}}{2}\right) \\ 0 \end{bmatrix} \right) + \mathbf{b}^{j0}, \tag{7.6}$$

where $\mathbf{R}_3(\cdot) \in \mathbb{R}^{3\times 3}$ is the matrix representing a rotation about an axis aligned with $\mathbf{e}_3$ and is defined in (2.22). The face corner angles ϕ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, are determined via (3.18).

Consider the mapping of the position vector of a point from the face with corner angle $\phi_{j\,k-1}$ to the face with corner angle ϕ_{jk}. The transformation associated with "crossing" the edge module located between these two faces can be decomposed as the following ordered transformations:

1. Translation by $\mathbf{b}^{jk} - \mathbf{b}^{j\,k-1}$. This translation is formulated in matrix form as $\mathbf{T}(\mathbf{b}^{jk} - \mathbf{b}^{j\,k-1})$, where $\mathbf{T}(\cdot)$ is defined in (2.53)
2. Rotation of $\phi_{jk} + \psi_{jk}$ about an axis aligned to $\mathbf{e}_3$ and intersecting the point with position vector $\mathbf{b}^{jk}$. This rotation is formulated in matrix form as $\mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk} + \psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{jk})$, where $\mathbf{Q}_3(\cdot)$ is defined in (2.57)

The product of the former transformation matrices is given as follows:

$$\begin{aligned} &\mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk} + \psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{jk})\,\mathbf{T}(\mathbf{b}^{jk} - \mathbf{b}^{j\,k-1}) \\ &\quad = \mathbf{T}(\mathbf{b}^{jk})\,\mathbf{Q}_3(\phi_{jk} + \psi_{jk})\,\mathbf{T}^{-1}(\mathbf{b}^{j\,k-1}). \end{aligned} \tag{7.7}$$

The composition of the transformations provided in (7.7) associated with crossing the edge modules with variables W_{jk}, ψ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, must be the identity transformation such that these edge modules, along with the trimmed faces with corner angles ϕ_{jk}, $k = 1, \ldots, n_j^{\mathcal{N}}$, form a closed loop:

$$\begin{aligned} \mathbf{I}_4 &= \prod_{k=1}^{n_j^{\mathcal{N}}} \mathbf{T}(\mathbf{b}^{jk})\, \mathbf{Q}_3(\phi_{jk} + \psi_{jk})\, \mathbf{T}^{-1}(\mathbf{b}^{j\,k-1}) \\ &= \mathbf{T}(\mathbf{b}^{jn_j^{\mathcal{N}}})\, \mathbf{Q}_3 \left(\sum_{k=1}^{n_j^{\mathcal{N}}} (\phi_{jk} + \psi_{jk}) \right) \mathbf{T}^{-1}(\mathbf{b}^{j0}). \end{aligned} \tag{7.8}$$

The condition (7.8) holds if $\sum_{k=1}^{n_j^{\mathcal{N}}} (\phi_{jk} + \psi_{jk}) = 2\pi n$ with $n \in \mathbb{Z}$.[4] Specifically, n must be equal to 1 for the closed strip to be developable [7, 8]:

$$\boxed{2\pi = \sum_{k=1}^{n_j^{\mathcal{N}}} \phi_{jk} + \sum_{k=1}^{n_j^{\mathcal{N}}} \psi_{jk},} \tag{7.9}$$

In addition to (7.9), it is required that $\mathbf{b}^{jn_j^{\mathcal{N}}} = \mathbf{b}^{j0}$ for (7.8) to hold. Considering (7.6), $\mathbf{b}^{jn_j^{\mathcal{N}}} = \mathbf{b}^{j0}$ implies the following constraint:

$$\boxed{\begin{aligned} \mathbf{0}_3 = \sum_{k=1}^{n_j^{\mathcal{N}}} \Bigg(& \mathbf{R}_3 \left(\sum_{l=1}^{k-1} \left(\tfrac{\psi_{jl}}{2} + \phi_{jl} + \tfrac{\psi_{j\,l+1}}{2} \right) \right) \\ & \times \begin{bmatrix} (d_{21}^{jk} - d_{11}^{jk}) \cos\left(\tfrac{\psi_{jk}}{2}\right) \\ W_{jk} + (d_{11}^{jk} + d_{21}^{jk} - \|\mathbf{z}^{jk}\|) \sin\left(\tfrac{\psi_{jk}}{2}\right) \\ 0 \end{bmatrix} \Bigg), \end{aligned}} \tag{7.10}$$

cf. (4.7). The vector constraint (7.10) provides two scalar constraints because the third component of such an equation is always equal to 0. Therefore, the *loop closure constraints* (7.9) and (7.10) provide a total of $3N_{\mathcal{N}}^I$ equality constraints.

[4] The set of the integer numbers is denoted as $\mathbb{Z}$.

7.2.3.2 Constraints for Valid Edge Module Geometry

In addition to the loop closure constraints provided in (7.9)–(7.10), other constraints must be imposed to ensure that the geometry of each individual edge module is valid. The interior fold (having fold width $\hat{w}_i^{\mathsf{I0}}$; see Fig. 7.5a) and the faces of an edge module degenerate to straight line segments if $\hat{\psi}_i$ reaches $\pm\pi$ (refer to Fig. 7.5a). Therefore, the following bounds must be imposed for this variable:

$$\boxed{-\pi < \hat{\psi}_i < \pi,} \tag{7.11}$$

where $i = 1, \ldots, N_{\mathcal{E}}^I$. Furthermore, the exterior folds (each having fold width $\hat{w}_i^{\mathsf{E0}}$; see Fig. 7.5a) of an edge module must not overlap with the interior fold. This requirement yields the following constraint (refer to Fig. 7.5a, b):

$$\boxed{\begin{aligned}\hat{W}_i \geq \hat{w}_i^{\mathsf{I0}} + 2\hat{w}_i^{\mathsf{E0}} \cos\left(\tfrac{\hat{\psi}_i}{2}\right) + \|\hat{\boldsymbol{z}}^i\| \sin\left(\left|\tfrac{\hat{\psi}_i}{2}\right|\right) \\ - \begin{cases} (\hat{d}_{11}^i + \hat{d}_{21}^i) \sin\left(\tfrac{\hat{\psi}_i}{2}\right); & \hat{\psi}_i \geq 0 \\ (\hat{d}_{12}^i + \hat{d}_{22}^i) \sin\left(-\tfrac{\hat{\psi}_i}{2}\right); & \hat{\psi}_i < 0 \end{cases}.\end{aligned}} \tag{7.12}$$

where $i = 1, \ldots, N_{\mathcal{E}}^I$; cf. (4.12). Equation (7.12) and the upper and lower bounds of $\hat{\psi}_i$ in (7.11) provide $3N_{\mathcal{E}}^I$ inequality constraints.

7.2.3.3 Constraints to Prevent Intersections Among Tuck-Folded Edge Modules

Intersections between regions of the sheet are not allowed in valid configurations (see Sect. 5.2) and hence must be avoided when adjacent edge modules are tuck-folded in the goal configuration $\mathcal{S}_\star$ (see Fig. 7.3e). For this purpose, certain regions of the edge modules are *trimmed* as shown in Fig. 7.5c such that each edge module does not intersect any of its neighboring edge modules in $\mathcal{S}_\star$. This procedure is called *edge module trimming*.

Figure 7.7 shows the importance of edge module trimming. If the edge module trimming procedure is not considered and the edge module geometry is as given in Fig. 7.5b, adjacent edge modules intersect at the goal configuration $\mathcal{S}_\star$ as observed in Fig. 7.7d. However, if the edge module trimming procedure is considered and the geometry of the edge modules is as illustrated in Fig. 7.5c, no intersections among adjacent edge modules occur as shown in Fig. 7.7g. The *trim angles* $\hat{\tau}_1^i$ and $\hat{\tau}_2^i$, $i =$

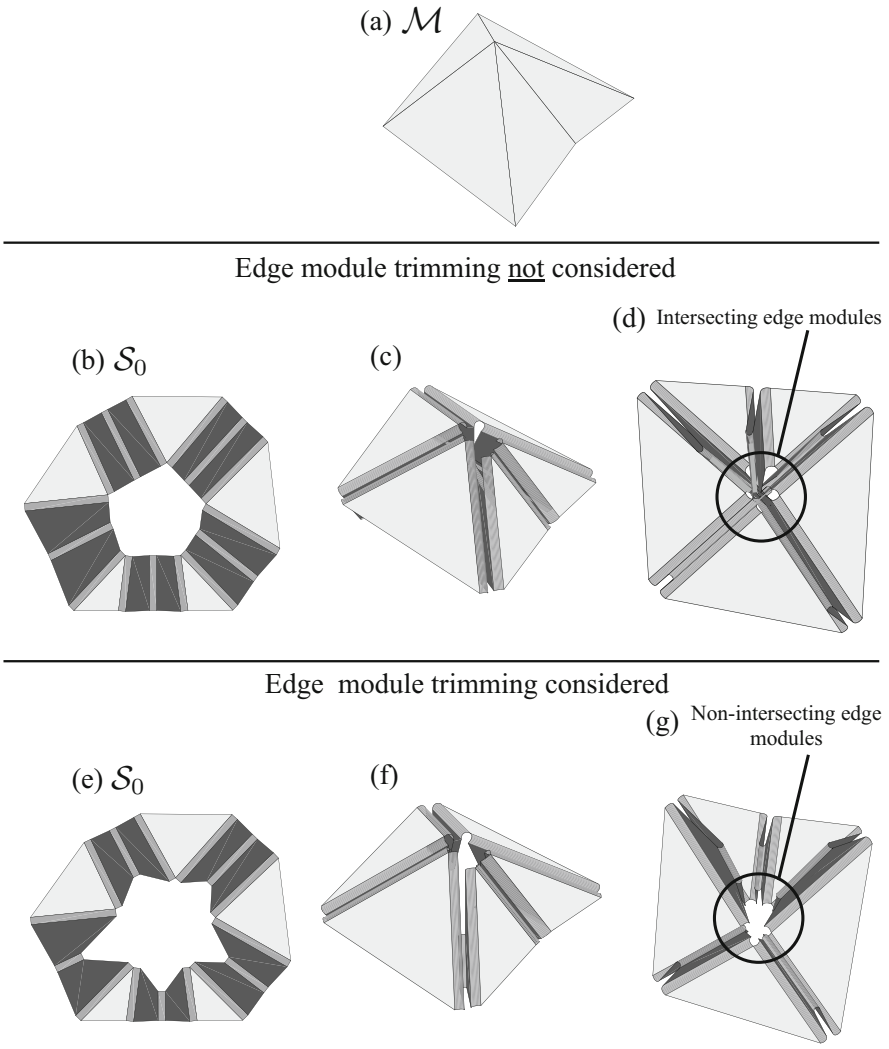


Fig. 7.7 Example showing the need for *edge module trimming*. (**a**) A simple goal mesh $\mathcal{M}$. (**b**) and (**e**) Determined sheet designs $\mathcal{S}_0$. Edge module trimming is not considered for the sheet design shown in (**b**) while such a procedure is considered for the design in (**e**). (**c**)–(**d**), (**f**)–(**g**) Exterior and interior views of the goal configuration $\mathcal{S}_\star$ for both cases. It is observed in (**d**) that if the edge module trimming procedure described in Sect. 4.2.4 is not considered, the tuck-folded edge modules intersect at $\mathcal{S}_\star$. If such a procedure is considered, adjacent edge modules do not intersect in $\mathcal{S}_\star$ as shown in (**g**)

$1, \ldots, N_{\mathcal{E}}^I$, are introduced to define the trimmed regions of the ith edge module as shown in Fig. 7.5c. The procedure used to compute these angles is presented in Sect. 4.2.4.

Each edge module *must remain connected after edge module trimming*. This requirement is satisfied if the following inequality holds for each edge module (refer to Fig. 7.5c)[5]:

$$\begin{aligned} \|\hat{\mathbf{z}}^i\| \cos\left(\tfrac{\hat{\psi}_i}{2}\right) > &\left(\tfrac{\hat{W}_i}{2} - \tfrac{\|\hat{\mathbf{z}}^i\|}{2} \sin\left(\tfrac{\hat{\psi}_i}{2}\right)\right) \tan\left(\langle \hat{\tau}_1^i + \tfrac{\hat{\psi}_i}{2} \rangle\right) \\ &+ \left(\tfrac{\hat{W}_i}{2} + \tfrac{\|\hat{\mathbf{z}}^i\|}{2} \sin\left(\tfrac{\hat{\psi}_i}{2}\right)\right) \tan\left(\langle \hat{\tau}_2^i - \tfrac{\hat{\psi}_i}{2} \rangle\right) \\ &+ \left(\max(\hat{d}_{11}^i, \hat{d}_{21}^i) + \max(\hat{d}_{12}^i, \hat{d}_{22}^i)\right) \cos\left(\tfrac{\hat{\psi}_i}{2}\right). \end{aligned} \tag{7.13}$$

where $i = 1, \ldots, N_{\mathcal{E}}^I$; cf. (4.13). The preceding equation provides $N_{\mathcal{E}}^I$ additional inequality constraints. It is noted that if $(\hat{\tau}_1^i + \hat{\psi}_i/2) < 0$ or $(\hat{\tau}_2^i - \hat{\psi}_i/2) < 0$, these angles are set to 0 to prevent any increase in area of the edge modules during this procedure.

7.2.3.4 Summary of Design Constraints

The proposed design method introduces $2N_{\mathcal{E}}^I$ *design variables* corresponding to $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$. The loop closure constraints (7.9) and (7.10) provide $3N_{\mathcal{N}}^I$ *equality constraints*. The constraints (7.11)–(7.13) allowing for valid edge module geometries and self-intersection avoidance in $\mathcal{S}_\star$ provide $4N_{\mathcal{E}}^I$ *inequality constraints*. We utilize the numerical procedure outlined in Sect. 4.2.5 to solve for the design variables that satisfy the derived constraints. Furthermore, the determination of a history of folding motion from the computed reference configuration of the origami sheet $\mathcal{S}_0$ towards the *goal configuration* $\mathcal{S}_\star$ that approximates $\mathcal{M}$ is performed as described in Sect. 4.2.6.

7.3 Examples of the Tuck-Folding Method

In this section, we provide representative examples of the tuck-folding method for origami with smooth folds described in Sect. 7.2. We implemented this method in MATLAB®. The faces of the goal meshes and the edge modules are visualized

[5]The Macaulay brackets are denoted as $\langle \cdot \rangle$ and defined as: $\langle y \rangle = \begin{cases} y; & y \geq 0 \\ 0; & y < 0 \end{cases}$.

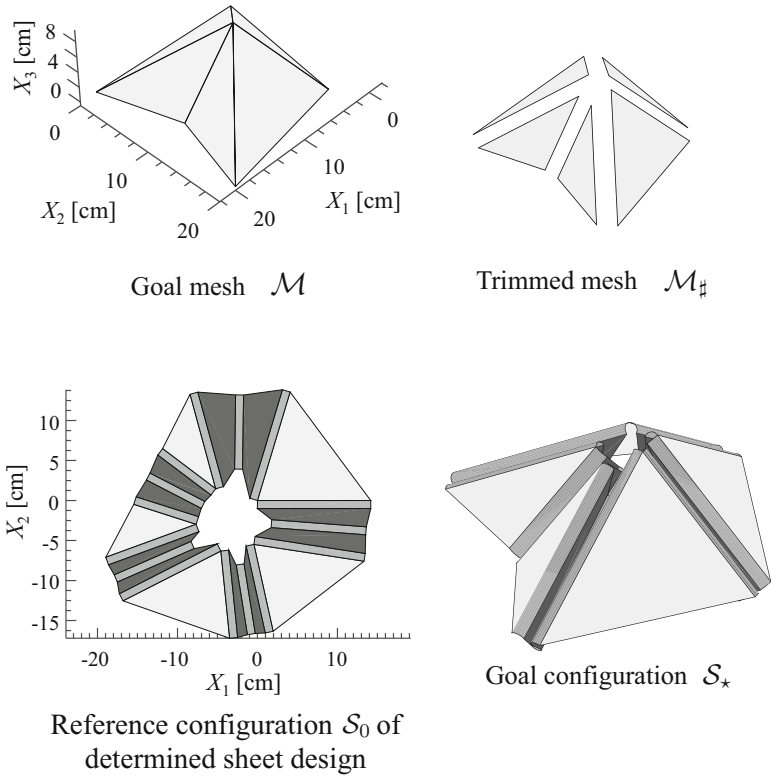


Fig. 7.8 Goal mesh $\mathcal{M}$ and associated trimmed mesh $\mathcal{M}_\sharp$, sheet design $\mathcal{S}_0$, and folded goal configuration $\mathcal{S}_\star$ obtained using the tuck-folding method for origami with smooth folds. A physical prototype of this sheet design was fabricated using a polyjet additive manufacturing approach and is shown in Fig. 7.9

as filled three-dimensional polygons using the MATLAB command `fill3` while the smooth folds are visualized as three-dimensional surfaces using the MATLAB command `surf`. The complete set of MATLAB scripts used to generate the examples is included in the Supplemental Materials for this chapter and described in Appendix B.6. We use the MATLAB implementation of the kinematic model for origami with smooth folds presented in Chap. 5 to simulate the folding motion of the designed origami sheets.

The first example considers a goal mesh $\mathcal{M}$ with one interior vertex and is illustrated in Fig. 7.8. The associated trimmed mesh $\mathcal{M}_\sharp$, determined sheet design $\mathcal{S}_0$, and folded goal configuration $\mathcal{S}_\star$ computed using the tuck-folding method for origami with smooth folds are also shown in Fig. 7.8. We provide an experimental demonstration of the present design method for qualitative comparisons between a simulated structure and its associated prototype. Photos of the fabricated prototype are shown in Fig. 7.9. The prototype was fabricated using a polyjet additive

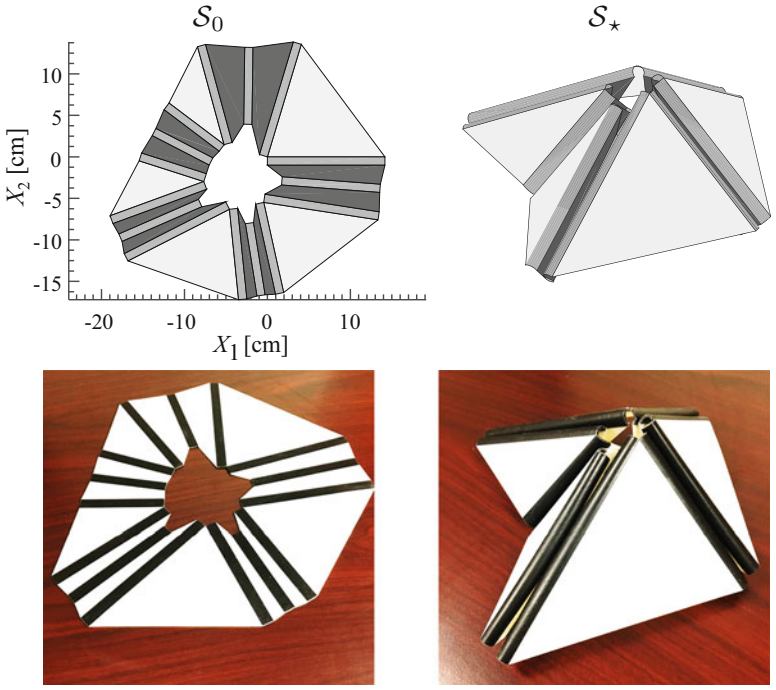


Fig. 7.9 Comparison between computational and experimental models in the reference planar configuration (left) and the goal configuration (right). *Credit: Cullen Nauck*

manufacturing approach. The faces of the prototype (white regions) are comprised of VeroWhite material with Young's modulus of approximately 1.5 GPa (1500 MPa) and the smooth folds (black regions) are comprised of TangoBlack material with Young's modulus of approximately 0.34 MPa. The large discrepancy between the values of the Young's modulus of the faces and the smooth folds ensures that our assumptions of rigid faces and flexible smooth folds are applicable for the prototype. A comparison between the computational and the experimental models in the reference planar configuration $\mathcal{S}_0$ and the goal configuration $\mathcal{S}_\star$ is provided in Fig. 7.9. The prototype is manually folded and the edge modules are held in their tuck-folded configuration using clips.

A sheet design obtained using the tuck-folding method and the corresponding goal configuration $\mathcal{S}_\star$ for a goal mesh $\mathcal{M}$ of constant positive Gaussian curvature [8, 9] are shown in Fig. 7.10. A goal mesh associated with a shape of constant negative Gaussian curvature is considered in Fig. 7.11. It is interesting to contrast the similarities in $\mathcal{S}_0$ between Figs. 7.10 and 7.11 with the clear differences in the associated goal configurations $\mathcal{S}_\star$.

Variations of the goal mesh shown in Fig. 7.10 and their corresponding sheet designs are shown in Fig. 7.12. It is verified that each sheet in Fig. 7.12 can be

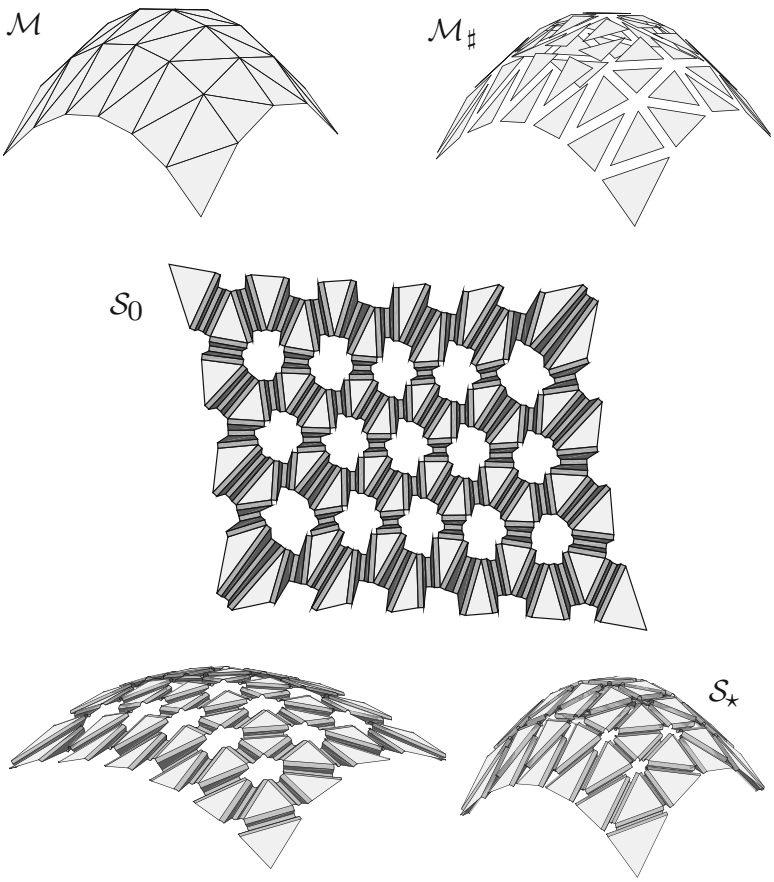


Fig. 7.10 Goal mesh $\mathcal{M}$ representing a shape of constant positive Gaussian curvature and its associated trimmed mesh $\mathcal{M}_\sharp$, determined sheet design $\mathcal{S}_0$, and folding motion towards the goal configuration $\mathcal{S}_\star$

folded towards their goal configuration $\mathcal{S}_\star$ that approximates $\mathcal{M}$, although the goal configurations are not shown here for the sake of brevity.

To illustrate the multiplicity of possible design solutions, three sheet designs associated with a single goal mesh are shown in Fig. 7.13. These different design solutions are obtained by considering different initial guesses for the sheet design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, in the iterative solution procedure described in Sect. 4.2.5. Values of surface area efficiency E, defined in (4.35), for the determined sheet designs are also indicated in Fig. 7.13.

A more complex example of a torus goal shape is presented in Fig. 7.14. Two different discretizations of the goal shape are provided in Fig. 7.14a. The design method successfully generates sheet designs for both discretizations as shown in Fig. 7.14b–c. Figures 7.15 and 7.16 show the successful design and simulation

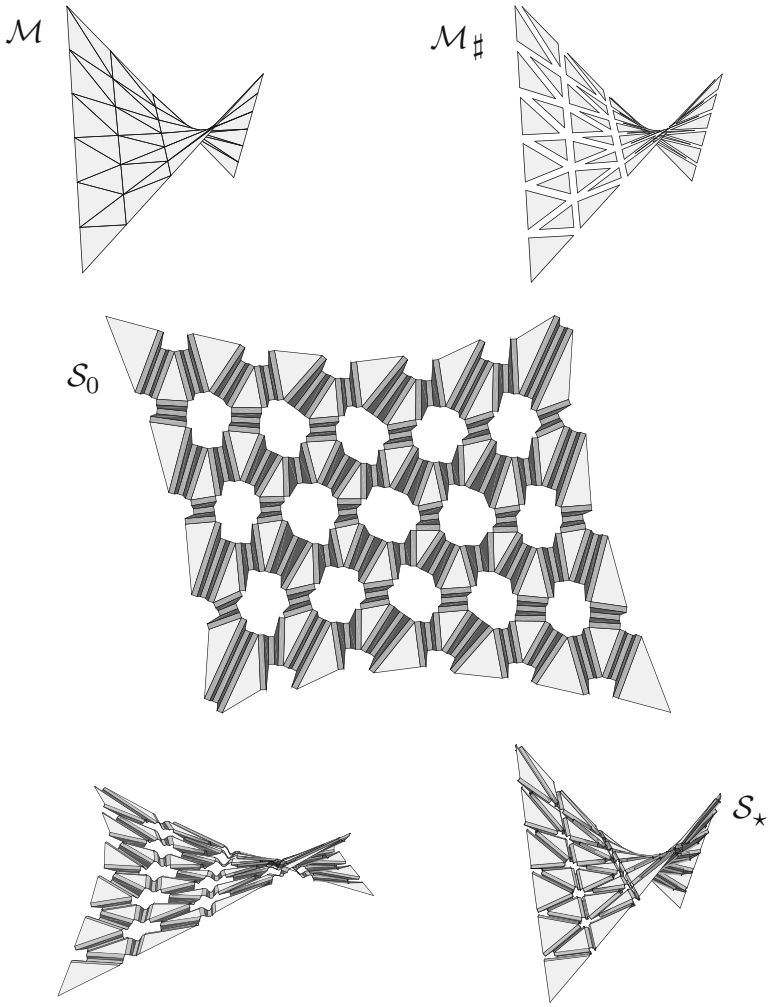


Fig. 7.11 Goal mesh $\mathcal{M}$ representing a shape of constant negative Gaussian curvature and its associated trimmed mesh $\mathcal{M}_\sharp$, determined sheet design $\mathcal{S}_0$, and folding motion towards the goal configuration $\mathcal{S}_\star$

results for goal meshes associated with a sinusoidal tessellation and an autonomous underwater vehicle, respectively.

The proposed design method is not limited to triangulated meshes as the only condition on the shape of the faces in $\mathcal{M}$ is that they must be convex (refer to (4.34) and its associated discussion). Figure 7.17 shows an example of a sheet design and folding motion associated with a goal mesh comprised of quadrilaterals while Fig. 7.18 shows results for a goal mesh comprised of octagons and triangles.

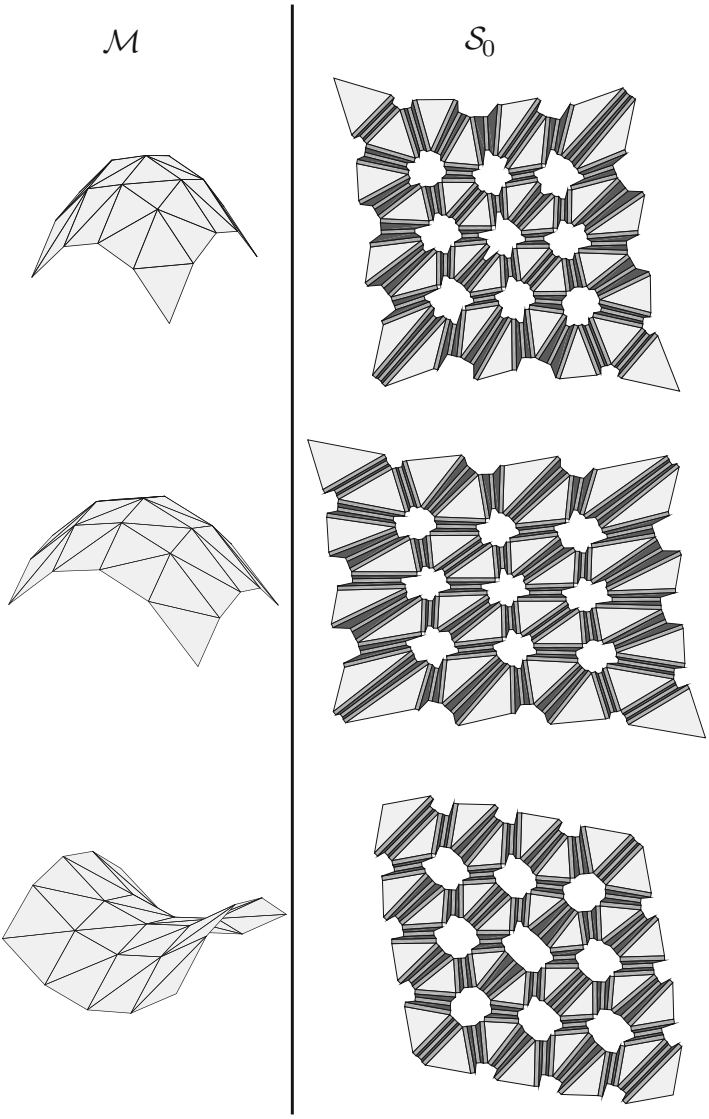


Fig. 7.12 Different goal meshes and their corresponding sheet designs determined using the tuck-folding method for origami with smooth folds

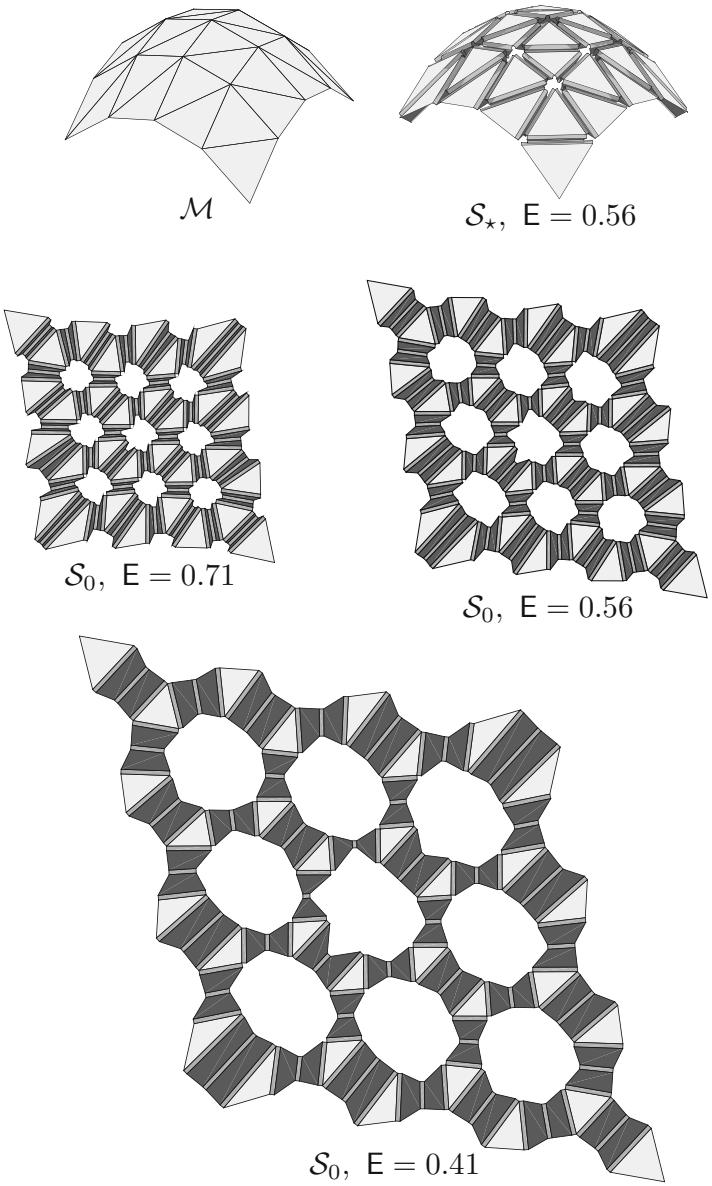


Fig. 7.13 A goal mesh and three sheet designs with different values of surface area efficiency E. The goal configuration $\mathcal{S}_\star$ associated with the sheet design with $\mathsf{E} = 0.56$ is also shown

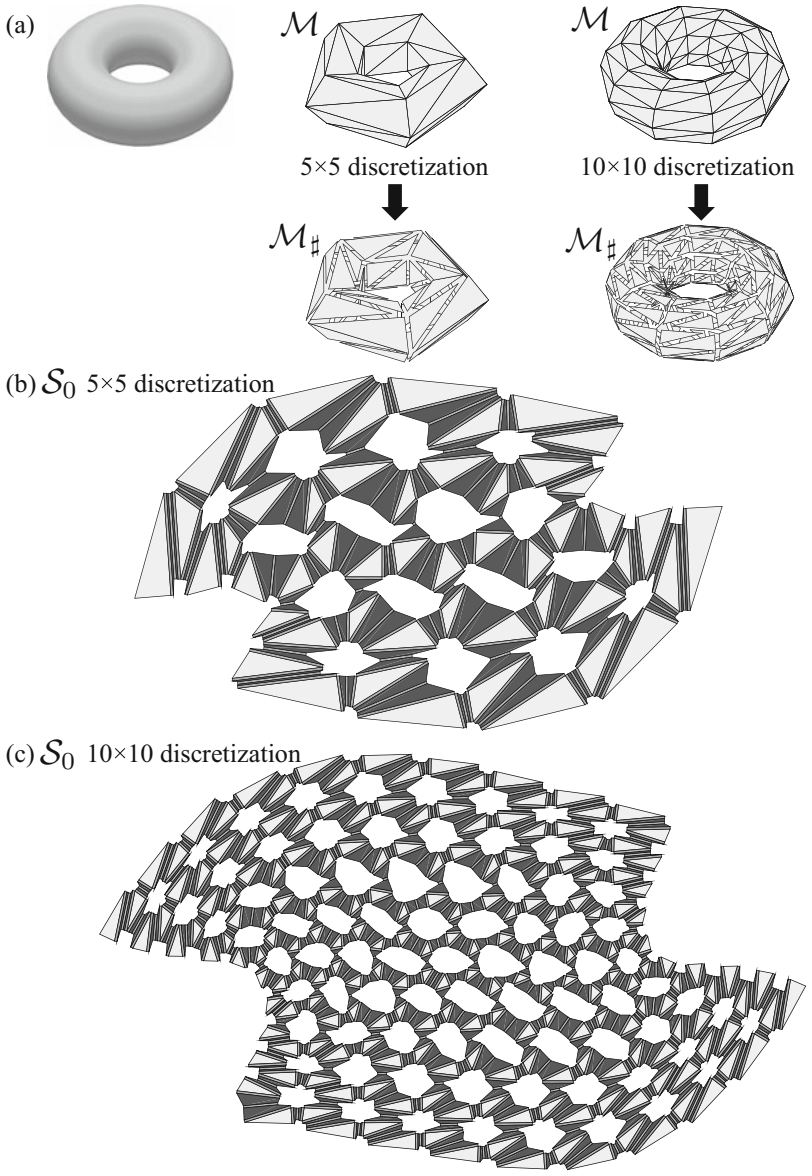


Fig. 7.14 (**a**) Two mesh discretizations of a torus. (**b**) Sheet design for the 5×5 discretization. (**c**) Sheet design for the 10×10 discretization

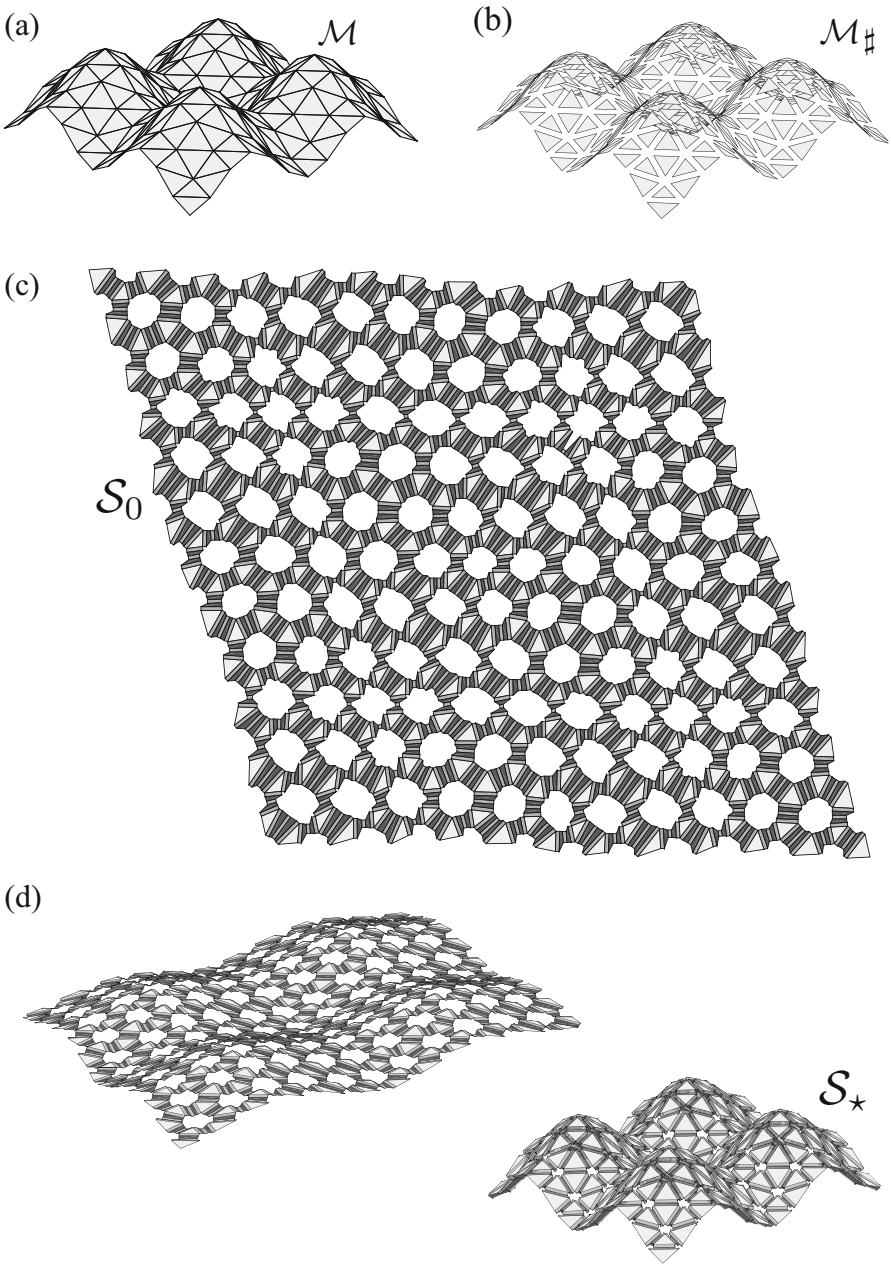


Fig. 7.15 (**a**) Goal mesh $\mathcal{M}$ of a sinusoidal tessellation. (**b**) Trimmed mesh $\mathcal{M}_\sharp$. (**c**) Determined sheet design $\mathcal{S}_0$. (**d**) Folding motion towards the goal configuration $\mathcal{S}_\star$

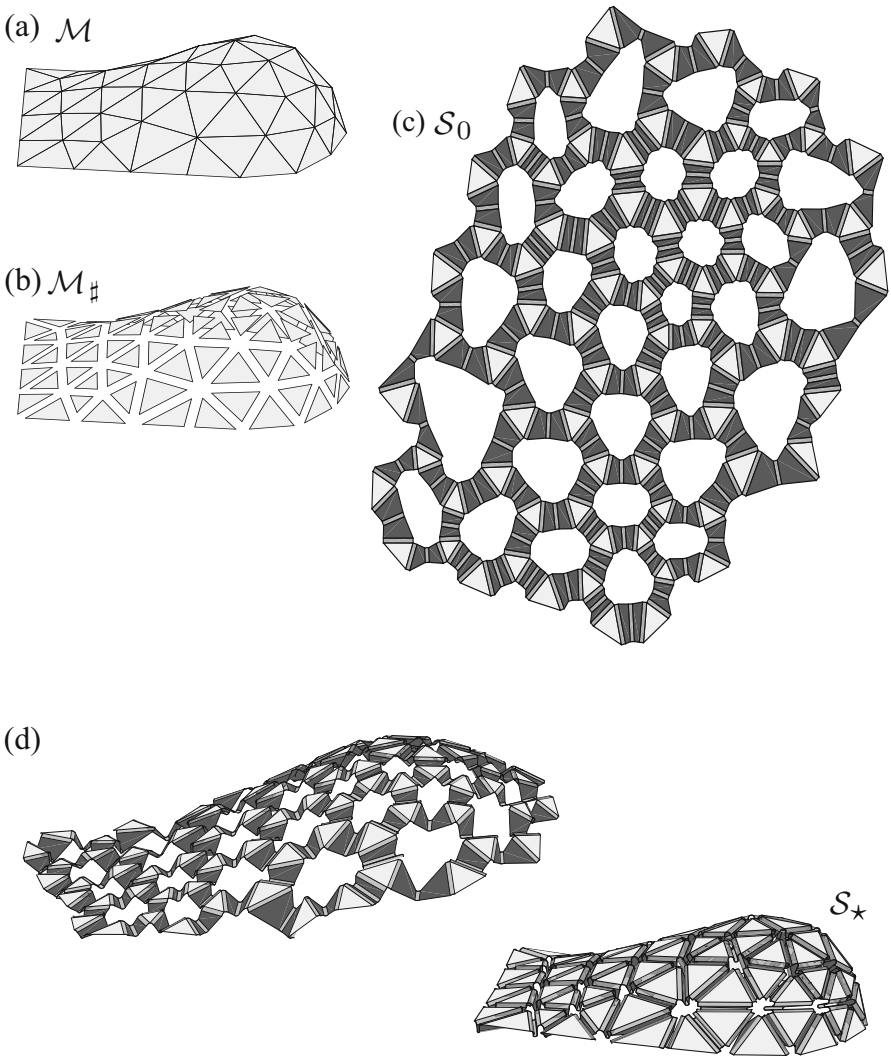


Fig. 7.16 (**a**) Goal mesh $\mathcal{M}$ of the upper surface of an autonomous underwater vehicle [10]. *Credit: Doe Young Hur*. (**b**) Trimmed mesh $\mathcal{M}_\sharp$. (**c**) Determined sheet design $\mathcal{S}_0$. (**d**) Folding motion towards the goal configuration $\mathcal{S}_\star$

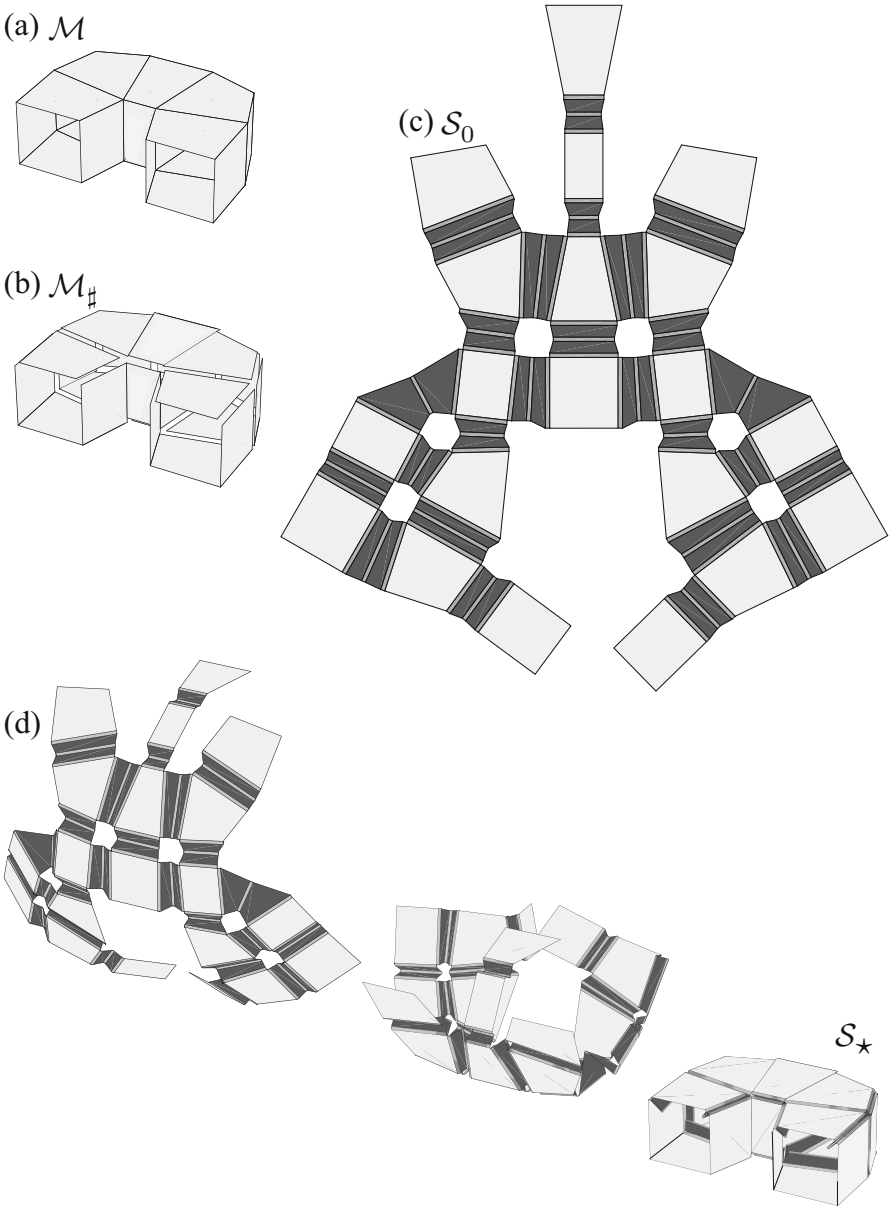


Fig. 7.17 (**a**) Goal mesh $\mathcal{M}$ comprised of quadrilaterals. (**b**) Trimmed mesh $\mathcal{M}_\sharp$. (**c**) Determined sheet design $\mathcal{S}_0$. (**d**) Folding motion towards the goal configuration $\mathcal{S}_\star$

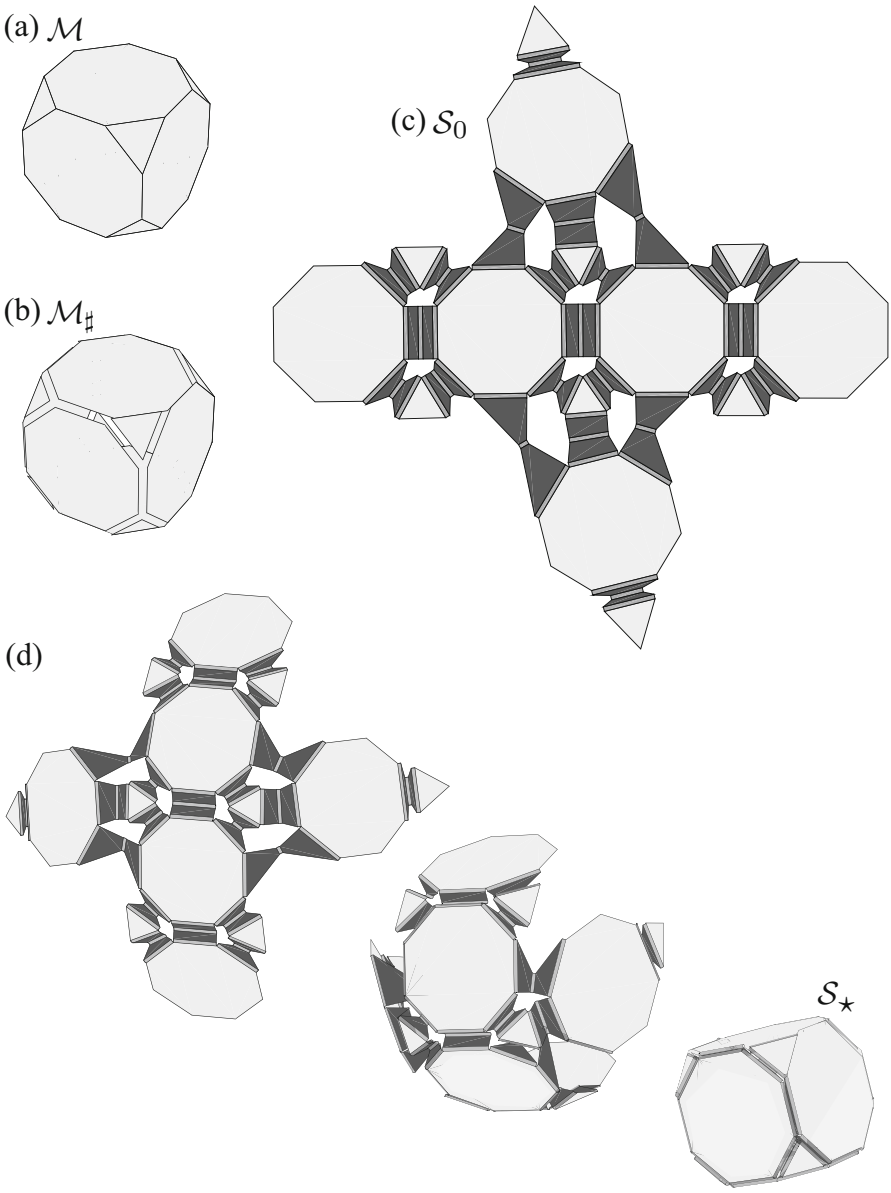


Fig. 7.18 (**a**) Goal mesh $\mathcal{M}$ comprised of octagons and triangles. (**b**) Trimmed mesh $\mathcal{M}_\sharp$. (**c**) Determined sheet design $\mathcal{S}_0$. (**d**) Folding motion towards the goal configuration $\mathcal{S}_\star$

7.3.1 *Design and Fabrication of Shape Memory Polymer Self-Folding Sheets*

In this section, we apply the tuck-folding method derived in this chapter to the design and fabrication of SMP-based self-folding sheets.[6] The goal is to synthesize SMP-based self-folding sheets that morph towards arbitrary three-dimensional goal shapes starting from a planar configuration.

7.3.1.1 Fabrication of Shape Memory Polymer Active Folds

Our method for the fabrication of SMP-based self-folding sheets is inspired by the techniques developed in [11]. The shape memory composite comprising the sheets consists of three layers: An SMP layer in the middle bonded to stiff passive layers on its top and bottom sides. We use flexible polystyrene plastic sheets, commercially known as *Shrinky Dinks*, for the SMP layer and paperboard for the passive layers. These layers are bonded together with a silicone-based adhesive. A schematic of the cross-section of the laminated folds is shown in Fig. 7.19, where $\hat{w}_i^0$ is the width of the folds, t_{paper} is the thickness of the paperboard layers, t_{SMP} is the thickness of the SMP layers, δ is the length between the SMP layer and the axis of rotation during folding, and $\hat{\theta}_i^\star$ is the goal fold angle. The paperboard layer in the folding direction is removed for each fold as illustrated in Fig. 7.19. Such a laminate configuration is obtained from [11]. In all the results shown in this section, $t_{\text{paper}} = 0.508\,\text{mm}$, $t_{\text{SMP}} = 0.5\,\text{mm}$, and $\delta = 0.2\,\text{mm}$.

Unlike the examples presented in Sect. 7.3, the width $\hat{w}_i^0$ of each fold in the sheets considered here is *not* arbitrary. We geometrically determine an expression for the fold width $\hat{w}_i^0$ as a function of the fold angle $\hat{\theta}_i^\star$ achievable by the SMP smooth folds illustrated in Fig. 7.19. Such an expression is given as follows (refer to Fig. 7.19c):

$$\hat{w}_i^0 = 2(t_{\text{paper}} + t_{\text{SMP}} + \delta)\tan\left(\frac{\hat{\theta}_i^\star}{2}\right). \tag{7.14}$$

Figure 7.20 shows a comparison between the results obtained using (7.14) and experimental tests for single folds.

We experimentally determined that the laminate layup for the smooth folds shown in Fig. 7.19 does not achieve fold angles of 180° (π radians), which are necessary in the tuck-folding method addressed in this chapter (refer to Fig. 7.4). For those special cases, (7.14) is not applicable. To achieve fold angles of 180°, we fabricated smooth folds for which one of the paperboard layers of the composite had a 6 mm gap width and the opposite paperboard layer had five evenly separated

[6]The authors would like to gratefully acknowledge Milton Garza for performing the experimental work presented in this section.

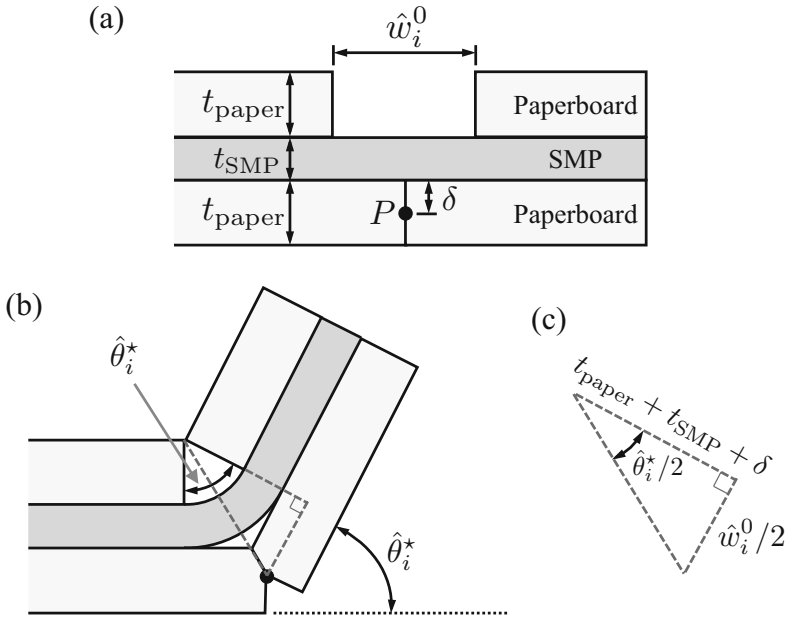


Fig. 7.19 Cross-section of an SMP-based active fold. The laminated composite consists of an SMP middle layer and two paperboard outer layers. (**a**) Reference configuration. When the SMP layer is thermally actuated, it shrinks and causes the laminate to fold in the direction of the side with a gap about the axis P. (**b**) Folding continues until the two sides of the layer with a gap come into contact, resulting in a goal fold angle $\hat{\theta}_i^\star$. (**c**) Schematic used to derive (7.14)

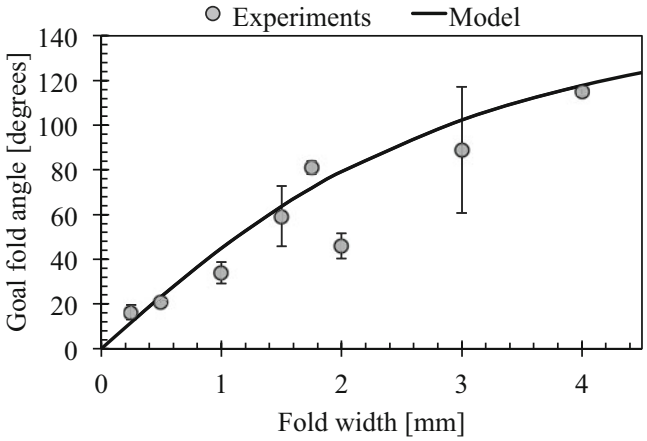


Fig. 7.20 Goal fold angle $\hat{\theta}_i^\star$ vs. fold width $\hat{w}_i^0$ for the considered SMP-based active folds. Results from experimental tests are compared with those obtained via (7.14). Error bars indicate the standard deviation of fold angles at each fold width tested

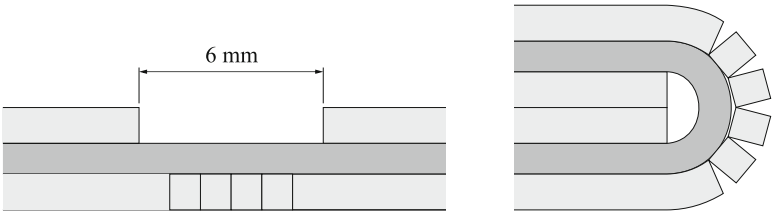


Fig. 7.21 Additional cuts in the bottom paperboard layer made in order to achieve fold angles of 180°. Left: Reference configuration. Right: Final configuration with fold angle of 180°

cuts within the range of the gap as illustrated in Fig. 7.21. This fold design allows for lower bending stiffness in the composite, ultimately allowing the composite to achieve 180° fold angles.

The composite sheets were fabricated using a laser cutting machine with their dimensions according to their computational design obtained via the tuck-folding method. To prevent the SMP layer from actuating during fabrication, low power and speed settings are used in the laser cutting machine.

7.3.1.2 Self-Folding Behavior of Shape Memory Polymer Sheets

Figures 7.22 and 7.23 show origami sheet designs obtained via the tuck-folding method for a saddle goal shape and a pyramid goal shape, respectively. A fabricated SMP-based composite sheet associated with the saddle goal shape of Fig. 7.22 is shown in Fig. 7.24a. The three layers of the composite are individually shown in Fig. 7.25. The fabricated composite sheet was inserted into a thermal chamber for uniform heating from room temperature up to a temperature of 130 °C (403 K), which allowed the SMP to actuate and the sheet to fold towards its goal configuration shown in Fig. 7.24b. A fabricated composite sheet associated with the pyramid goal shape of Fig. 7.23 is shown in Fig. 7.26a and its corresponding final configuration after thermally-driven actuation is shown in Fig. 7.26b. For both examples, the tuck-folding method for origami with smooth folds successfully generated the geometry of the SMP sheet designs such that they achieve the goal configurations that approximate their goal meshes.

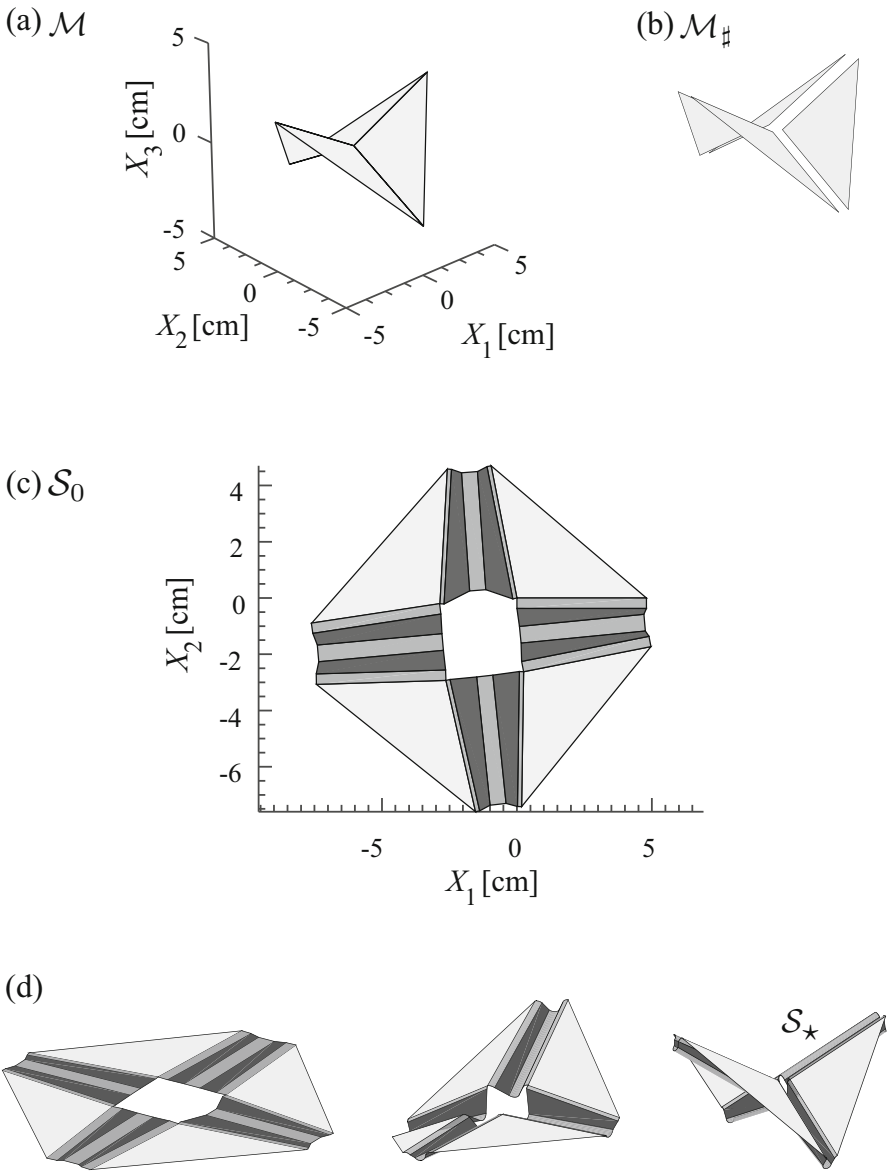


Fig. 7.22 Origami sheet design of an SMP self-folding structure for a goal mesh with one interior node of negative Gaussian curvature: (**a**) Goal mesh; (**b**) Trimmed mesh; (**c**) Determined sheet design; (**d**) Configurations attained by the structure during its folding history (kinematic simulation)

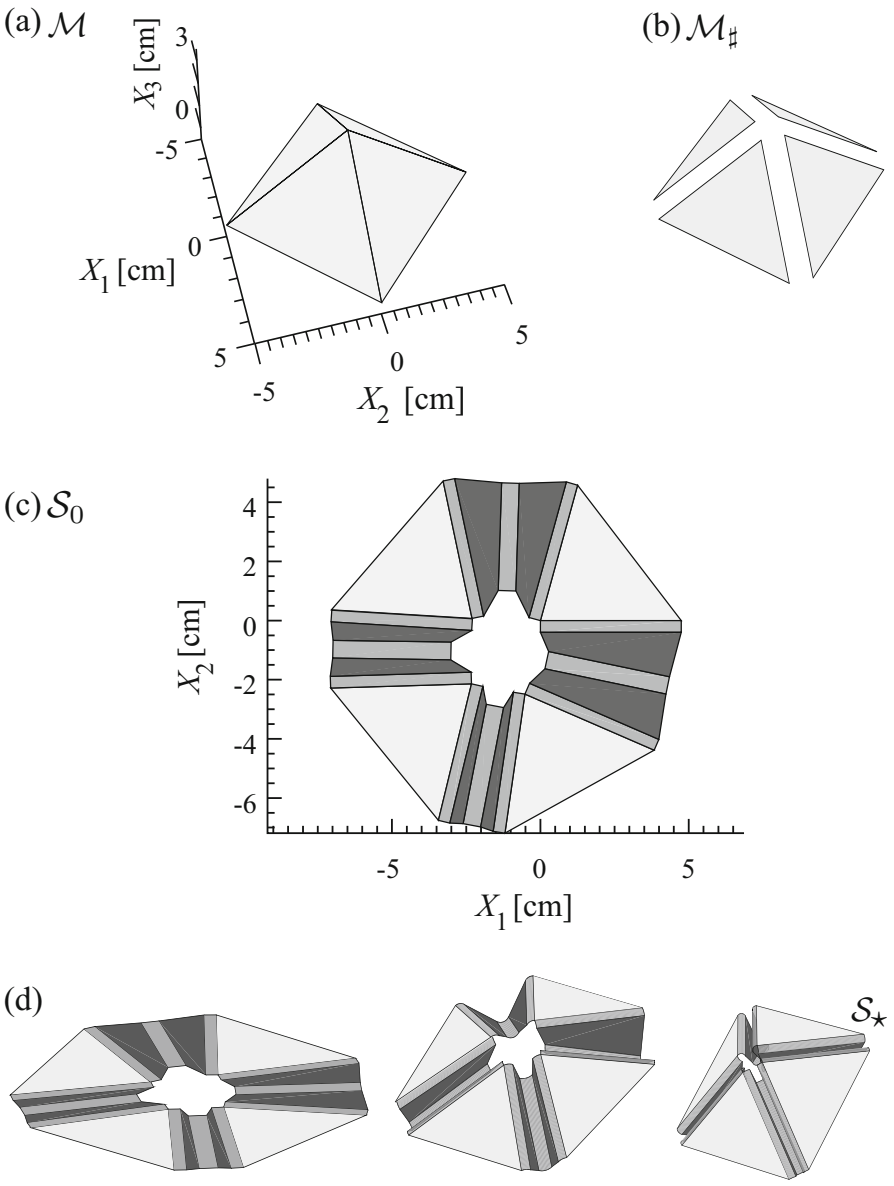


Fig. 7.23 Origami sheet design of an SMP self-folding structure for a goal mesh with one interior node of positive Gaussian curvature: (**a**) Goal mesh; (**b**) Trimmed mesh; (**c**) Determined sheet design; (**d**) Configurations attained by the structure during its folding history (kinematic simulation)

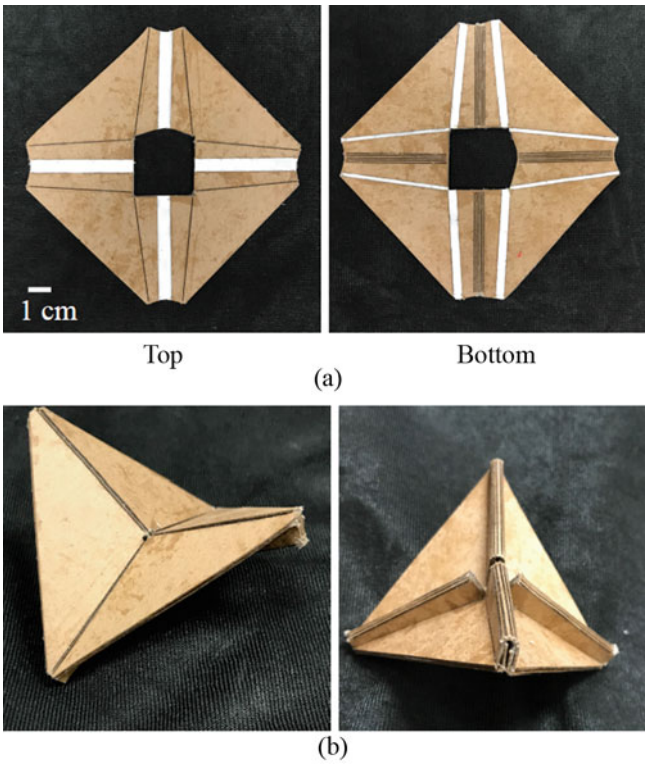


Fig. 7.24 (**a**) Top and bottom views of the self-folding sheet for the saddle goal shape (Fig. 7.22). The white regions in the top view correspond to the valley folds while the white regions in the bottom view correspond to the mountain folds. (**b**) Views of the self-folding sheet in the goal configuration

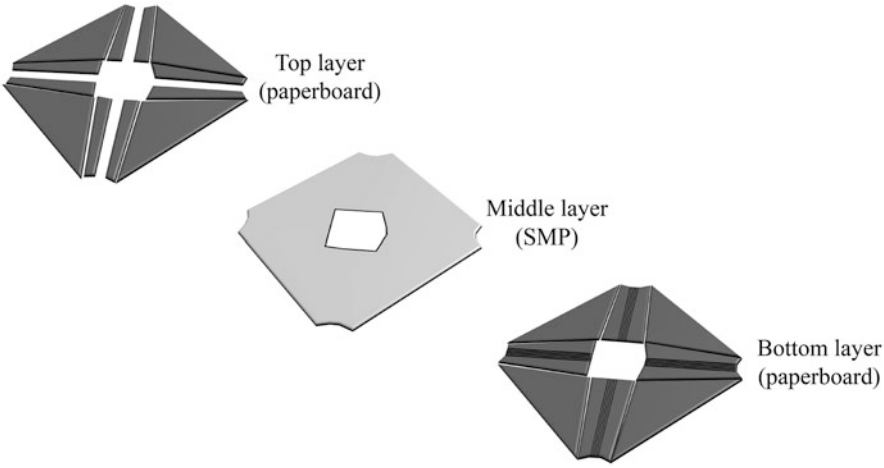


Fig. 7.25 Individual layers of the SMP self-folding sheet for a goal mesh having one interior node of negative Gaussian curvature

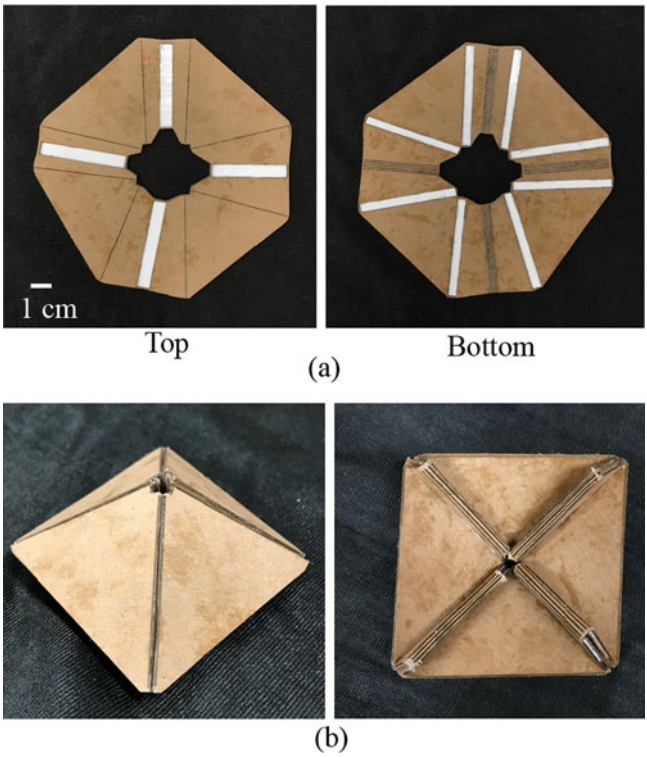


Fig. 7.26 (**a**) Top and bottom views of the self-folding sheet for the pyramid goal shape (Fig. 7.23). The white regions in the top view correspond to the valley folds while the white regions in the bottom view correspond to the mountain folds. (**b**) Views of the self-folding sheet in the goal configuration

Chapter Summary

A method for the design of origami structures with smooth folds was presented in this chapter. The method is based on the known idea of using folds to create flaps that are tuck-folded to morph an initially planar sheet into an arbitrary three-dimensional goal shape represented as a polygonal mesh [4, 5]. The method solves the origami design problem of determining the shape of a planar sheet including a pattern of smooth folds that allows for the approximation of the goal mesh through such a tuck-based folding. The present design method successfully provides fold patterns that can be realized with various engineering materials (e.g., metals, glassy polymers, active materials) due to the consideration of arbitrary order of continuity G^n in the smooth folds, as opposed to G^0 creased folds considered in most origami design methods from the literature.

For origami design problems in which the goal shape is not given as a polyhedral surface but rather as a *smooth surface*, the process of determining a mesh

discretization for such a surface before the utilization of the method presented in this chapter must be addressed. Multiple algorithms for obtaining polygonal mesh discretizations of smooth surfaces are available in the literature [12, 13]. We include an example in Chap. 8 where the problem of determining an optimal mesh for a given application is addressed for a self-folding parabolic antenna.

Section 7.2.1 summarized the problem description and solution approach of the tuck-folding method for origami with smooth folds. The face trimming step that is introduced in the tuck-folding method for origami with smooth folds was addressed in Sect. 7.2.2 We then studied the parameterization of the origami sheet design and the associated constraints on the design variables. Representative examples of the implemented method considering several goal shapes were provided in Sect. 7.3. Lastly, in Sect. 7.3.1, we presented an example where we applied the tuck-folding method to the design and fabrication of SMP self-folding sheets that morph towards arbitrary three-dimensional goal shapes starting from planar configurations.

Problems

7.1 Derive (7.3) using the schematic provided in Fig. 7.4b.

7.2 Draw a schematic such as that provided in Fig. 7.4 considering a concave edge (i.e., an edge having dihedral angle $\hat{\Theta}_i < \pi$). Then, show that (7.3) is also applicable for concave edges.

7.3 Derive the loop closure constraints provided in Sect. 7.2.3 considering vectors $\mathbf{b}^{j1}, \ldots, \mathbf{b}^{jn_j^{\mathcal{N}}}$ (refer to Fig. 7.6a) defined in clockwise order instead of the counterclockwise order assumed in Sect. 7.2.3. Are the resulting constraints equivalent to those provided in (7.9) and (7.10)?

7.4 An alternative formulation of the constraints on the design variables W_i, ψ_i, $i = 1, \ldots, N_{\mathcal{E}}^I$, can be obtained by replacing the inequality constraint (7.12) with the following equality constraint:

$$\hat{W}_i = \hat{w}_i^{\mathsf{I}0} + 2\hat{w}_i^{\mathsf{E}0} \cos\left(\frac{\hat{\psi}_i}{2}\right) + \|\hat{\mathbf{z}}^i\| \sin\left(\left|\frac{\hat{\psi}_i}{2}\right|\right) - \begin{cases} (\hat{d}_{11}^i + \hat{d}_{21}^i) \sin\left(\frac{\hat{\psi}_i}{2}\right); & \hat{\psi}_i \geq 0 \\ (\hat{d}_{12}^i + \hat{d}_{22}^i) \sin\left(-\frac{\hat{\psi}_i}{2}\right); & \hat{\psi}_i < 0 \end{cases} . \tag{7.15}$$

What would be the implications of using the more restrictive constraint (7.15) instead of (7.12)? How many equality and inequality constraints would this alternative formulation have?

Implement this alternative formulation in a computational environment and determine origami sheet designs for two examples from Sect. 7.3 and compare your results with those of Sect. 7.3.

7.5 Propose and implement a procedure to determine the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N_{\mathcal{E}}^I$, by solving the design problem stated in (4.21) where the function $f(\mathbf{D})$ to be minimized is given as follows:

$$f(\mathbf{D}) = \mathsf{E}(\mathbf{D}), \tag{7.16}$$

and the surface area efficiency E is defined in (4.35).

7.6 Since the resulting goal configuration $\mathcal{S}_\star$ in the tuck-folding method for origami with smooth folds contains the trimmed mesh $\mathcal{M}_\sharp$ as a subset instead of the given goal mesh $\mathcal{M}$ (see Figs. 7.1 and 7.2), the approximation error between $\mathcal{S}_\star$ and $\mathcal{M}$ must be quantified. Propose parameter(s) that can be used to quantify such an error.

7.7 Following the approach presented in Sect. 3.2.4, formulate the mapping of the trimmed faces from their position in the trimmed mesh $\mathcal{M}_\sharp$ to their corresponding position in the reference configuration $\mathcal{S}_0$ of an origami sheet designed using the tuck-folding method.

7.8 Implement the tuck-folding method for origami with smooth folds in a computational environment. Then, determine a sheet design for the goal mesh with two interior nodes of positive Gaussian curvature shown in Fig. 7.27. Assume fold widths of 0.002 for all the smooth folds. The position vectors of the nodes are given as follows:

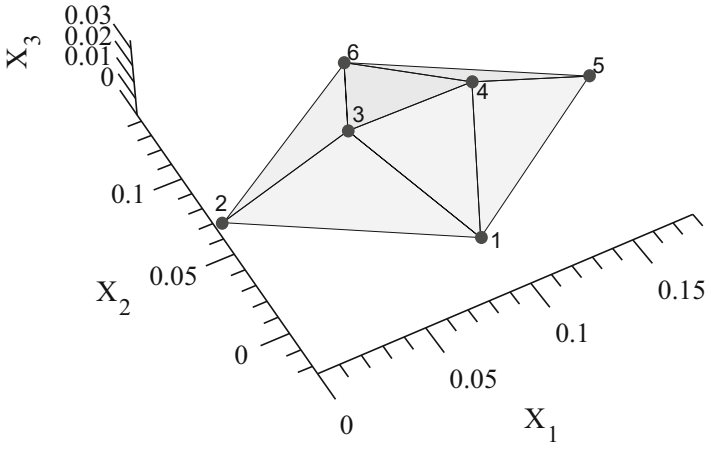


Fig. 7.27 Goal mesh $\mathcal{M}$ for Problem 7.8

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\right]^\top = \begin{bmatrix} 0.09 & 0 & 0 \\ 0 & 0.06 & 0 \\ 0.06 & 0.06 & 0.025 \\ 0.12 & 0.06 & 0.025 \\ 0.18 & 0.06 & 0 \\ 0.09 & 0.12 & 0 \end{bmatrix}, \tag{7.17}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 3 & 2 \\ 1 & 4 & 3 \\ 1 & 5 & 4 \\ 2 & 3 & 6 \\ 3 & 4 & 6 \\ 4 & 5 & 6 \end{bmatrix}. \tag{7.18}$$

7.9 Repeat Problem 7.8 considering the goal mesh with two interior nodes of negative Gaussian curvature shown in Fig. 7.28. The position vectors of the nodes are given as follows:

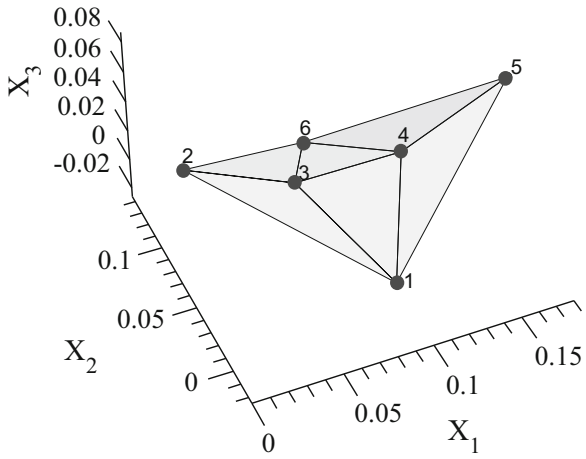
$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\right]^\top = \begin{bmatrix} 0.09 & 0 & 0 \\ 0 & 0.06 & 0.06 \\ 0.06 & 0.06 & 0.03 \\ 0.12 & 0.06 & 0.03 \\ 0.18 & 0.06 & 0.06 \\ 0.09 & 0.12 & 0 \end{bmatrix}, \tag{7.19}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given in (7.18).

7.10 Repeat Problem 7.8 considering the goal mesh with one interior node of negative Gaussian curvature (node 3) and one of positive Gaussian curvature (node 4) shown in Fig. 7.29. The position vectors of the nodes are given as follows:

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\right]^\top = \begin{bmatrix} 0.09 & 0 & 0 \\ 0 & 0.06 & 0.05 \\ 0.06 & 0.06 & 0.025 \\ 0.12 & 0.06 & 0.025 \\ 0.18 & 0.06 & 0 \\ 0.09 & 0.12 & 0 \end{bmatrix}, \tag{7.20}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given in (7.18).



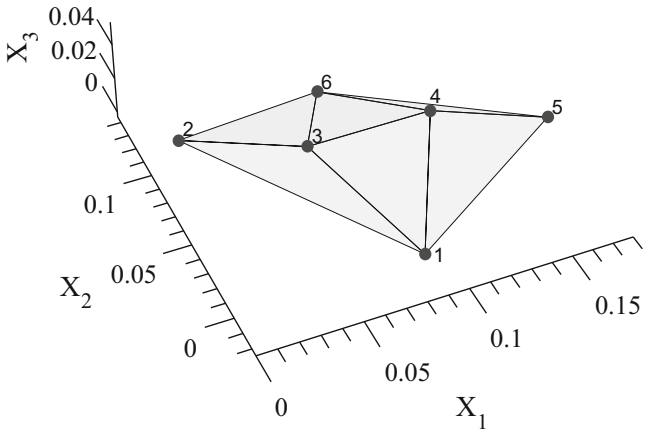


Fig. 7.28 Goal mesh $\mathcal{M}$ for Problem 7.9

Fig. 7.29 Goal mesh $\mathcal{M}$ for Problem 7.10

7.11 Implement the tuck-folding method for origami with smooth folds in a computational environment and determine sheet designs for the 13 Archimedean polyhedra. Compare your results with those of Problem 6.12.

References

1. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr., D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
2. Y. Zhang, F. Zhang, Z. Yan, Q. Ma, X. Li, Y. Huang, J.A. Rogers, Printing, folding and assembly methods for forming 3D mesostructures in advanced materials. Nat. Rev. Mater. **2**, 17019 (2017)

3. M. Johnson, Y. Chen, S. Hovet, S. Xu, B. Wood, H. Ren, J. Tokuda, Z. Tse, Fabricating biomedical origami: a state-of-the-art review. Int. J. Comput. Assist. Radiol. Surg. **12**(11), 2023–2032 (2017)
4. T. Tachi, Origamizing polyhedral surfaces. IEEE Trans. Vis. Comput. Graph. **16**(2), 298–311 (2010)
5. T. Tachi, 3D origami design based on tucking molecule, in *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education* (A K Peters, Natick, 2009), pp. 259–272
6. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Design and simulation of origami structures with smooth folds. Proc. R. Soc. A **473**(2200), 20160716 (2017)
7. J.M. Sullivan, Curvatures of smooth and discrete surfaces, in *Discrete Differential Geometry* (Springer, Berlin, 2008), pp. 175–188
8. E. Akleman, J. Chen, Insight for practical subdivision modeling with discrete gauss-bonnet theorem, in *International Conference on Geometric Modeling and Processing* (Springer, Berlin, 2006) , pp. 287–298
9. C.R. Calladine, *Theory of Shell Structures* (Cambridge University Press, Cambridge, 1989)
10. D.Y. Hur, E. Peraza Hernandez, E. Galvan, D. Hartl, R. Malak, Design optimization of folding solar powered autonomous vehicles using origami architecture, in *Proceedings of the ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2017-67848 (American Society of Mechanical Engineers, New York, 2017)
11. M.T. Tolley, S.M. Felton, S. Miyashita, D. Aukes, D. Rus, R.J. Wood, Self-folding origami: shape memory composites activated by uniform heating. Smart Mater. Struct. **23**(9), 094006 (2014)
12. P.J. Frey, P.L. George, *Mesh Generation: Application to Finite Elements*, 2nd edn. Wiley online library (ISTE/Wiley, London/Hoboken, 2008). ISBN 9780470611166
13. Y. Zheng, R.W. Lewis, D.T. Gethin, Three-dimensional unstructured mesh generation: part 2. Surface meshes. Comput. Methods Appl. Mech. Eng. **134**(3), 269–284 (1996)

Chapter 8
Structural Mechanics and Design of Active Origami Structures

Abstract A model for the mechanics of active origami structures with smooth folds is presented in this chapter. The model entails the integration of the surface kinematics model for origami with smooth folds developed in Chap. 5 and existing plate theories to obtain a structural representation for folds of non-zero thickness. The implementation of the model in a computational environment is also addressed. We provide examples including origami structures comprised of both elastic materials and active materials. Afterwards, the unfolding polyhedra and tuck-folding methods studied in Chaps. 6 and 7 are extended to develop frameworks for the design of *active origami structures*. The extensions account for the folding deformation achievable by smooth folds of specified thickness and constituent materials, which is not considered in the purely kinematic formulations of Chaps. 6 and 7.

8.1 Introduction

As reviewed in Chap. 1, origami has the potential to impact numerous fields in engineering by providing new approaches to the fabrication and morphing of various structures and devices. Modeling the mechanics of origami structures is of special interest in the engineering community [1–3]. It allows for the analysis and design of origami structures taking into account the constitutive behavior of their comprising materials and the influence of the external environment on the resulting folding response.

Most available models for the mechanics of origami structures are limited to the idealization of folds as creases of G^0 continuity, which is not proper for origami structures having non-negligible fold thickness or with maximum curvature at the folds restricted by material limitations (e.g., *active origami structures*) [4–6]. Structural analysis of origami bodies having creased folds requires further idealizations of the mechanical response such as the representation of the folds as torsional springs [1, 2, 7]. On the other hand, although finite element analysis (FEA) allows for high-fidelity physical simulation of origami structures having arbitrary thickness [8–18], the mathematical insights provided in the classical approaches of origami (e.g., geometric constructions [19]) are lost in the generality of FEA.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2_8

Moreover, the kinematic variables associated with FEA models such as node displacements are generally not compatible with those of conventional origami such as fold angles. In view of this, a novel model for the mechanics of origami structures with smooth fold domains of non-zero thickness is developed in this chapter. The presented structural model is developed on the basis of the kinematic model for origami structures with smooth folds described in Chap. 5. The arbitrary order of geometric continuity (G^n) of the smooth fold surfaces described in Chap. 5 allows for a proper structural analysis of smooth fold domains having non-zero thickness using plate or shell representations (in contrast to torsional spring representations used for creased folds [1, 2, 7]). Furthermore, the present model considers kinematic variables (e.g., fold angles) and constraints analogous to those of conventional origami with creased folds. The present model is derived independently from the behavior of the materials comprising the origami structures; therefore, it is applicable to origami structures having regions comprised of passive materials (e.g., elastic) and those having regions comprised of *active materials*. Thus, the present model is useful for the analysis of *active origami structures* such as those reviewed in Chap. 1. To finalize the chapter, we extend the unfolding polyhedra and tuck-folding methods addressed in Chaps. 6 and 7 to develop frameworks for the design of *active origami structures*. The new extensions account for the folding deformation achievable by smooth folds of specified thickness and constituent materials, which is not considered in the purely kinematic formulations of Chaps. 6 and 7.

The outline of this chapter is as follows: Sect. 8.2 presents the kinematic model for origami structures with smooth folds of non-zero thickness used herein. It involves the integration of the surface kinematics model for origami with smooth folds developed in Chap. 5 and existing plate theories to obtain a structural representation for folds of non-zero thickness. The development of the model for the mechanics of origami structures with smooth folds is presented in Sects. 8.3 and 8.4. Section 8.5 provides results of the implemented method and its associated numerical validation against FEA. In Sect. 8.6, we extend the method of unfolding polyhedra from Chap. 6 for the design of active self-folding structures. Likewise, an extension of the tuck-folding method from Chap. 7 is considered in Sect. 8.7.

8.2 Kinematics of Origami Structures with Smooth Folds of Non-Zero Thickness

The kinematic model used to represent origami structures with smooth folds of non-zero thickness is presented here. Employing the notation of Chap. 5, the orthonormal vectors $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ with $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$ form the basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ that defines the fixed global coordinate system. The studied origami structure is a three-dimensional, connected, 3-manifold continuum body with boundary. It is divided into $N_{\mathcal{F}}$ *smooth fold domains* and $N_{\mathcal{P}}$ *face domains*. The origami structure has surface-like geometry and an associated mid-surface which is a three-dimensional, orientable, connected surface with boundary.

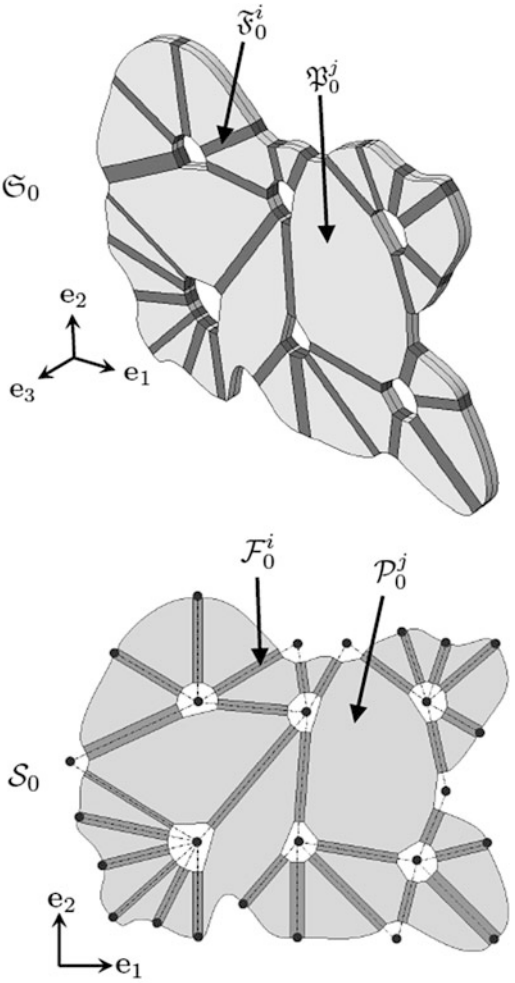


Fig. 8.1 Plate structural representation of an origami structure of non-zero thickness (reference configuration shown). The full volumetric reference configuration of an origami structure of non-zero thickness denoted as $\mathfrak{S}_0$ (thickness not displayed to scale) and its mid-surface $\mathcal{S}_0$ are shown

The *reference configuration* of the origami structure is denoted as $\mathfrak{S}_0$ and has a planar mid-surface denoted as $\mathcal{S}_0 \subset \mathfrak{S}_0$, which is defined such that it is contained in the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$ (refer to Fig. 8.1). The reference configuration of the ith smooth fold domain is denoted as $\mathfrak{F}_0^i \subset \mathfrak{S}_0$ and has an associated mid-surface denoted as $\mathcal{F}_0^i \subset \mathcal{S}_0$, $i = 1, \ldots, N_{\mathcal{F}}$. Also, the reference configuration of the jth face domain is denoted as $\mathfrak{P}_0^j \subset \mathfrak{S}_0$ and has an associated mid-surface denoted as $\mathcal{P}_0^j \subset \mathcal{S}_0$, $j = 1, \ldots, N_{\mathcal{P}}$. Therefore, $\mathfrak{S}_0 = \bigcup_{i=1}^{N_{\mathcal{F}}} \mathfrak{F}_0^i \cup \bigcup_{j=1}^{N_{\mathcal{P}}} \mathfrak{P}_0^j$ and $\mathcal{S}_0 = \bigcup_{i=1}^{N_{\mathcal{F}}} \mathcal{F}_0^i \cup \bigcup_{j=1}^{N_{\mathcal{P}}} \mathcal{P}_0^j$.

A *current configuration* of the origami structure is denoted as $\mathfrak{S}_t$ and has a mid-surface denoted as $\mathcal{S}_t \subset \mathfrak{S}_t$, where t is a *time parameter* that tracks the history of deformation from the reference configuration ($t = 0$) to a current configuration

($t > 0$). The configurations of the smooth fold domains and the face domains in $\mathfrak{S}_t$ are, respectively, denoted as $\mathfrak{F}_t^i, \mathfrak{P}_t^j \subset \mathfrak{S}_t$, $i = 1, \ldots, N_{\mathcal{F}}$, $j = 1, \ldots, N_{\mathcal{P}}$. Their respective mid-surfaces are denoted as $\mathcal{F}_t^i, \mathcal{P}_t^j \subset \mathcal{S}_t$, $i = 1, \ldots, N_{\mathcal{F}}$, $j = 1, \ldots, N_{\mathcal{P}}$. Therefore, $\mathfrak{S}_t = \bigcup_{i=1}^{N_{\mathcal{F}}} \mathfrak{F}_t^i \cup \bigcup_{j=1}^{N_{\mathcal{P}}} \mathfrak{P}_t^j$ and $\mathcal{S}_t = \bigcup_{i=1}^{N_{\mathcal{F}}} \mathcal{F}_t^i \cup \bigcup_{j=1}^{N_{\mathcal{P}}} \mathcal{P}_t^j$.

The deformation of the mid-surface of the origami structure is represented using the surface kinematics model provided in Chap. 5. In this section, we present the plate kinematics used to model a non-zero thickness origami structure having a mid-surface described using the model presented in Chap. 5. As previously stated, the origami structure is divided into $N_{\mathcal{F}}$ smooth fold domains and $N_{\mathcal{P}}$ face domains. Since the face domains are assumed rigid (refer to Sect. 5.2), the only domains that may undergo non-rigid deformations are the smooth folds and hence the focus of this section is on these smooth fold domains.

A schematic of the plate structural representation for the smooth fold domains of non-zero thickness is provided in Fig. 8.2. In the reference configuration, the ith smooth fold domain in the structure has mid-surface $\mathcal{F}_0^i$ and *thickness* $\mathfrak{h}_i$. In a current configuration, the ith smooth fold domain has mid-surface $\mathcal{F}_t^i$ and thickness $\mathfrak{h}_i$ (i.e., the thickness is assumed constant during deformation). The smooth fold domains are assumed to have a parameterization $\mathfrak{F}^i(s_1, s_2, s_3) \in \mathbb{R}^3$ of the following form:

$$\begin{aligned} \mathfrak{F}^i(s_1, s_2, s_3) &= \mathbf{c}^i(s_2) + s_1\mathbf{h}^i + s_3\left(\mathbf{h}^i \times \frac{\mathrm{d}\mathbf{c}^i(s_2)}{\mathrm{d}s_2}\right) \\ &= \boldsymbol{\mathcal{F}}^i(s_1, s_2) + s_3\left(\mathbf{h}^i \times \frac{\mathrm{d}\mathbf{c}^i(s_2)}{\mathrm{d}s_2}\right), \end{aligned} \tag{8.1}$$

where $\boldsymbol{\mathcal{F}}^i(s_1, s_2) \in \mathbb{R}^3$ is a parameterization of the fold mid-surface $\mathcal{F}_t^i$, s_1 and s_2 are the mid-surface coordinates, and $s_3 \in [-\frac{\mathfrak{h}_i}{2}, \frac{\mathfrak{h}_i}{2}]$ is the coordinate through the thickness of the ith smooth fold domain. Note that (8.1) is a clear extension of (5.1) and accounts for the thickness of the smooth fold domain. We also note that the arc-length coordinates s_1 and s_2 are utilized in (8.1) instead of the normalized coordinates ζ_1 and ζ_2 used in (5.1) for convenience in the derivation of the model presented in this chapter.

Here, only folds comprised of materials that exhibit relatively small strains ($\sim$5% or less) are considered. Therefore, the second-order linearized strain tensor $\boldsymbol{\varepsilon}$ is used as the measure for non-rigid deformation [20]. The tensorial components of any field quantity for each smooth fold domain are expressed in their associated local coordinate system with basis $\{\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i\}$ at the reference configuration with origin at $s_1 = s_2 = s_3 = 0$ (refer to Fig. 8.2b, c). Making use of the *classical plate theory* [21], the only considered components of $\boldsymbol{\varepsilon}$ are assumed to linearly vary through the thickness of each smooth fold domain as follows:

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \end{bmatrix} = \begin{bmatrix} \varepsilon_{11_0} \\ \varepsilon_{22_0} \\ \varepsilon_{12_0} \end{bmatrix} - s_3 \begin{bmatrix} \kappa_{11} \\ \kappa_{22} \\ \kappa_{12} \end{bmatrix}, \tag{8.2}$$

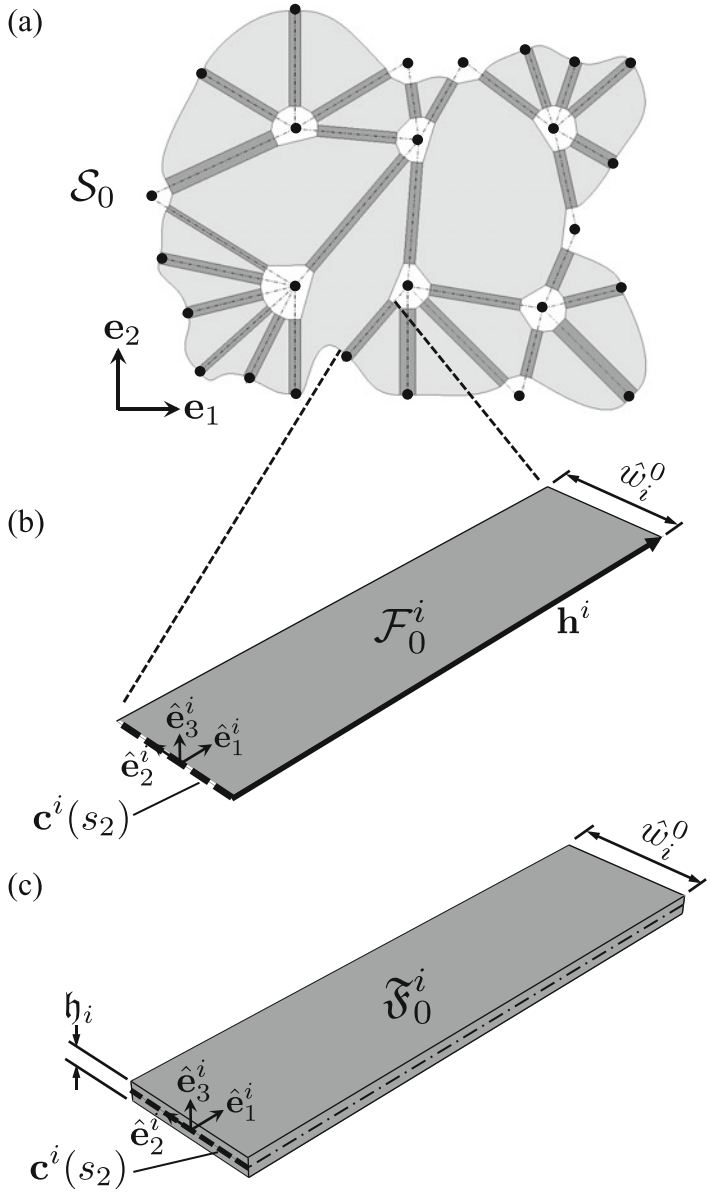


Fig. 8.2 Plate structural representation of an origami structure of non-zero thickness (reference configuration shown): (**a**) Mid-surface of the structure corresponding to $\mathcal{S}_0$; (**b**) Smooth fold surface $\mathcal{F}_0^i$ corresponding to the mid-surface of the ith smooth fold domain; (**c**) Continuum smooth fold domain $\mathfrak{F}_0^i$ having thickness $\mathfrak{h}_i$ and mid-surface $\mathcal{F}_0^i$

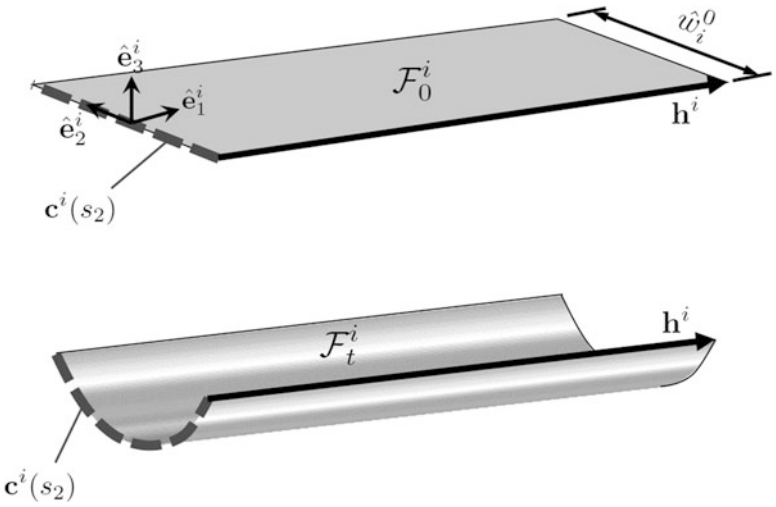


Fig. 8.3 Reference and current configurations of a smooth fold surface ($\mathcal{F}_0^i$ and $\mathcal{F}_t^i$, respectively)

where ε_{mn_0} are the in-plane strains at the mid-surface, and κ_{mn} are the signed curvatures of the mid-surface. Thus, the strain field of the entire smooth fold domain of non-zero thickness depends only on the deformation of its mid-surface.

Reference and current configurations of the smooth fold surface ($\mathcal{F}_0^i$ and $\mathcal{F}_t^i$, respectively) associated with the ith smooth fold domain are shown in Fig. 8.3. It is stated in Chap. 5 that the length $\hat{L}_i = \|\mathbf{h}^i\|$ of the fold along the direction of $\mathbf{h}^i$ is constant upon deformation and hence it is assumed that the strain along the 1-direction is 0. It is also noted that $\frac{\mathrm{d}\mathbf{c}^i(s_2)}{\mathrm{d}s_2} \cdot \mathbf{h}^i = 0$ holds at any configuration (cf. (5.1)) and therefore the shear strain along the 12-plane is also assumed 0 everywhere in the smooth fold domain. In summary:

$$\begin{gathered} \varepsilon_{11} = \varepsilon_{11_0} = \kappa_{11} = 0, \qquad \varepsilon_{22} = \varepsilon_{22_0} - s_3 \kappa_{22}, \\ \varepsilon_{12} = \varepsilon_{12_0} = \kappa_{12} = 0. \end{gathered} \tag{8.3}$$

To simplify the notation, ε_{22}, ε_{22_0}, and κ_{22} are denoted as ε, ε_0, and κ, respectively. As remarked in Chap. 5, the deformation of the smooth folds is limited to *cylindrical bending* (i.e., the deformation of such surfaces is limited to bending and stretch along only one direction). Cylindrical bending is an assumption widely employed in the simplification of the analysis of plate structures that have a high length-to-width ratio and reduces the two-dimensional plate models to one-dimensional ones [22–24]. We assume cylindrical bending because the length of the smooth folds along the direction parallel to $\mathbf{h}^i$ (refer to Fig. 8.3) is often much larger than their fold width. This allows us to neglect stretch and bending along the direction parallel to $\mathbf{h}^i$. As mentioned in Chap. 5, there are material systems such as liquid crystal elastomers [25] and anisotropic materials that do not exhibit

cylindrical bending in general. The present model could be extended to account for general deformation of the smooth fold surfaces by modifying (8.1) and allowing stretch and bending along the direction perpendicular to the fold width. However, such a model would require a higher number of kinematic variables and more complex kinematic relations, which are beyond the scope of this chapter. Finite element approaches using plate, shell, or three-dimensional continuum elements can also be employed to model the deformation of more general cases of folds that do not exhibit cylindrical bending. In Sect. 8.5, simulation results of the present model are compared against finite element models that do not assume cylindrical bending for several boundary value problems.

Following the approach presented in Chap. 5 to determine the shape of the smooth fold cross-section curve $\mathbf{c}^i(s_2)$, by making assumptions on the form of the extensibility field ε_0 and curvature field κ for each smooth fold, the shape of each smooth fold domain is fully determined by its fold angle $\hat{\theta}_i$ and its fold arc-length $\hat{s}_i$ (defined in (5.4)). Therefore:

$$\varepsilon(\hat{\theta}_i, \hat{s}_i) = \varepsilon_0(\hat{\theta}_i, \hat{s}_i) - s_3\, \kappa(\hat{\theta}_i, \hat{s}_i). \tag{8.4}$$

Thus, the strain components of every smooth fold domain expressed in their associated coordinate system with basis $\{\hat{\mathbf{e}}_1^i, \hat{\mathbf{e}}_2^i, \hat{\mathbf{e}}_3^i\}$ are given as follows:

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \end{bmatrix} = \begin{bmatrix} 0 \\ \varepsilon(\hat{\theta}_i, \hat{s}_i) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \varepsilon_0(\hat{\theta}_i, \hat{s}_i) - s_3\, \kappa(\hat{\theta}_i, \hat{s}_i) \\ 0 \end{bmatrix}. \tag{8.5}$$

The derivatives of the strain $\varepsilon(\hat{\theta}_i, \hat{s}_i)$ with respect to $\hat{\theta}_i$ and $\hat{s}_i$ are then given as follows:

$$\frac{\partial \varepsilon(\hat{\theta}_i, \hat{s}_i)}{\partial \hat{\theta}_i} = \frac{\partial \varepsilon_0(\hat{\theta}_i, \hat{s}_i)}{\partial \hat{\theta}_i} - s_3 \frac{\partial \kappa(\hat{\theta}_i, \hat{s}_i)}{\partial \hat{\theta}_i}, \tag{8.6}$$

$$\frac{\partial \varepsilon(\hat{\theta}_i, \hat{s}_i)}{\partial \hat{s}_i} = \frac{\partial \varepsilon_0(\hat{\theta}_i, \hat{s}_i)}{\partial \hat{s}_i} - s_3 \frac{\partial \kappa(\hat{\theta}_i, \hat{s}_i)}{\partial \hat{s}_i}. \tag{8.7}$$

The mid-surface strain $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ and signed curvature $\kappa(\hat{\theta}_i, \hat{s}_i)$ must satisfy the continuity conditions of the order of geometry continuity assumed for the smooth folds (refer to Sect. 5.3). Additionally, the following kinematic relations hold for any $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ and $\kappa(\hat{\theta}_i, \hat{s}_i)$ regardless of the order of geometry continuity:

$$\hat{s}_i - \hat{w}_i^0 = \int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \varepsilon_0 \, \mathrm{d}s_2, \qquad \hat{\theta}_i = \int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \kappa \, \mathrm{d}s_2. \tag{8.8}$$

where $\hat{w}_i^0$ is the *fold width* of the ith smooth fold, illustrated in Fig. 8.2.

The continuity conditions for smooth folds exhibiting G^1 and G^2 continuity are provided in Sect. 5.3. For smooth folds exhibiting G^1 continuity, $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ and $\kappa(\hat{\theta}_i, \hat{s}_i)$ are assumed uniform throughout the fold. For smooth folds exhibiting G^2 continuity, $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ and $\kappa(\hat{\theta}_i, \hat{s}_i)$ are assumed to have a quadratic form with respect to s_2. Such a quadratic form allows for the satisfaction of the conditions for G^2 continuity (i.e., zero curvature at the ends of the curve $\mathbf{c}^i(s_2)$). Making use of the relations provided in (8.8), the following expressions for $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ and $\kappa(\hat{\theta}_i, \hat{s}_i)$ are obtained for smooth folds exhibiting G^1 continuity:

$$\begin{aligned} \varepsilon_0(\hat{\theta}_i, \hat{s}_i) &= \frac{\hat{s}_i - \hat{w}_i^0}{\hat{w}_i^0}, \\ \kappa(\hat{\theta}_i, \hat{s}_i) &= \frac{\hat{\theta}_i}{\hat{w}_i^0}, \end{aligned} \tag{8.9}$$

and the following expressions for $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ and $\kappa(\hat{\theta}_i, \hat{s}_i)$ are obtained for smooth folds exhibiting G^2 continuity:

$$\begin{aligned} \varepsilon_0(\hat{\theta}_i, \hat{s}_i) &= \frac{3}{2}\frac{(\hat{s}_i - \hat{w}_i^0)((\hat{w}_i^0)^2 - 4(s_2)^2)}{(\hat{w}_i^0)^3}, \\ \kappa(\hat{\theta}_i, \hat{s}_i) &= \frac{3}{2}\frac{\hat{\theta}_i((\hat{w}_i^0)^2 - 4(s_2)^2)}{(\hat{w}_i^0)^3}. \end{aligned} \tag{8.10}$$

It is remarked that $\hat{s}_i = \hat{w}_i^0$ in the reference configuration $\mathcal{S}_0$.

Example 8.1 Mid-surface strain and signed curvature fields for smooth folds with G^1 continuity.
Statement: Show that the mid-surface strain and signed curvature fields for smooth folds with G^1 continuity provided in (8.9) satisfy the relations stated in (8.8).
Solution: We first substitute the expression for mid-surface strain $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ from (8.9) into (8.8):

$$\begin{aligned} \int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \varepsilon_0 \, \mathrm{d}s_2 &= \int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \frac{\hat{s}_i - \hat{w}_i^0}{\hat{w}_i^0} \, \mathrm{d}s_2 && \text{substituting (8.9)} \\ &= \frac{\hat{s}_i - \hat{w}_i^0}{\hat{w}_i^0} s_2 \Big|_{s_2 = -\frac{\hat{w}_i^0}{2}}^{s_2 = \frac{\hat{w}_i^0}{2}} \\ &= \frac{\hat{s}_i - \hat{w}_i^0}{\hat{w}_i^0}\left(\frac{\hat{w}_i^0}{2} - \left(-\frac{\hat{w}_i^0}{2}\right)\right) = \hat{s}_i - \hat{w}_i^0. \end{aligned} \tag{8.11}$$

Therefore, the expression for mid-surface strain $\varepsilon_0(\hat{\theta}_i, \hat{s}_i)$ from (8.9) satisfies the relations stated in (8.8). We then substitute the expression for signed curvature $\kappa(\hat{\theta}_i, \hat{s}_i)$ from (8.9) into (8.8):

$$\begin{aligned}
\int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \kappa \, \mathrm{d}s_2 &= \int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \frac{\hat{\theta}_i}{\hat{w}_i^0} \, \mathrm{d}s_2 && \text{substituting (8.9)} \\
&= \frac{\hat{\theta}_i}{\hat{w}_i^0} s_2 \Big|_{s_2=-\frac{\hat{w}_i^0}{2}}^{s_2=\frac{\hat{w}_i^0}{2}} \\
&= \frac{\hat{\theta}_i}{\hat{w}_i^0} \left(\frac{\hat{w}_i^0}{2} - \left(-\frac{\hat{w}_i^0}{2} \right) \right) = \hat{\theta}_i .
\end{aligned} \tag{8.12}$$

Hence, the expression for signed curvature $\kappa(\hat{\theta}_i, \hat{s}_i)$ from (8.9) also satisfies the relations stated in (8.8).

It is also straightforward to show that the mid-surface strain and signed curvature fields in (8.10) satisfy the relations provided in (8.8); refer to Problem 8.1.

8.3 Structural Mechanics Modeling Approach

In this section, the governing field equations used to determine the mechanical response of origami structures with smooth fold domains are presented. Such field equations arise from the law of *conservation of linear momentum* and the law of *conservation of angular momentum* [20, 26]. This section is intended as a review of such conservation laws. For detailed information on these topics, we refer the reader to other sources [20, 26–28]. Since linearized strains are assumed here, no specific distinction between reference and current configurations is considered in this section to simplify the notation.

8.3.1 Conservation of Linear and Angular Momentum

To model the mechanical response of origami structures with smooth fold domains, the law of conservation of linear momentum and the law of conservation of angular momentum must be considered.

The law of conservation of linear momentum states that *the time rate of change of linear momentum of a continuum body is equal to the sum of the surface and body*

forces applied to it [29]. The integral form of conservation of linear momentum is given as follows [29]:

$$\frac{D}{Dt}\left(\int_{\Omega} \rho \dot{\mathbf{u}}\, \mathrm{d}v\right) = \int_{\Omega} \mathbf{b}\, \mathrm{d}v + \int_{\partial\Omega} \mathbf{t}\, \mathrm{d}a, \tag{8.13}$$

where Ω is the region occupied by the material body, $\partial\Omega$ is the boundary of Ω, $\frac{D}{Dt}$ denotes the material time derivative, ρ is the *mass density* of a material point, $\mathbf{u}$ is the *displacement vector* of a material point, $\mathbf{b}$ is the *body force vector*, and $\mathbf{t}$ is the *surface traction vector*. The velocity of a material point is denoted as $\dot{\mathbf{u}}$ where $(\dot{})$ represents the first-order material time derivative. The pointwise form of the conservation of linear momentum is given as follows:

$$\boxed{\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \rho \ddot{\mathbf{u}},} \tag{8.14}$$

where $\boldsymbol{\sigma}$ is the second-order *Cauchy stress tensor* and $\nabla \cdot \boldsymbol{\sigma}$ denotes the divergence of $\boldsymbol{\sigma}$ [27]. The acceleration is denoted as $\ddot{\mathbf{u}}$ where $(\ddot{})$ represents the second-order material time derivative.

The law of conservation of angular momentum states that *the time rate of change of angular momentum of a continuum body is equal to the sum of the moments applied by surface and body forces in addition to body couples* [29]. If there are no body couples, as in all the examples considered in this chapter, the integral form of the conservation of angular momentum is given as follows:

$$\frac{D}{Dt}\left(\int_{\Omega} \mathbf{X} \times \rho \dot{\mathbf{u}}\, \mathrm{d}v\right) = \int_{\Omega} \mathbf{X} \times \mathbf{b}\, \mathrm{d}v + \int_{\partial\Omega} \mathbf{X} \times \mathbf{t}\, \mathrm{d}a, \tag{8.15}$$

where $\mathbf{X}$ is the position vector of a material point. The pointwise form of the conservation of angular momentum states that the second-order Cauchy stress tensor is symmetric:

$$\boxed{\boldsymbol{\sigma} = \boldsymbol{\sigma}^{\top}.} \tag{8.16}$$

Only governing equations arising from conservation of linear and angular momentum are taken into account in the model presented in this chapter. Even though structures comprised of materials exhibiting thermomechanical coupling (e.g., shape memory alloys (SMAs)) are considered herein, the introduced field variables such as the absolute temperature T are taken as given and thus conservation of energy is not considered within the governing equations.

8.3.2 Constitutive Equations

In both conservation of linear momentum (8.14) and conservation of angular momentum (8.16), the constitutive response of the material is manifested in the formulation for the stress tensor $\boldsymbol{\sigma}$, which is a function of the deformation of the material and other physical fields. For *linear elastic materials*, the stress tensor $\boldsymbol{\sigma}$ is simply determined via Hooke's law:

$$\boldsymbol{\sigma} = \boldsymbol{C}\boldsymbol{\varepsilon}, \tag{8.17}$$

where $\boldsymbol{C}$ is the constant fourth-order *stiffness tensor*. For these materials, the stress tensor $\boldsymbol{\sigma}$ is only a function of the strain tensor $\boldsymbol{\varepsilon}$. For *active materials*, *inelastic strains* triggered by *thermal, chemical, or electromagnetic stimuli* are present. We denote the second-order *tensor of actuation strains* as $\boldsymbol{\varepsilon}^A$. The stress tensor $\boldsymbol{\sigma}$ for an active material is determined as follows:

$$\boxed{\boldsymbol{\sigma} = \boldsymbol{C}(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^A).} \tag{8.18}$$

The following expression relates the tensor of actuation strains $\boldsymbol{\varepsilon}^A$ to different physical fields and applies to a large spectrum of commonly used active materials [29–33][1]:

$$\boldsymbol{\varepsilon}^A = \boldsymbol{\alpha}(T - T_0) + \boldsymbol{d}^P\boldsymbol{E} + \boldsymbol{d}^M\boldsymbol{H} + \sum_{i=1}^{n} \boldsymbol{e}^i(c^i - c_0^i) + \boldsymbol{\varepsilon}^{MS}, \tag{8.19}$$

where $\boldsymbol{\alpha}$ is the second-order tensor of thermoelastic expansion coefficients, T_0 is the reference absolute temperature, $\boldsymbol{d}^P$ is the third-order tensor of piezoelectric coefficients, $\boldsymbol{E}$ is the electric field vector, $\boldsymbol{d}^M$ is the third-order tensor of piezomagnetic coefficients, $\boldsymbol{H}$ is the magnetic field vector, $\boldsymbol{e}^i$ is the second-order tensor of expansion due to concentration of the ith chemical species, c^i is the concentration of the ith chemical species, c_0^i is the reference concentration of the ith chemical species, and $\boldsymbol{\varepsilon}^{MS}$ is the second-order tensor of strains caused by changes in the material micro- and/or nano-structure. This last contribution ($\boldsymbol{\varepsilon}^{MS}$) may be altered due to phase transformation [34–36], variant reorientation [37, 38], change in the crosslinked structure in certain polymers [39–42], etc. Such strains are coupled in certain extent to field variables such as $\boldsymbol{\sigma}$, T, $\boldsymbol{H}$, and $\boldsymbol{E}$ depending on the considered material.

[1]This additive decomposition of strain is valid only in the case of linearized strains. The work presented in this chapter considers such a case.

8.3.3 Boundary Value Problem

In the structural analysis problems addressed in this chapter, the objective is to determine the deformation of origami structures with smooth folds subject to certain boundary conditions. First, let the boundary of Ω be partitioned into the boundaries $\partial\Omega_u$ and $\partial\Omega_\sigma$ such that:

$$\partial\Omega_u \cup \partial\Omega_\sigma = \partial\Omega, \qquad \partial\Omega_u \cap \partial\Omega_\sigma = \varnothing, \tag{8.20}$$

where $\partial\Omega_u$ is the boundary of the region occupied by the material body where displacement boundary conditions are applied, $\partial\Omega_\sigma$ is the boundary of the region occupied by the material body where traction boundary conditions are applied, and $\varnothing$ denotes an empty set.

The boundary value problem statement applicable to all the problems considered in this work is given as follows:

$$\begin{aligned}
&\text{Find} && \mathbf{u} && \mathbf{X} \in \Omega, \quad t \geq 0 \\
&\text{Subject to} && \nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \rho\ddot{\mathbf{u}}, && \mathbf{X} \in \Omega, \quad t \geq 0 \\
& && \mathbf{u} = \tilde{\mathbf{u}}, && \mathbf{X} \in \partial\Omega_u, \quad t \geq 0 \\
& && \boldsymbol{\sigma}\mathbf{n} = \tilde{\mathbf{t}}, && \mathbf{X} \in \partial\Omega_\sigma, \quad t \geq 0 \\
& && \Xi = \tilde{\Xi}, && \mathbf{X} \in \Omega, \quad t \geq 0,
\end{aligned} \tag{8.21}$$

where $\tilde{\mathbf{u}}$ is the vector field of *applied boundary displacements*, $\tilde{\mathbf{t}}$ is the vector field of *applied boundary tractions*, $\mathbf{n}$ is the *unit normal vector of the boundary*, Ξ represents other physical fields in the material body (e.g., the absolute temperature field), and $\tilde{\Xi}$ represents other physical field variables imposed in the material body (e.g., the applied absolute temperature field).

The governing field equation of the boundary value problem in (8.21) arises from conservation of linear momentum (cf. (8.14)). The other conditions stated in (8.21) correspond to the *boundary conditions* of the problem. These boundary conditions appear in the form of applied boundary displacements $\tilde{\mathbf{u}}$, applied boundary tractions $\tilde{\mathbf{t}}$, and applied field variables $\tilde{\Xi}$. Although not stated explicitly in (8.21), it is assumed beforehand that the second-order Cauchy stress tensor $\boldsymbol{\sigma}$ is symmetric to satisfy conservation of angular momentum (8.16).

8.3.4 Variational Formulation

Variational approaches are powerful tools for the development of exact or approximate solutions to problems in structural mechanics of continuum bodies [26, 27] (e.g., FEA). Here, we employ a variational approach to determine approximate

solutions for boundary value problems of the form presented in Sect. 8.3.3. Hamilton's principle is considered for the modeling of dynamic deformation of continuum bodies [27]:

$$\int_{t_1}^{t_2} (\delta G + \delta W_I - \delta W_E)\, \mathrm{d}t = 0, \tag{8.22}$$

where δG is the virtual work done by the inertial forces given a virtual displacement, δW_I is the virtual work of all internal forces given a virtual displacement, δW_E is the virtual work of all external forces given a virtual displacement, and t_1 and t_2 are two distinct arbitrary times. The formulations of δG, δW_I, and δW_E are, respectively, given as follows[2]:

$$\delta G = \int_{\Omega} \rho \ddot{\mathbf{u}} \cdot \delta \mathbf{u}\, \mathrm{d}v, \tag{8.23}$$

$$\delta W_I = \int_{\Omega} \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon}\, \mathrm{d}v, \tag{8.24}$$

$$\delta W_E = \int_{\Omega} \mathbf{b} \cdot \delta \mathbf{u}\, \mathrm{d}v + \int_{\partial \Omega_\sigma} \mathbf{t} \cdot \delta \mathbf{u}\, \mathrm{d}a, \tag{8.25}$$

where $\delta \mathbf{u}$ is the *virtual displacement* which is null on the boundary $\partial \Omega_u$ where displacements are applied through boundary conditions (i.e., $\delta \mathbf{u}(\mathbf{X}, t) = \mathbf{0}_3,\ \mathbf{X} \in \partial \Omega_u,\ t \in [t_1, t_2]$) as well at times t_1 and t_2 (i.e., $\delta \mathbf{u}(\mathbf{X}, t_1) = \delta \mathbf{u}(\mathbf{X}, t_2) = \mathbf{0}_3,\ \mathbf{X} \in \Omega$), and $\delta \boldsymbol{\varepsilon}$ is the *virtual strain* due to the virtual displacement:

$$\delta \boldsymbol{\varepsilon} = \frac{1}{2}\Big(\nabla(\delta \mathbf{u}) + \nabla(\delta \mathbf{u})^{\top}\Big), \tag{8.26}$$

where $\nabla(\delta \mathbf{u})$ denotes the gradient of $\delta \mathbf{u}$.

8.4 Structural Mechanics Model Formulation

8.4.1 Model Development

The structural modeling approach described in the previous section is used here to develop a model for the mechanics of origami structures with smooth fold domains comprised of arbitrary materials.

[2] $(\cdot : \cdot)$ denotes the inner product of two second-order tensors. If such tensors are expressed in an orthonormal coordinate system, their inner product is given as $\mathbf{Y} : \mathbf{Z} = \sum_{i=1}^{3} \sum_{j=1}^{3} Y_{ij} Z_{ij}$.

The plate kinematics addressed in Sect. 8.2 lead to a displacement field defined by the deformation of the mid-surface of the origami structure. The deformation of such a mid-surface is represented using the surface kinematics model presented in Chap. 5. The configuration of such a mid-surface is fully defined by the fold angles and fold arc-lengths of every smooth fold in the sheet. Therefore, the displacement field of an origami structure with smooth fold domains is a function of $\hat{\theta}_i$ and $\hat{s}_i$, $i = 1, \ldots, N_{\mathcal{F}}$.

In the surface kinematics model presented in Chap. 5, it is assumed that one face of the sheet is fixed in space (see Sect. 5.6). The reference configuration of such a *fixed face* is denoted as $\mathcal{P}_0^*$. Therefore, the only displacement boundary condition considered here is that the displacement vector is equal to the zero vector at every point in the fixed face:

$$\mathbf{u} = \mathbf{0}_3, \quad \mathbf{X} \in \mathcal{P}_0^*, \quad t \geq 0. \tag{8.27}$$

The virtual displacement and the virtual strain, encountered in the formulation of the Hamilton's principle (refer to (8.23)–(8.25)), are related to virtual changes in fold angles and fold arc-lengths as follows:

$$\delta\mathbf{u} = \frac{\partial \mathbf{u}}{\partial \hat{\boldsymbol{\theta}}} \cdot \delta\hat{\boldsymbol{\theta}} + \frac{\partial \mathbf{u}}{\partial \hat{\mathbf{s}}} \cdot \delta\hat{\mathbf{s}}, \qquad \delta\boldsymbol{\varepsilon} = \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{\boldsymbol{\theta}}} \cdot \delta\hat{\boldsymbol{\theta}} + \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{\mathbf{s}}} \cdot \delta\hat{\mathbf{s}}. \tag{8.28}$$

Since the configuration of an origami structure is defined by the fold angles and fold arc-lengths of each smooth fold, the kinematic constraints (5.66) and (5.67) allowing for valid configurations in origami sheets with smooth folds must be considered. The Hamilton's principle as stated in (8.22) does not take into consideration any kinematic constraints. This kind of constraints are conventionally included through the method of Lagrange multipliers or penalty methods [26, 27]. A penalty method is employed here. Accordingly, let $\mathfrak{L}$ be a *penalty function associated with the kinematic constraints* (5.66) and (5.67):

$$\begin{aligned} \mathfrak{L}(\hat{\boldsymbol{\theta}}, \hat{\mathbf{s}}) = \lambda_R \sum_{i=1}^{N_{\mathcal{I}}} \left(\left(R_{23}^i(\hat{\boldsymbol{\theta}}) \right)^2 + \left(R_{31}^i(\hat{\boldsymbol{\theta}}) \right)^2 + \left(R_{12}^i(\hat{\boldsymbol{\theta}}) \right)^2 \right) \\ + \lambda_d \sum_{i=1}^{N_{\mathcal{I}}} \left(\left(d_1^i(\hat{\boldsymbol{\theta}}, \hat{\mathbf{s}}) \right)^2 + \left(d_2^i(\hat{\boldsymbol{\theta}}, \hat{\mathbf{s}}) \right)^2 + \left(d_3^i(\hat{\boldsymbol{\theta}}, \hat{\mathbf{s}}) \right)^2 \right), \end{aligned} \tag{8.29}$$

where $\lambda_R \in \mathbb{R}_{>0}$ is the penalty scalar for rotation constraints and $\lambda_d \in \mathbb{R}_{>0}$ is the penalty scalar for translation constraints. The modified Hamilton's principle allowing for the consideration of kinematic constraints is then given as follows:

$$\int_{t_1}^{t_2} (\delta G + \delta W_I - \delta W_E + \delta \mathfrak{L}) \, \mathrm{d}t = 0, \tag{8.30}$$

cf. (8.22). By substituting (8.28) into the expression for δW_I in (8.24), the following is obtained:

$$\delta W_I = \int_{\mathfrak{S}} \boldsymbol{\sigma} : \delta\boldsymbol{\varepsilon} \, \mathrm{d}v = \int_{\mathfrak{S}} \boldsymbol{\sigma} : \left(\frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{\boldsymbol{\theta}}} \cdot \delta\hat{\boldsymbol{\theta}} + \frac{\partial \boldsymbol{\varepsilon}}{\partial \hat{\mathbf{s}}} \cdot \delta\hat{\mathbf{s}} \right) \mathrm{d}v, \tag{8.31}$$

where $\mathfrak{S}$ is the region occupied by the origami structure. The expression for δW_I provided in (8.31) can be decomposed into the contributions of each smooth fold domain in the origami body[3]:

$$\delta W_I = \sum_{i=1}^{N_{\mathcal{F}}} \int_{\mathfrak{F}^i} \sigma \left(\frac{\partial \varepsilon}{\partial \hat{\theta}_i} \delta\hat{\theta}_i + \frac{\partial \varepsilon}{\partial \hat{s}_i} \delta\hat{s}_i \right) \mathrm{d}v, \tag{8.32}$$

where $\sigma = \sigma_{22}$ in the fold coordinate system of the ith smooth fold domain and $\mathfrak{F}^i$ is the region occupied by the ith smooth fold domain. By performing integration over the thickness in the expression presented in (8.32), the following is obtained:

$$\delta W_I = \sum_{i=1}^{N_{\mathcal{F}}} \int_{\mathcal{F}_i} \left(\left(N \frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} - M \frac{\partial \kappa}{\partial \hat{\theta}_i} \right) \delta\hat{\theta}_i + \left(N \frac{\partial \varepsilon_0}{\partial \hat{s}_i} - M \frac{\partial \kappa}{\partial \hat{s}_i} \right) \delta\hat{s}_i \right) \mathrm{d}a, \tag{8.33}$$

where N and M are, respectively, the *axial force and moment resultants* given as follows [21]:

$$N = \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \sigma \, \mathrm{d}s_3, \qquad M = \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} s_3 \sigma \, \mathrm{d}s_3. \tag{8.34}$$

By substituting (8.28) into the expression for δW_E in (8.25), the following is obtained:

$$\begin{aligned}
\delta W_E &= \int_{\mathfrak{S}} \mathbf{b} \cdot \delta\mathbf{u} \, \mathrm{d}v + \int_{\partial\mathfrak{S}} \mathbf{t} \cdot \delta\mathbf{u} \, \mathrm{d}a \\
&= \int_{\mathfrak{S}} \mathbf{b} \cdot \left(\frac{\partial \mathbf{u}}{\partial \hat{\boldsymbol{\theta}}} \cdot \delta\hat{\boldsymbol{\theta}} + \frac{\partial \mathbf{u}}{\partial \hat{\mathbf{s}}} \cdot \delta\hat{\mathbf{s}} \right) \mathrm{d}v + \int_{\partial\mathfrak{S}} \mathbf{t} \cdot \left(\frac{\partial \mathbf{u}}{\partial \hat{\boldsymbol{\theta}}} \cdot \delta\hat{\boldsymbol{\theta}} + \frac{\partial \mathbf{u}}{\partial \hat{\mathbf{s}}} \cdot \delta\hat{\mathbf{s}} \right) \mathrm{d}a \\
&= \int_{\mathfrak{S}} \mathbf{b} \cdot \sum_{i=1}^{N_{\mathcal{F}}} \left(\frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} \delta\hat{\theta}_i + \frac{\partial \mathbf{u}}{\partial \hat{s}_i} \delta\hat{s}_i \right) \mathrm{d}v + \int_{\partial\mathfrak{S}} \mathbf{t} \cdot \sum_{i=1}^{N_{\mathcal{F}}} \left(\frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} \delta\hat{\theta}_i + \frac{\partial \mathbf{u}}{\partial \hat{s}_i} \delta\hat{s}_i \right) \mathrm{d}a,
\end{aligned} \tag{8.35}$$

[3] Integration over the volume of the smooth fold domain is given as $\int_{\mathfrak{F}^i} (\cdot) \, \mathrm{d}v = \int_{\mathcal{F}^i} \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} (\cdot) \, \mathrm{d}s_3 \, \mathrm{d}a = \int_{-\frac{\hat{w}_i^0}{2}}^{\frac{\hat{w}_i^0}{2}} \int_0^{\hat{L}_i} \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} (\cdot) \, \mathrm{d}s_3 \, \mathrm{d}s_2 \, \mathrm{d}s_1$.

where $\partial\mathfrak{S}$ is the boundary of $\mathfrak{S}$. The displacement vector $\mathbf{u}$ is a function of fold angles and fold arc-lengths that is determined using the mapping between reference and current configurations for origami with smooth folds provided in Sect. 5.6.

The virtual increment in the penalty function $\mathfrak{L}$ defined in (8.29) can be expressed as follows:

$$\begin{aligned}\delta\mathfrak{L} &= \frac{\partial\mathfrak{L}}{\partial\hat{\boldsymbol{\theta}}}\cdot\delta\hat{\boldsymbol{\theta}} + \frac{\partial\mathfrak{L}}{\partial\hat{\mathbf{s}}}\cdot\delta\hat{\mathbf{s}} \\ &= \sum_{i=1}^{N_{\mathcal{F}}}\left(\frac{\partial\mathfrak{L}}{\partial\hat{\theta}_i}\delta\hat{\theta}_i + \frac{\partial\mathfrak{L}}{\partial\hat{s}_i}\delta\hat{s}_i\right).\end{aligned} \tag{8.36}$$

For the case of *quasi-static* deformation assumed in the examples considered in this work, Hamilton's principle reduces to the *principle of virtual work* [27]. The principle of virtual work states that a continuum body is in mechanical equilibrium (i.e., the boundary conditions and the field equation without the inertial terms in (8.21) are satisfied) if and only if the virtual work of all forces, internal and external, acting on the body is zero given a virtual displacement [27]:

$$\delta W = \delta W_I - \delta W_E = 0, \tag{8.37}$$

where δW represents the virtual work of all internal and external forces given a virtual displacement and δW_I and δW_E are, respectively, provided in (8.24) and (8.25). The modified principle of virtual work allowing for the consideration of kinematic constraints is then given as follows:

$$\delta W_I - \delta W_E + \delta\mathfrak{L} = 0. \tag{8.38}$$

It is noted that the principle of virtual work in (8.38) does not consider any assumptions on the constitutive behavior of the material(s) comprising the continuum body. Therefore, it is applicable to any continuum body having arbitrary constitutive behavior [27] so long as strains are small enough such that the linearized strain tensor can be utilized to properly model the deformation.

The following is obtained by substituting the expressions for δW_I, δW_E, and $\delta\mathfrak{L}$ provided in (8.33), (8.35), and (8.36), respectively, into (8.38):

$$
\begin{aligned}
0 &= \delta W_I - \delta W_E + \delta \mathfrak{L} \\
&= \sum_{i=1}^{N_{\mathcal{F}}} \int_{\mathcal{F}_i} \left(\left(N \frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} - M \frac{\partial \kappa}{\partial \hat{\theta}_i} \right) \delta \hat{\theta}_i + \left(N \frac{\partial \varepsilon_0}{\partial \hat{s}_i} - M \frac{\partial \kappa}{\partial \hat{s}_i} \right) \delta \hat{s}_i \right) \mathrm{d}a \\
&\quad - \int_{\mathfrak{S}} \mathbf{b} \cdot \sum_{i=1}^{N_{\mathcal{F}}} \left(\frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} \delta \hat{\theta}_i + \frac{\partial \mathbf{u}}{\partial \hat{s}_i} \delta \hat{s}_i \right) \mathrm{d}v - \int_{\partial \mathfrak{S}} \mathbf{t} \cdot \sum_{i=1}^{N_{\mathcal{F}}} \left(\frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} \delta \hat{\theta}_i + \frac{\partial \mathbf{u}}{\partial \hat{s}_i} \delta \hat{s}_i \right) \mathrm{d}a \\
&\quad + \sum_{i=1}^{N_{\mathcal{F}}} \left(\frac{\partial \mathfrak{L}}{\partial \hat{\theta}_i} \delta \hat{\theta}_i + \frac{\partial \mathfrak{L}}{\partial \hat{s}_i} \delta \hat{s}_i \right).
\end{aligned}
\tag{8.39}
$$

Constitutive models for the materials comprising the smooth fold domains in the examples presented in this chapter are provided in Appendix C. These include linear elastic, thermoelastic, and SMA materials.

8.4.2 Numerical Implementation

To determine a configuration satisfying conservation of linear and angular momentum and kinematic constraints (5.66) and (5.67), the coefficients multiplying each virtual increment in fold angle $\delta \hat{\theta}_i$ and in fold arc-length $\delta \hat{s}_i$ in (8.39) must be set to 0. In a numerical framework such as the one to be developed here, these coefficients are placed as components of residual vectors and iteratively minimized in magnitude using Newton's method [43].

Let $\mathfrak{R}^\theta \in \mathbb{R}^{N_{\mathcal{F}}}$ be the residual vector whose components $\mathfrak{R}_i^\theta$ are the coefficients of each virtual increment in fold angle $\delta \hat{\theta}_i$ in (8.39):

$$
\mathfrak{R}_i^\theta = \int_{\mathcal{F}_i} \left(N \frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} - M \frac{\partial \kappa}{\partial \hat{\theta}_i} \right) \mathrm{d}a - \int_{\mathfrak{S}} \mathbf{b} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} \mathrm{d}v - \int_{\partial \mathfrak{S}} \mathbf{t} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} \mathrm{d}a + \frac{\partial \mathfrak{L}}{\partial \hat{\theta}_i}. \tag{8.40}
$$

Likewise, let $\mathfrak{R}^s \in \mathbb{R}^{N_{\mathcal{F}}}$ be the residual vector whose components $\mathfrak{R}_i^s$ are the coefficients of each virtual increment in fold arc-length $\delta \hat{s}_i$ in (8.39):

$$
\mathfrak{R}_i^s = \int_{\mathcal{F}_i} \left(N \frac{\partial \varepsilon_0}{\partial \hat{s}_i} - M \frac{\partial \kappa}{\partial \hat{s}_i} \right) \mathrm{d}a - \int_{\mathfrak{S}} \mathbf{b} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} \mathrm{d}v - \int_{\partial \mathfrak{S}} \mathbf{t} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} \mathrm{d}a + \frac{\partial \mathfrak{L}}{\partial \hat{s}_i}. \tag{8.41}
$$

The loads applied to the origami structures with smooth fold domains are evolved in various loading increments. For each load increment, the configuration that satisfies (8.39) must be determined. The present implementation iteratively solves

Table 8.1 Numerical procedure followed at the lth loading increment to determine the response of origami structures with smooth fold domains of non-zero thickness

1:	Determine the components of the residual vectors $\mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(1)}, \hat{\mathbf{s}}^{l(1)})$ and $\mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(1)}, \hat{\mathbf{s}}^{l(1)})$
2:	`IF` $(\|\mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(1)}, \hat{\mathbf{s}}^{l(1)})\| <$ `tol1 AND` $\|\mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(1)}, \hat{\mathbf{s}}^{l(1)})\| <$ `tol2)`
	`THEN` set $\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{l(1)}, \hat{\mathbf{s}}^{l} = \hat{\mathbf{s}}^{l(1)}$ and `EXIT`
	`ELSE CONTINUE`
3:	Determine corrections $\Delta\hat{\boldsymbol{\theta}}^{l(k)}$ and $\Delta\hat{\mathbf{s}}^{l(k)}$ using (8.42)
4:	Update $\hat{\boldsymbol{\theta}}^{l(k+1)}$ and $\hat{\mathbf{s}}^{l(k+1)}$ using (8.44)
5:	`IF` $(\|\mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(k+1)}, \hat{\mathbf{s}}^{l(k+1)})\| <$ `tol1 AND` $\|\mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(k+1)}, \hat{\mathbf{s}}^{l(k+1)})\| <$ `tol2)`
	`OR` $(\|\Delta\hat{\boldsymbol{\theta}}^{l(k)}\| <$ `tol3 AND` $\|\Delta\hat{\mathbf{s}}^{l(k)}\| <$ `tol4)`
	`THEN` set $\hat{\boldsymbol{\theta}}^{l} = \hat{\boldsymbol{\theta}}^{l(k+1)}, \hat{\mathbf{s}}^{l} = \hat{\mathbf{s}}^{l(k+1)}$ and `EXIT`
	`ELSE` set $k \leftarrow k+1$ and `GOTO` 3

for increments in fold angles and fold arc-lengths that minimize the magnitudes of the components of the vectors $\mathfrak{R}^{\theta}$ and $\mathfrak{R}^{s}$, which must tend to zero (or sufficiently close to zero by introducing a numerical tolerance) for a configuration to satisfy (8.39). Using Newton's method, the fold angles and fold arc-lengths at the lth loading increment are corrected iteratively as follows:

$$\begin{bmatrix} \Delta\hat{\boldsymbol{\theta}}^{l(k)} \\ \Delta\hat{\mathbf{s}}^{l(k)} \end{bmatrix} = -\begin{bmatrix} \dfrac{\partial\mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(k)}, \hat{\mathbf{s}}^{l(k)})}{\partial\hat{\boldsymbol{\theta}}} & \dfrac{\partial\mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(k)}, \hat{\mathbf{s}}^{l(k)})}{\partial\hat{\mathbf{s}}} \\ \dfrac{\partial\mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(k)}, \hat{\mathbf{s}}^{l(k)})}{\partial\hat{\boldsymbol{\theta}}} & \dfrac{\partial\mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(k)}, \hat{\mathbf{s}}^{l(k)})}{\partial\hat{\mathbf{s}}} \end{bmatrix}^{-1} \begin{bmatrix} \mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(k)}, \hat{\mathbf{s}}^{l(k)}) \\ \mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(k)}, \hat{\mathbf{s}}^{l(k)}) \end{bmatrix}, \tag{8.42}$$

$$\hat{\boldsymbol{\theta}}^{l(k+1)} = \hat{\boldsymbol{\theta}}^{l(k)} + \Delta\hat{\boldsymbol{\theta}}^{l(k)}, \tag{8.43}$$

$$\hat{\mathbf{s}}^{l(k+1)} = \hat{\mathbf{s}}^{l(k)} + \Delta\hat{\mathbf{s}}^{l(k)}, \tag{8.44}$$

where the superscripts inside parenthesis correspond to the correction iteration number. The correction iterations in (8.42)–(8.44) are repeated until $\|\mathfrak{R}^{\theta}(\hat{\boldsymbol{\theta}}^{l(k+1)}, \hat{\mathbf{s}}^{l(k+1)})\| <$ `tol1` and $\|\mathfrak{R}^{s}(\hat{\boldsymbol{\theta}}^{l(k+1)}, \hat{\mathbf{s}}^{l(k+1)})\| <$ `tol2`, or alternatively until $\|\Delta\hat{\boldsymbol{\theta}}^{l(k)}\| <$ `tol3` and $\|\Delta\hat{\mathbf{s}}^{l(k)}\| <$ `tol4` (where `tol1`, `tol2`, `tol3`, and `tol4` are numerical tolerances). Table 8.1 summarizes the numerical procedure followed to simulate origami structures of non-zero thickness.

The components of the tangent matrices $\frac{\partial\mathfrak{R}^{\theta}}{\partial\hat{\boldsymbol{\theta}}}$, $\frac{\partial\mathfrak{R}^{\theta}}{\partial\hat{\mathbf{s}}}$, $\frac{\partial\mathfrak{R}^{s}}{\partial\hat{\boldsymbol{\theta}}}$, and $\frac{\partial\mathfrak{R}^{s}}{\partial\hat{\mathbf{s}}}$ must be calculated to determine the corrective increments in fold angles and fold arc-lengths as given in (8.42). These components are given as follows (see Problem 8.2):

$$\begin{aligned}\frac{\partial \mathfrak{R}_i^\theta}{\partial \hat{\theta}_i} &= \int_{\mathcal{F}_i} \left(\frac{\partial N}{\partial \hat{\theta}_i} \frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} + N \frac{\partial^2 \varepsilon_0}{\partial \hat{\theta}_i^2} - \frac{\partial M}{\partial \hat{\theta}_i} \frac{\partial \kappa}{\partial \hat{\theta}_i} - M \frac{\partial^2 \kappa}{\partial \hat{\theta}_i^2} \right) \mathrm{d}a \\ &\quad - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{\theta}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i^2} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{\theta}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i^2} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{\theta}_i^2},\end{aligned} \tag{8.45}$$

$$\begin{aligned}\frac{\partial \mathfrak{R}_i^\theta}{\partial \hat{s}_i} &= \int_{\mathcal{F}_i} \left(\frac{\partial N}{\partial \hat{s}_i} \frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} + N \frac{\partial^2 \varepsilon_0}{\partial \hat{\theta}_i \partial \hat{s}_i} - \frac{\partial M}{\partial \hat{s}_i} \frac{\partial \kappa}{\partial \hat{\theta}_i} - M \frac{\partial^2 \kappa}{\partial \hat{\theta}_i \partial \hat{s}_i} \right) \mathrm{d}a \\ &\quad - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{s}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{s}_i} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{s}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{s}_i} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{\theta}_i \partial \hat{s}_i},\end{aligned} \tag{8.46}$$

$$\begin{aligned}\left. \frac{\partial \mathfrak{R}_i^\theta}{\partial \hat{\theta}_j} \right|_{j \neq i} &= - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{\theta}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{\theta}_j} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{\theta}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{\theta}_j} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{\theta}_i \partial \hat{\theta}_j},\end{aligned} \tag{8.47}$$

$$\begin{aligned}\left. \frac{\partial \mathfrak{R}_i^\theta}{\partial \hat{s}_j} \right|_{j \neq i} &= - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{s}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{s}_j} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{s}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{\theta}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{s}_j} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{\theta}_i \partial \hat{s}_j},\end{aligned} \tag{8.48}$$

$$\begin{aligned}\frac{\partial \mathfrak{R}_i^s}{\partial \hat{s}_i} &= \int_{\mathcal{F}_i} \left(\frac{\partial N}{\partial \hat{s}_i} \frac{\partial \varepsilon_0}{\partial \hat{s}_i} + N \frac{\partial^2 \varepsilon_0}{\partial \hat{s}_i^2} - \frac{\partial M}{\partial \hat{s}_i} \frac{\partial \kappa}{\partial \hat{s}_i} - M \frac{\partial^2 \kappa}{\partial \hat{s}_i^2} \right) \mathrm{d}a \\ &\quad - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{s}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{s}_i^2} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{s}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{s}_i^2} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{s}_i^2},\end{aligned} \tag{8.49}$$

$$\begin{aligned}\frac{\partial \mathfrak{R}_i^s}{\partial \hat{\theta}_i} &= \int_{\mathcal{F}_i} \left(\frac{\partial N}{\partial \hat{\theta}_i} \frac{\partial \varepsilon_0}{\partial \hat{s}_i} + N \frac{\partial^2 \varepsilon_0}{\partial \hat{\theta}_i \partial \hat{s}_i} - \frac{\partial M}{\partial \hat{\theta}_i} \frac{\partial \kappa}{\partial \hat{s}_i} - M \frac{\partial^2 \kappa}{\partial \hat{\theta}_i \partial \hat{s}_i} \right) \mathrm{d}a \\ &\quad - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{\theta}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{s}_i} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{\theta}_i} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{\theta}_i \partial \hat{s}_i} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{\theta}_i \partial \hat{s}_i},\end{aligned} \tag{8.50}$$

$$\begin{aligned}\left. \frac{\partial \mathfrak{R}_i^s}{\partial \hat{s}_j} \right|_{j \neq i} &= - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{s}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{s}_i \partial \hat{s}_j} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{s}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{s}_i \partial \hat{s}_j} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{s}_i \partial \hat{s}_j},\end{aligned} \tag{8.51}$$

$$\begin{aligned}\left. \frac{\partial \mathfrak{R}_i^s}{\partial \hat{\theta}_j} \right|_{j \neq i} &= - \int_{\mathfrak{S}} \left(\frac{\partial \mathbf{b}}{\partial \hat{\theta}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{b} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{s}_i \partial \hat{\theta}_j} \right) \mathrm{d}v \\ &\quad - \int_{\partial \mathfrak{S}} \left(\frac{\partial \mathbf{t}}{\partial \hat{\theta}_j} \cdot \frac{\partial \mathbf{u}}{\partial \hat{s}_i} + \mathbf{t} \cdot \frac{\partial^2 \mathbf{u}}{\partial \hat{s}_i \partial \hat{\theta}_j} \right) \mathrm{d}a + \frac{\partial^2 \mathfrak{L}}{\partial \hat{s}_i \partial \hat{\theta}_j},\end{aligned} \tag{8.52}$$

where the derivatives of the axial force and moment resultants with respect to fold angle and fold arc-length are given as follows:

$$\begin{aligned}\frac{\partial N}{\partial \hat{\theta}_i} &= \frac{\partial}{\partial \hat{\theta}_i} \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \sigma \, \mathrm{d}s_3 \\ &= \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \frac{\partial \sigma}{\partial \hat{\theta}_i} \, \mathrm{d}s_3 \\ &= \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \frac{\partial \sigma}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial \hat{\theta}_i} \, \mathrm{d}s_3 \\ &= \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \frac{\partial \sigma}{\partial \varepsilon} \left(\frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} - s_3 \frac{\partial \kappa}{\partial \hat{\theta}_i} \right) \mathrm{d}s_3,\end{aligned} \tag{8.53}$$

Similarly:

$$\frac{\partial N}{\partial \hat{s}_i} = \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \frac{\partial \sigma}{\partial \varepsilon} \left(\frac{\partial \varepsilon_0}{\partial \hat{s}_i} - s_3 \frac{\partial \kappa}{\partial \hat{s}_i} \right) \mathrm{d}s_3, \tag{8.54}$$

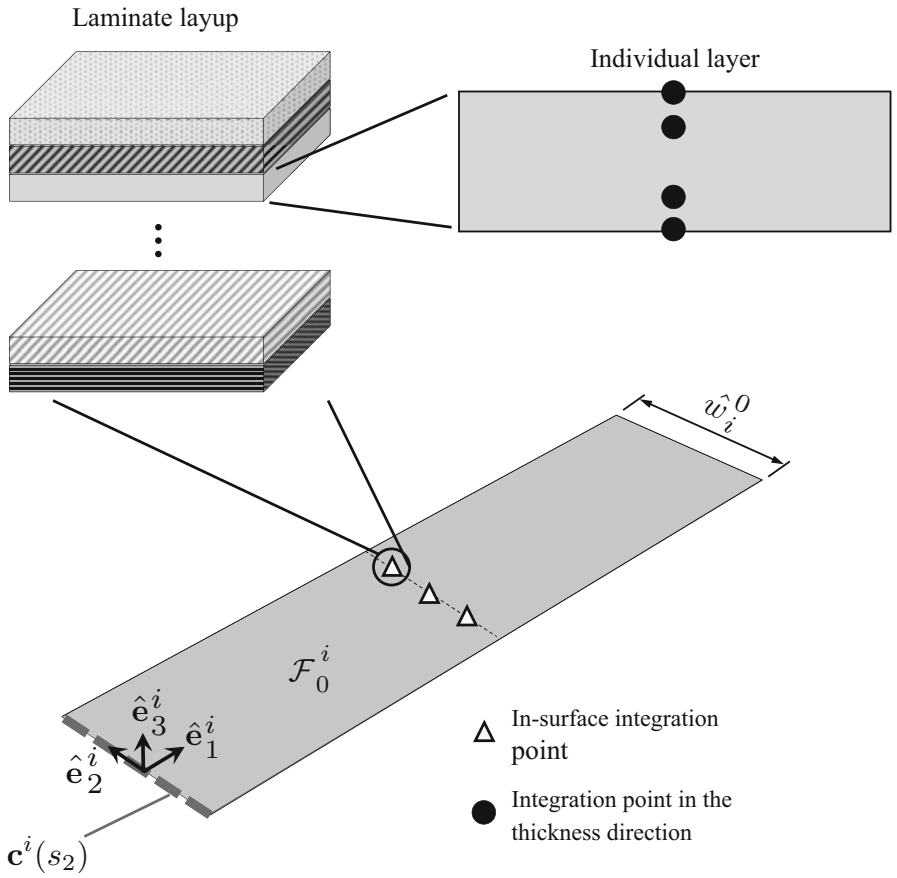


Fig. 8.4 Schematic of a smooth fold surface showing the location of in-surface integration points and integration points through the thickness of each layer

$$\frac{\partial M}{\partial \hat{\theta}_i} = \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \frac{\partial \sigma}{\partial \varepsilon} \left(s_3 \frac{\partial \varepsilon_0}{\partial \hat{\theta}_i} - (s_3)^2 \frac{\partial \kappa}{\partial \hat{\theta}_i} \right) \mathrm{d}s_3, \tag{8.55}$$

$$\frac{\partial M}{\partial \hat{s}_i} = \int_{-\frac{\mathfrak{h}_i}{2}}^{\frac{\mathfrak{h}_i}{2}} \frac{\partial \sigma}{\partial \varepsilon} \left(s_3 \frac{\partial \varepsilon_0}{\partial \hat{s}_i} - (s_3)^2 \frac{\partial \kappa}{\partial \hat{s}_i} \right) \mathrm{d}s_3. \tag{8.56}$$

The evaluation of the integrals over the smooth fold surfaces and over the thickness of the smooth fold domains in the previous equations is performed through *numerical integration*. Figure 8.4 shows the location of integration points used for in-surface integration and also the location of integration points through the thickness of each layer. Integration points at only one location along the 1-direction

(i.e., that aligned with $\hat{\mathbf{e}}_1^i$) are needed as strain is assumed uniform throughout this direction. A Gauss-Legendre quadrature rule is used for integration along the 2-direction [44, 45]. The number of integration points needed in the 2-direction is dictated by assumed in-plane strain and curvature fields in the fold domain (refer to Sect. 8.2). Integration through the thickness of the smooth fold domains is performed layerwise as shown in Fig. 8.4. For each individual layer, a Gauss-Lobatto quadrature rule is used for integration because it considers integration points at the ends of the integration domain (i.e., the top and bottom of each layer), which often are the locations of maximum stress.

8.4.2.1 Quadrature Rules for Numerical Integration

Here we provide the quadrature rules used in the implementation of the model for the mechanics of origami structures presented in this chapter. Previously, we stated that Gauss-Legendre and Gauss-Lobatto quadrature rules are employed. To provide a general formulation for these quadrature rules, let us consider a function $f(x)$ that we want to integrate in the interval $[a, b]$:

$$\int_a^b f(x)\,\mathrm{d}x. \tag{8.57}$$

The function $f(x)$ may or may not have a closed form *antiderivative* $F(x)$ where:

$$\frac{\mathrm{d}F(x)}{\mathrm{d}x} = f(x). \tag{8.58}$$

In the case where $f(x)$ has a closed-form antiderivative, we evaluate the integral (8.57) as follows:

$$\int_a^b f(x)\,\mathrm{d}x = F(b) - F(a). \tag{8.59}$$

In general, the function $f(x)$ that we want to integrate may not have a closed-form antiderivative or it may be very difficult to determine. This is the case in many of the examples presented in this chapter. In such cases, we utilize *numerical integration* to evaluate (8.57). Here we employ two different *quadrature rules* to perform numerical integration. A *quadrature rule* is an approximation of the integral of a function that is stated as a weighted sum of the function values at specified points contained in the domain of integration.

To formulate the employed Gauss-Legendre and Gauss-Lobatto quadrature rules, we first perform a variable substitution from $x \in [a, b]$ to $\eta \in [-1, 1]$:

$$\int_a^b f(x)\,\mathrm{d}x = \frac{b-a}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}\eta + \frac{a+b}{2}\right)\mathrm{d}\eta. \tag{8.60}$$

Table 8.2 Location of integration points and weights for the Gauss-Legendre quadrature rule

Number of integration points	Position η_i	Weight ω_i
1	0	2
2	$-(\frac{1}{3})^{\frac{1}{2}}$	1
	$(\frac{1}{3})^{\frac{1}{2}}$	1

Table 8.3 Location of integration points and weights for the Gauss-Lobatto quadrature rule

Number of integration points	Position η_i	Weight ω_i
3	-1	$\frac{1}{3}$
	0	$\frac{4}{3}$
	1	$\frac{1}{3}$
4	-1	$\frac{1}{6}$
	$-(\frac{1}{5})^{\frac{1}{2}}$	$\frac{5}{6}$
	$(\frac{1}{5})^{\frac{1}{2}}$	$\frac{5}{6}$
	1	$\frac{1}{6}$
5	-1	$\frac{1}{10}$
	$-(\frac{3}{7})^{\frac{1}{2}}$	$\frac{49}{90}$
	0	$\frac{32}{45}$
	$(\frac{3}{7})^{\frac{1}{2}}$	$\frac{49}{90}$
	1	$\frac{1}{10}$

Note that integration points at the bounds of the integration domain ($\eta = \pm 1$) are always included

Both quadrature rules can then be expressed as follows:

$$\frac{b-a}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}\eta + \frac{a+b}{2}\right) \mathrm{d}\eta \approx \frac{b-a}{2}\sum_{i=1}^{n} \omega_i f\left(\frac{b-a}{2}\eta_i + \frac{a+b}{2}\right), \tag{8.61}$$

where n is the number of *integration points*, $\eta_1, \ldots, \eta_n$ are the locations of the integration points within the integration domain, and $\omega_1, \ldots, \omega_n$ are the *weights* for each integration point.

The location of the integration points and the weights for the *Gauss-Legendre quadrature rule* are provided in Table 8.2. The reader is referred to other sources for the derivation of the location of the integration points and the weights [44, 45]. The Gauss-Legendre quadrature rule provides exact results for functions $f(x)$ given as polynomials of degree $2n - 1$. We only require 1 or 2 integration points for the integrals evaluated using the Gauss-Legendre quadrature rule in this chapter.

The location of the integration points and the weights for the *Gauss-Lobatto quadrature rule* are provided in Table 8.3. The Gauss-Lobatto quadrature rule always includes integration points at the bounds of the integration domain (i.e., at $\eta = \pm 1$) and provides exact results for functions $f(x)$ given as polynomials of degree $2n - 3$. We use 3, 4, or 5 integration points for the integrals evaluated using the Gauss-Lobatto quadrature rule in this chapter.

8.5 Examples of the Implemented Model

In this section, implementation results of the model derived in Sect. 8.4 are presented. The numerical procedure used for the simulation of the response of origami structures with smooth fold domains presented in Sect. 8.4.2 was implemented in MATLAB®. Different patterns of smooth folds are analyzed using the proposed model. For visualization purposes, only the mid-surface of the structures are plotted in all examples. The smooth fold surfaces $\mathcal{F}_t^i$ are visualized using the MATLAB three-dimensional surface plot function `surf` while the face surfaces $\mathcal{P}_t^i$ are visualized as filled three-dimensional polygons using `fill3`. Smooth folds exhibiting G^1 continuity are considered in all the examples shown in this section.

Finite element analysis is used for numerical validation of the proposed model. In this work, FEA is performed using Abaqus finite element software. Static analysis steps considering large rotations (i.e., the `NLGEOM` option in Abaqus is active, see [46, 47]) are used. Abaqus `S4` elements (4-node doubly curved general-purpose shells, finite membrane strains) are used to discretize the structures in all FEA examples shown in this section. To capture rigid behavior of the faces as assumed in the model derived in Sect. 8.4, rigid body constraints in Abaqus are imposed on finite elements located at the face regions.

Implementation examples considering smooth fold domains comprised of elastic and SMA materials are explored herein. Constitutive equations for such materials are provided in Appendix C. Material parameters for such materials are provided in Tables 8.4, 8.5, and 8.6. Consult Appendix C for the definition of the material parameters. It is remarked that the present structural model is applicable to smooth folds comprised of other passive or active materials provided strains are small enough to justify a linearized strain tensor.

8.5.1 Examples of Structures Having One Fold

The first example considers a structure having one linear elastic fold domain of 0.5 mm thickness and subject to a point load. The applied force vector is parallel

Table 8.4 Parameters for the aluminum material (linear elastic behavior assumed)

Parameter	Value
E	70.0 GPa
ν	0.3

Table 8.5 Parameters for the elastomer material (NinjaFlex®) based on those provided in [48] (linear elastic behavior assumed)

Parameter	Value
E	15.2 MPa
ν	0.4

Table 8.6 Parameters for the SMA material based on those provided in [46]

Parameter	Value
E^A	48.0 GPa
E^M	42.9 GPa
ν^A	0.3
ν^M	0.3
α	0.0 K^{-1}
M_s	332.2 K
M_f	302.2 K
A_s	337.2 K
A_f	362.2 K
C^A	9.5 MPa/K
C^M	9.0 MPa/K
$H_{\min}$	4.4%
H_{sat}	4.4%
$n_i, i = 1, 2, 3, 4$	1.0

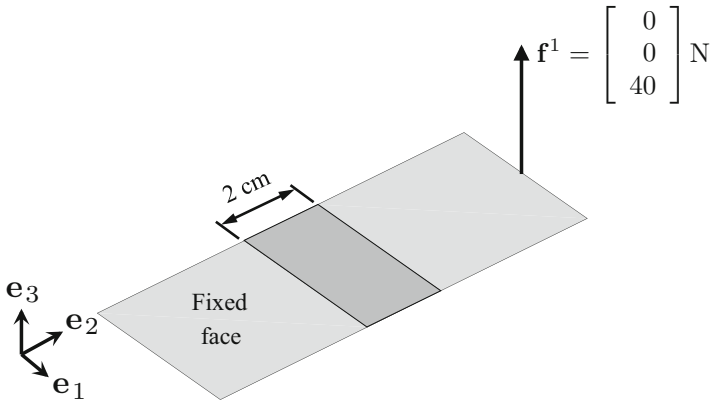


Fig. 8.5 Boundary value problem considering a structure having one smooth fold domain of thickness $\mathfrak{h}_i = 0.5$ mm. The material parameters of the elastic smooth fold domain are provided in Table 8.4

to the normal direction of the structure; therefore, it induces folding deformation. The boundary value problem is shown in Fig. 8.5 and the material parameters for the linear elastic smooth fold domain are provided in Table 8.4. Results from this example are shown in Fig. 8.6. A plot of the magnitude of the displacement vector at one of the points of maximum displacement vs. the magnitude of the applied point force vector is presented in Fig. 8.6b. Good agreement is observed between the smooth folds model developed here and FEA for the entire loading history.

The second example considers a boundary value problem with the same planar geometry as that shown in Fig. 8.5 but includes an SMA wires/elastomer composite layer in addition to elastic layers at the smooth fold domain. The boundary value

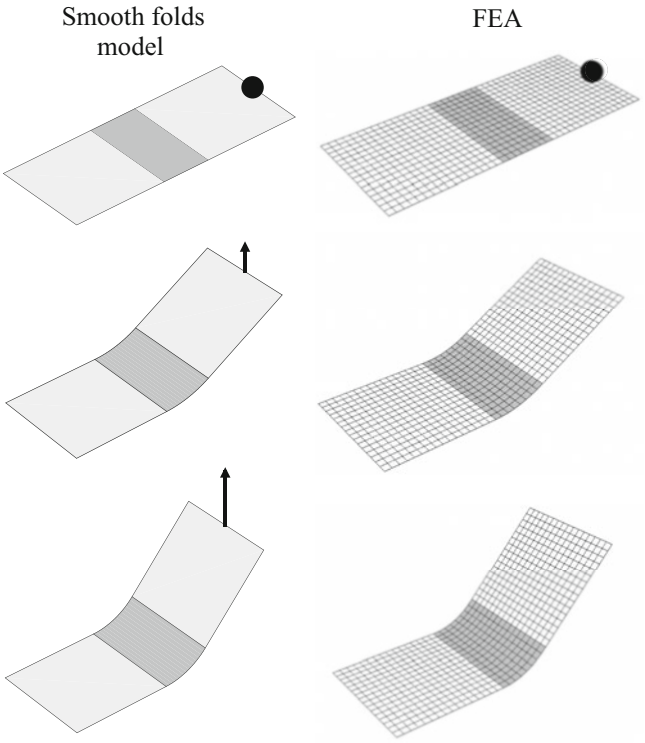


Fig. 8.6 Results associated with the boundary value problem of Fig. 8.5: (**a**) Configurations attained by the structure during the applied loading history (reference, intermediate, and final); (**b**) Magnitude of the displacement vector at one of the points of maximum displacement vs. magnitude of the applied point force vector

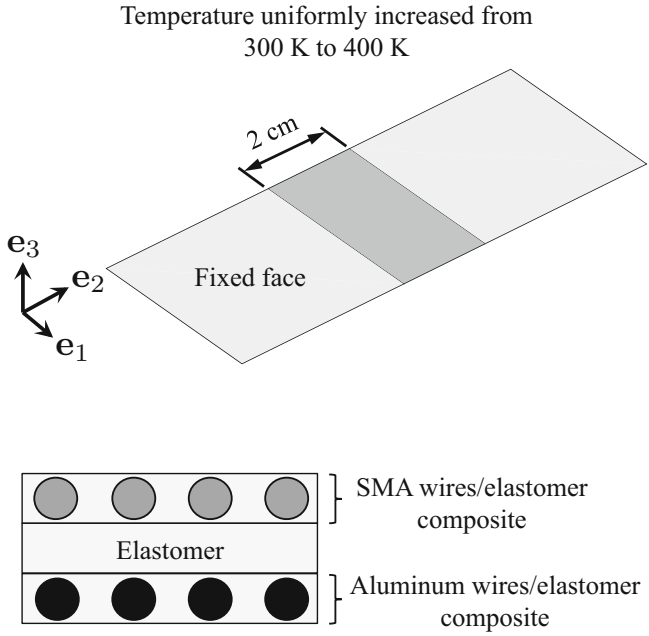


Fig. 8.7 Boundary value problem considering a structure having one smooth fold domain comprised of a three-layer laminate. The material parameters of the SMA, the elastomer, and aluminum are provided in Tables 8.6, 8.5, and 8.4, respectively

problem of this second example is shown in Fig. 8.7. The laminate comprising the smooth fold domain has three layers. The lower layer corresponds to a composite of aluminum wires embedded in an elastomer matrix with aluminum wire volume fraction $v_f = 0.01$. The upper layer corresponds to a composite of SMA wires embedded in an elastomer matrix with SMA wire volume fraction $v_f = 0.05$. The material parameters of the SMA wires, elastomer, and aluminum are provided in Tables 8.6, 8.5, and 8.4, respectively. We use the same volume fractions and material parameters for all the other examples considering SMA composites in this section.

Initially at a temperature of 300 K, the SMA is in a fully pre-strained martensitic phase. By increasing the temperature from 300 to 400 K, the SMA undergoes phase transformation from martensite to austenite and its transformation strain is recovered (because $A_s = 337.2$ K, refer to Table 8.6). Since the SMA wires/elastomer composite layer is placed off-center relative to the laminate mid-surface, the recovery of transformation strain induces folding deformation of the fold domain. The results for this problem are shown in Fig. 8.8. The contour plot in the simulation example of Fig. 8.8 denotes martensite volume fraction, which ranges from 1 (full martensite at low temperature) to 0 (full austenite at high temperature). It is observed from the plots of displacement vector magnitude at one of the points of maximum

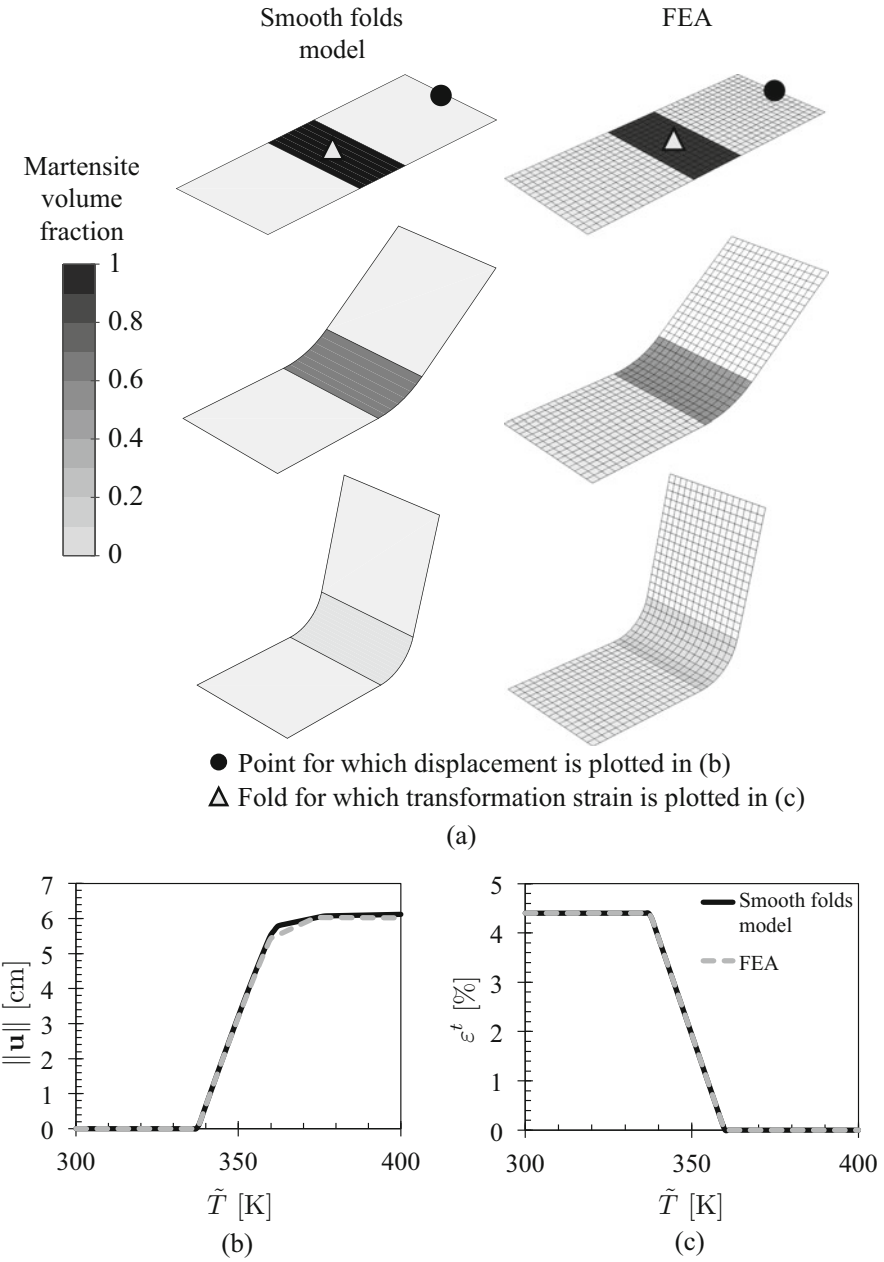


Fig. 8.8 Results associated with the boundary value problem of Fig. 8.7: (**a**) Configurations attained by the structure during the applied thermal history (reference, intermediate, and final). The martensite volume fraction contour plot of the SMA material is shown; (**b**) Magnitude of the maximum displacement vector vs. applied absolute temperature; (**c**) Transformation strain magnitude of the SMA material at the mid-point of the SMA wires/elastomer composite layer vs. applied absolute temperature

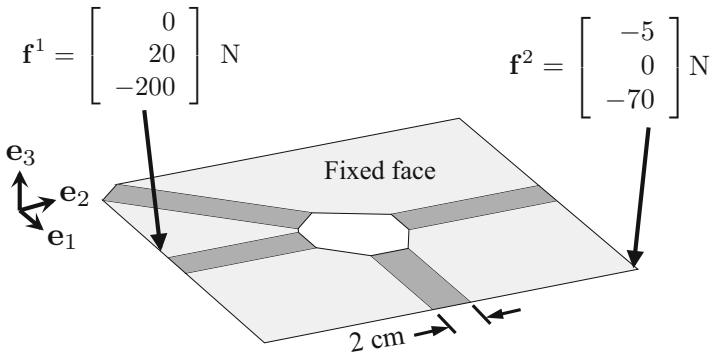


Fig. 8.9 Boundary value problem considering a structure having one fold intersection. The elastic smooth fold domains have thickness $\mathfrak{h}_i = 0.5$ mm. The material parameters of the elastic smooth fold domain are provided in Table 8.4

displacement vs. applied absolute temperature in Fig. 8.8b that the results obtained using the present model and those obtained from FEA are in good agreement. Also, there is good agreement between the present model and FEA regarding evolution of transformation strain magnitude at the mid-point of the SMA wires/elastomer composite layer as shown in Fig. 8.8c.

8.5.2 Examples of Structures Having One Fold Intersection

A more complex example considers a structure having four elastic fold domains meeting at a single fold intersection. The boundary value problem associated with this example is shown in Fig. 8.9. Two point loads are applied to the structure as shown in Fig. 8.9. Unlike the two previous examples shown in Figs. 8.5 and 8.7, the kinematic constraints (5.66) and (5.67) must be considered since a fold intersection is present in the structure. The results for this example are shown in Fig. 8.10. It is observed that the results between the smooth folds model and FEA are also in good agreement for this example of a structure having a fold intersection.

An example of a structure with one fold intersection that morphs via SMA actuation is shown in Fig. 8.11. It is noted that the location of the SMA wires/elastomer composite layer (top or bottom) determines the intended direction of the resulting fold (valley or mountain, respectively). Upon increasing the temperature from 300 to 400 K, the SMA material in the smooth fold domains actuate and induce folding of the structure as shown in Fig. 8.12. The deformed configuration plots in Fig. 8.12a show that there is good agreement between the smooth folds model and FEA also for this example. Good agreement is also observed in the plot of displacement vector magnitude at the point of maximum displacement vs. applied absolute temperature

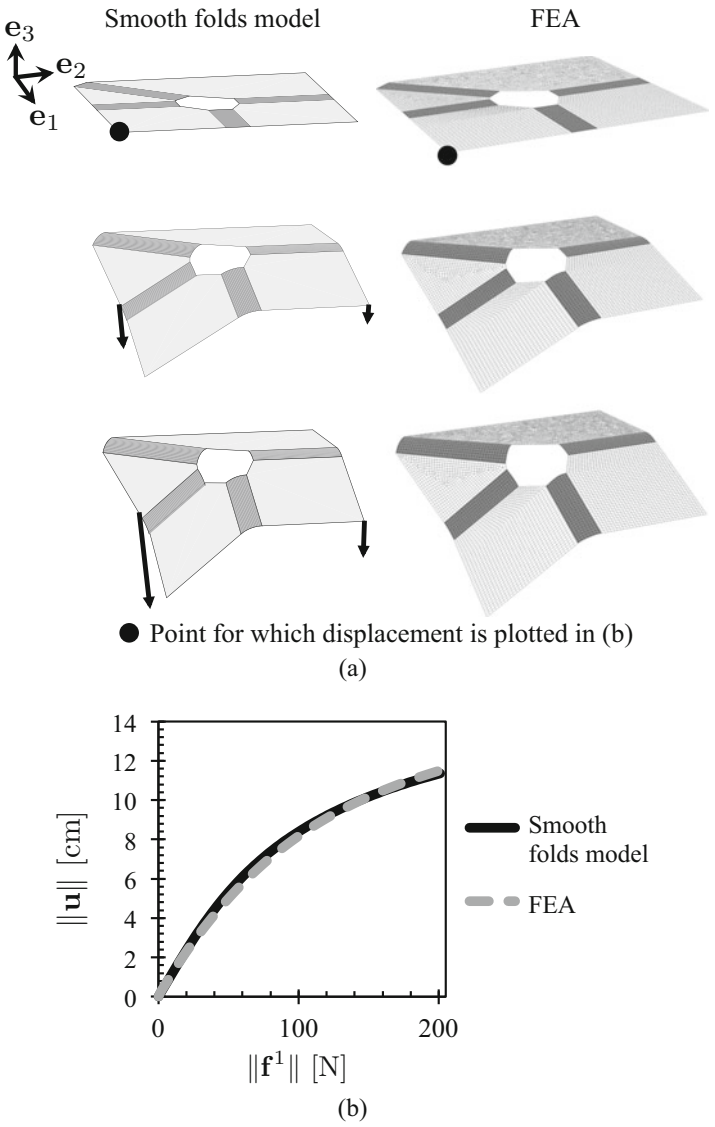


Fig. 8.10 Results associated with the boundary value problem of Fig. 8.9: (**a**) Configurations attained by the structure during the applied loading history (reference, intermediate, and final); (**b**) Magnitude of the maximum displacement vs. magnitude of applied point force vector

in Fig. 8.12b. The plot of transformation strain magnitude at the mid-point of the SMA wires/elastomer composite layer in the smooth fold exhibiting the largest fold angle vs. applied absolute temperature in Fig. 8.12c also shows good agreement between the models.

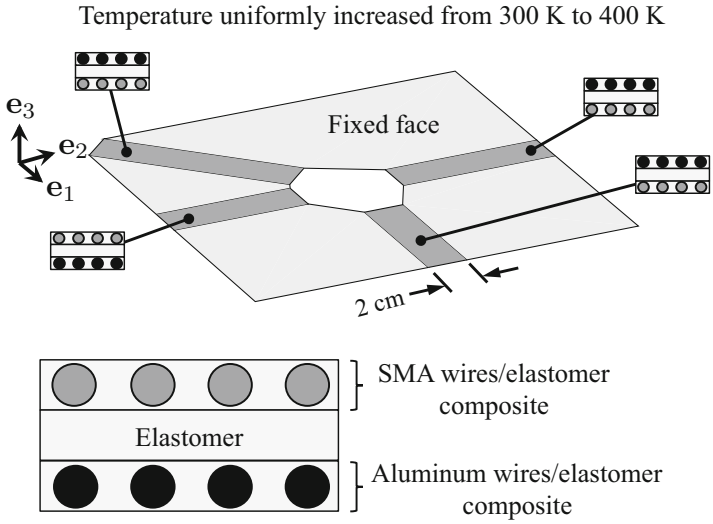


Fig. 8.11 Boundary value problem considering a structure having one fold intersection and smooth fold domains comprised of a three-layer laminate. The laminate layup for each individual smooth fold domain are shown. The material parameters of the SMA, the elastomer, and aluminum are provided in Tables 8.6, 8.5, and 8.4, respectively

8.5.3 Examples of Structures Having Multiple Fold Intersections

Figure 8.13 shows the boundary value problem of a structure having multiple fold intersections. The smooth fold domains in this example are all linear elastic having material parameters provided in Table 8.4. The central face is fixed and four point loads are applied. The direction and maximum magnitude of the four point loads are equal. The results for this example are shown in Fig. 8.14. Figure 8.14a shows configurations attained by the structure during the applied loading history (reference and final) for both the smooth folds model and FEA. Figure 8.14b shows a plot of displacement vector magnitude at one of the points of maximum displacement vs. magnitude of applied point force vector. It is observed from the results in Fig. 8.14a, b that there is good agreement between FEA and the smooth folds model for this more complex example.

The last example considers the planar geometry shown in Fig. 8.13 but includes smooth folds comprised of laminates having an SMA wires/elastomer composite layer. The boundary value problem for this example is shown in Fig. 8.15. The central face of the origami structure is fixed and the temperature at four folds is increased from 300 to 400 K to induce folding deformation in the structure. The

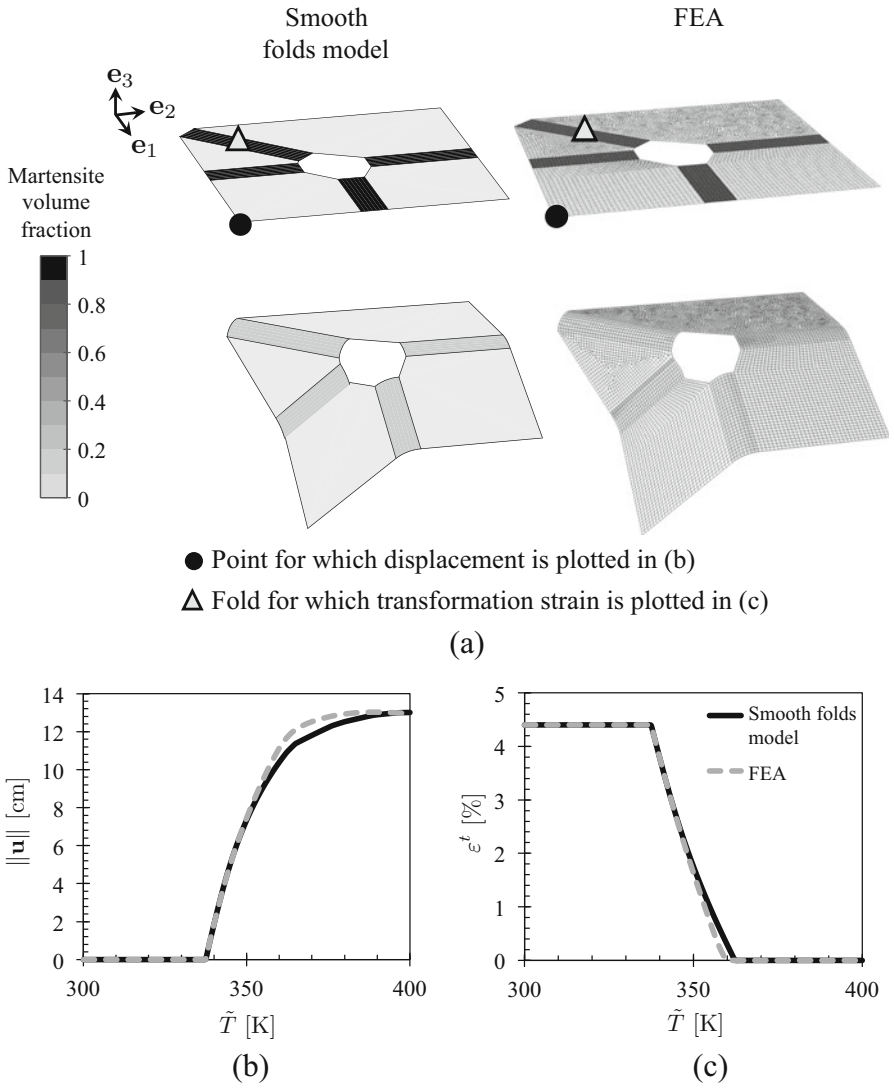
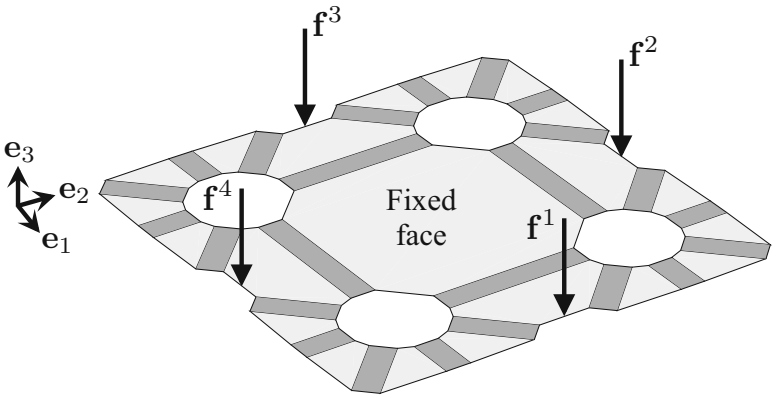


Fig. 8.12 Results associated with the boundary value problem of Fig. 8.11: (**a**) Configurations attained by the structure during the applied thermal history (reference and final). The martensite volume fraction contour plot of the SMA material is shown; (**b**) Magnitude of the maximum displacement vs. applied absolute temperature; (**c**) Transformation strain magnitude of the SMA material at the mid-point of the SMA wires/elastomer composite layer vs. applied absolute temperature



$$\mathbf{f}^1 = \mathbf{f}^2 = \mathbf{f}^3 = \mathbf{f}^4 = \begin{bmatrix} 0 \\ 0 \\ -200 \end{bmatrix} \text{ N}$$

Fig. 8.13 Boundary value problem considering a structure having multiple fold intersections. The elastic smooth fold domains have thickness $\mathfrak{h}_i = 0.5$ mm. The material parameters of the elastic smooth fold domain are provided in Table 8.4

results for this example are presented in Fig. 8.16. Just as in the previous examples, the deformation determined using the proposed model is in good agreement with that determined using FEA.

8.5.4 Computational Efficiency Comparison

The computational efficiency of the proposed model for origami structures with smooth fold domains as compared to FEA is also explored. Tables 8.7 and 8.8, respectively, provide the number of kinematic variables of each boundary value problem considering SMA behavior shown in Figs. 8.7, 8.11, and 8.15. The kinematic variables in the present model for origami with smooth folds correspond to the fold angles and fold arc-lengths of each smooth fold in the structure (i.e., a total of $2N_{\mathcal{F}}$ kinematic variables). The kinematic variables in the FEA models correspond to the displacements and rotations at all the nodes in the shell elements. It is observed in Table 8.7 that the present model for origami with smooth folds significantly reduces the number of kinematic variables required to determine the configuration of the considered structures. Table 8.8 shows the wall-clock time in seconds required for the simulation of each boundary value problem considering SMA behavior. All the simulations presented in this chapter (using the smooth

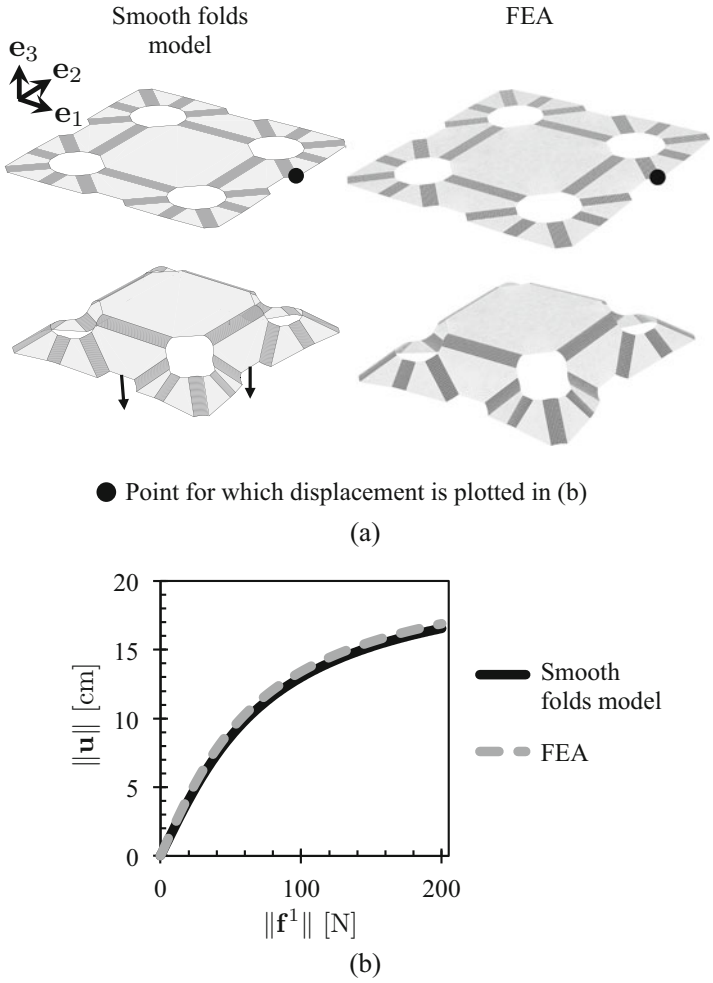


Fig. 8.14 Results associated with the boundary value problem of Fig. 8.13: (**a**) Configurations attained by the structure during the applied loading history (reference and final); (**b**) Magnitude of the maximum displacement vs. magnitude of applied point force vector

folds model and FEA) were performed on the same computer. The wall-clock time required for analysis using the proposed model is significantly lower than that required using FEA for all the problems. It is noted that no particular efforts to improve the computational efficiency of the implemented model were considered; therefore, the wall-clock time for the simulations performed using the present model can be further reduced if such efforts are considered in future work.

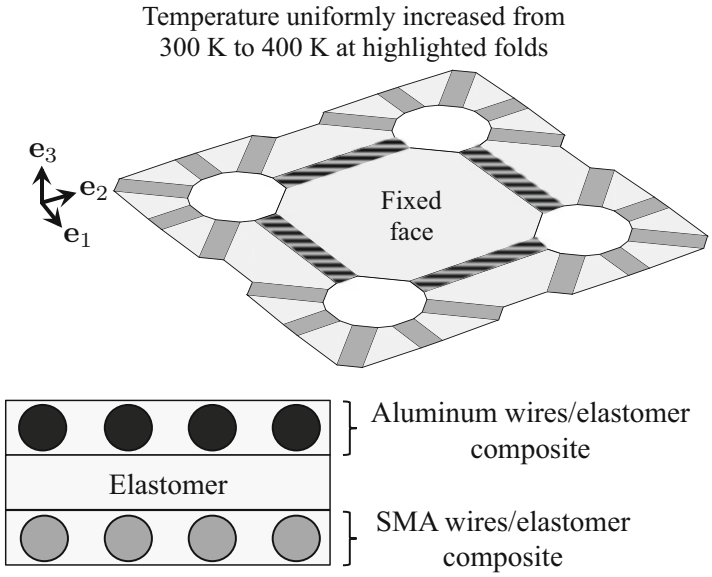


Fig. 8.15 Boundary value problem considering a structure having multiple fold intersections and smooth fold domains comprised of a three-layer laminate. The folds at which the temperature is uniformly increased are highlighted. The material parameters of the SMA, the elastomer, and aluminum are provided in Tables 8.6, 8.5, and 8.4, respectively

8.6 Unfolding Polyhedra Method for the Design of Self-Folding Structures

In this section, the *unfolding polyhedra method* for the design of origami structures with smooth folds presented in Chap. 6 is extended beyond an initial kinematics-based approach to consider the design of self-folding structures morphed through active material actuation.

In the present design method, the laminate layups (i.e., thickness and constituent material for each layer in the laminate) for each smooth fold domain are given. These can be determined prior to the implementation of the present method by optimizing layer thicknesses and other design variables allowing for higher curvatures at the smooth folds under material failure constraints. Such efforts are not discussed herein but are addressed throughout the literature [15, 16, 49, 50]. The design problem statement considered here is the following:

- *Given*: A goal three-dimensional shape represented as a polygonal mesh (the *goal mesh* $\mathcal{M}$) and laminate layups for the smooth fold domains in the structure,
- *Find*: The geometry of the mid-surface configuration $\mathcal{S}_0$ of a planar origami structure (Fig. 8.17) that morphs towards a configuration that approximates $\mathcal{M}$ via active material-driven folding (Fig. 8.18).

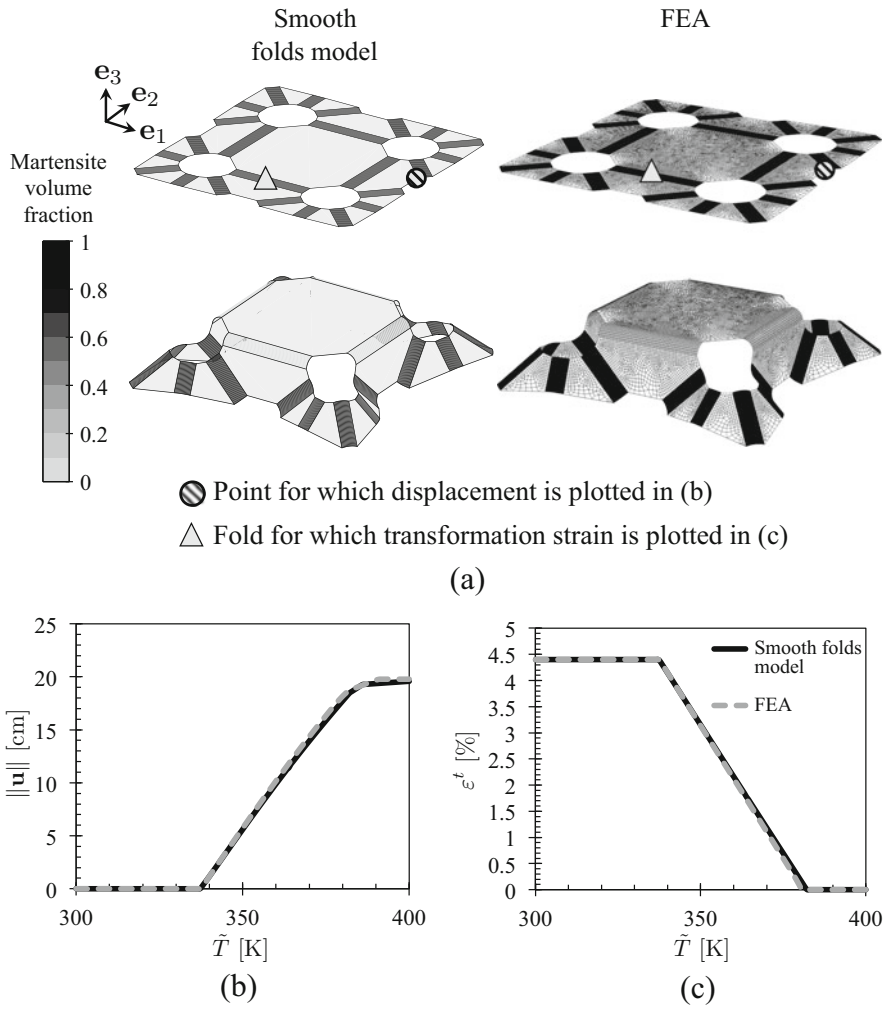


Fig. 8.16 Results associated with the boundary value problem of Fig. 8.15: (**a**) Configurations attained by the structure during the applied thermal history (reference and final). The martensite volume fraction contour plot of the SMA material is shown; (**b**) Magnitude of the maximum displacement vs. applied absolute temperature; (**c**) Transformation strain magnitude of the SMA material at the mid-point of the SMA wires/elastomer composite layer vs. applied absolute temperature

The unfolding polyhedra method for origami with smooth folds presented in Chap. 6 is used to solve the former origami design problem. However, such a method must be extended to account for the folding deformation feasibly achievable by smooth folds of specified thickness and constituent materials. The critical extension in the current framework is the consideration of the *fold widths* as design variables

Table 8.7 Number of model kinematic variables required for each example considering SMA behavior

Example	Number of kinematic variables		
	FEA	Smooth folds model	Ratio
Single fold example (Fig. 8.7)	4182	2	2091:1
Single fold intersection example (Fig. 8.11)	39648	8	4956:1
Multiple fold intersections example (Fig. 8.15)	97008	24	4042:1

Table 8.8 Simulation wall-clock time in seconds required for each example considering SMA behavior

Example	Wall-clock time (seconds)		
	FEA	Smooth folds model	Ratio
Single fold example (Fig. 8.7)	60	3.6	16.7:1
Single fold intersection example (Fig. 8.11)	2418	82	29.5:1
Multiple fold intersections example (Fig. 8.15)	6215	475	13.1:1

because they dictate the fold angle achieved by the active smooth folds under imposition of an activation field. To clarify, Fig. 8.19 shows simulation results of a structure having a single smooth fold with the laminate layup shown in Fig. 8.7. It is observed that by increasing the fold width $\hat{w}_i^0$, a higher fold angle magnitude can be achieved when the shape memory alloy (SMA) material is fully transformed from pre-strained martensite into austenite. Therefore, the fold widths are considered as design variables in this method. This represents an improvement over the kinematics-based method addressed in Chap. 6 where the fold widths were assumed given.

The vector $\hat{\boldsymbol{w}}^0 \in \mathbb{R}^{N_\mathcal{F}}$, where $N_\mathcal{F}$ is the total number of folds in the designed structure ($N_\mathcal{F} = 5$ in Fig. 8.17), is constructed by collecting the fold widths $\hat{w}_i^0$, $i = 1, \ldots, N_\mathcal{F}$, as follows:

$$\hat{\boldsymbol{w}}^0 = [\hat{w}_1^0 \ \cdots \ \hat{w}_{N_\mathcal{F}}^0]^\top. \tag{8.62}$$

The optimization problem associated with this design framework is stated as follows:

$$\begin{aligned} &\text{Find} && \hat{\boldsymbol{w}}^0 \\ &\text{That minimizes} && \|\hat{\boldsymbol{\theta}}^{t_f} - \hat{\boldsymbol{\theta}}^\star\| \\ &\text{Subject to} && \hat{\boldsymbol{w}}_L^0 \leq \hat{\boldsymbol{w}}^0 \leq \hat{\boldsymbol{w}}_U^0 \\ & && \text{Material failure constraints} \\ & && \text{Problem-dependent constraints,} \end{aligned} \tag{8.63}$$

where $\hat{\boldsymbol{\theta}}^{t_f} \in \mathbb{R}^{N_\mathcal{F}}$ is the vector whose components correspond to the fold angles at the final configuration attained by the active origami structure (e.g., the fold

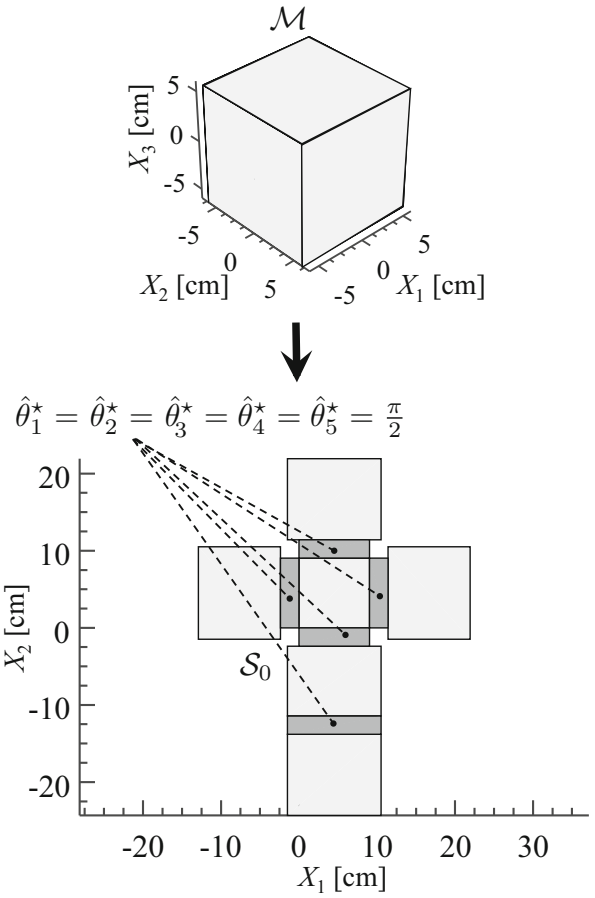


Fig. 8.17 Schematic of the unfolding polyhedra method for the design of active origami structures: Given a goal mesh $\mathcal{M}$ (cube), find the mid-surface reference configuration $\mathcal{S}_0$ of an origami structure that approximates $\mathcal{M}$ in a folded configuration. $\mathcal{S}_0$ is comprised of the faces of $\mathcal{M}$ mapped to the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$ and introduced smooth folds at each interior edge. The goal fold angles for each fold ($\hat{\theta}_1^\star, \ldots, \hat{\theta}_5^\star$) are indicated. Note that all the fold widths are equal because all the goal fold angles are equal and no externally applied loads are considered

angles of an SMA-based origami structure at full transformation from pre-strained martensite into austenite of the SMA material in its folds). The vector $\hat{\boldsymbol{\theta}}^\star \in \mathbb{R}^{N_\mathcal{F}}$ has components corresponding to the fold angles of the goal configuration $\mathcal{S}_\star$ (see Fig. 8.18). The components of the vectors $\hat{\boldsymbol{w}}_L^0, \hat{\boldsymbol{w}}_U^0 \in \mathbb{R}^{N_\mathcal{F}}$ correspond to the lower and upper bounds, respectively, for the fold width of each smooth fold.

A flowchart showing the steps for the optimization and design evaluation processes in the unfolding polyhedra method for the design of self-folding structures is provided in Fig. 8.20. The optimization problem (8.63) is solved here using the gradient-based constrained optimization algorithm `fmincon` in MATLAB [51]. Each design evaluation is performed using the structural mechanics model presented previously in this chapter. The laminate layup shown in Fig. 8.19 is assumed in all the examples. The material parameters of the aluminum, the elastomer, and the SMA of such a laminate layup are provided in Tables 8.4, 8.5, and 8.6, respectively. The orientation of the laminate layup (i.e., whether the SMA wires/elastomer composite

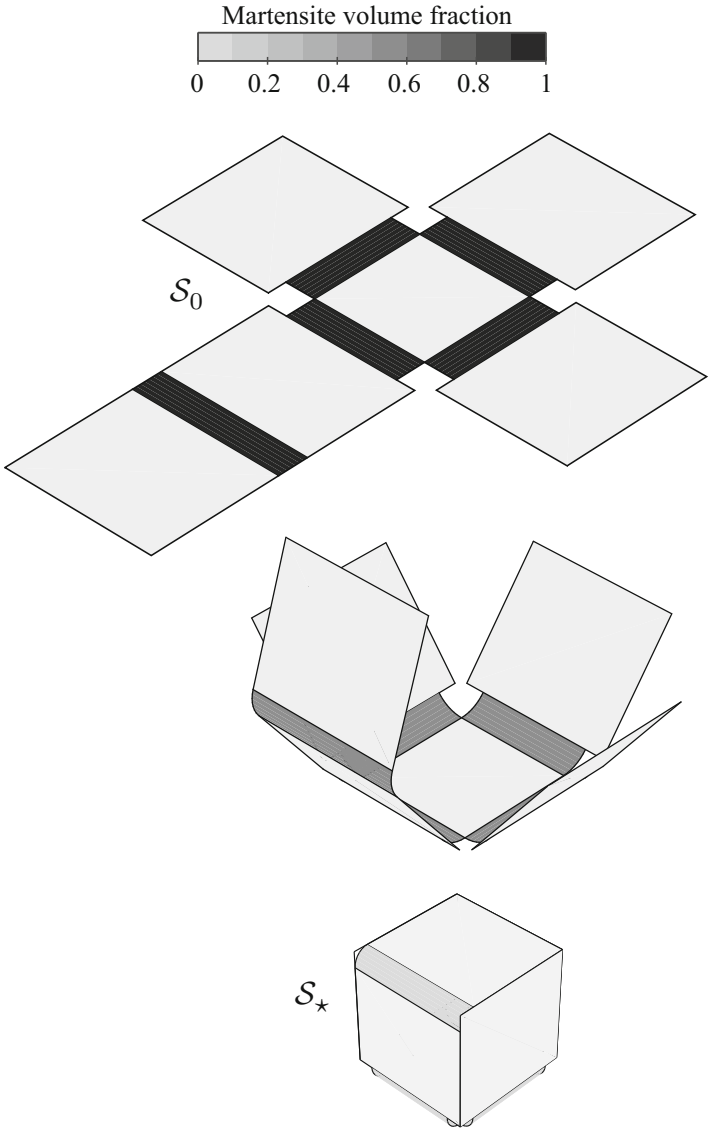


Fig. 8.18 Self-folding of a designed SMA-based origami structure (refer to Fig. 8.17) from the planar reference configuration to the configuration that approximates its associated goal mesh. The temperature is increased uniformly in the structure and all folds are simultaneously active

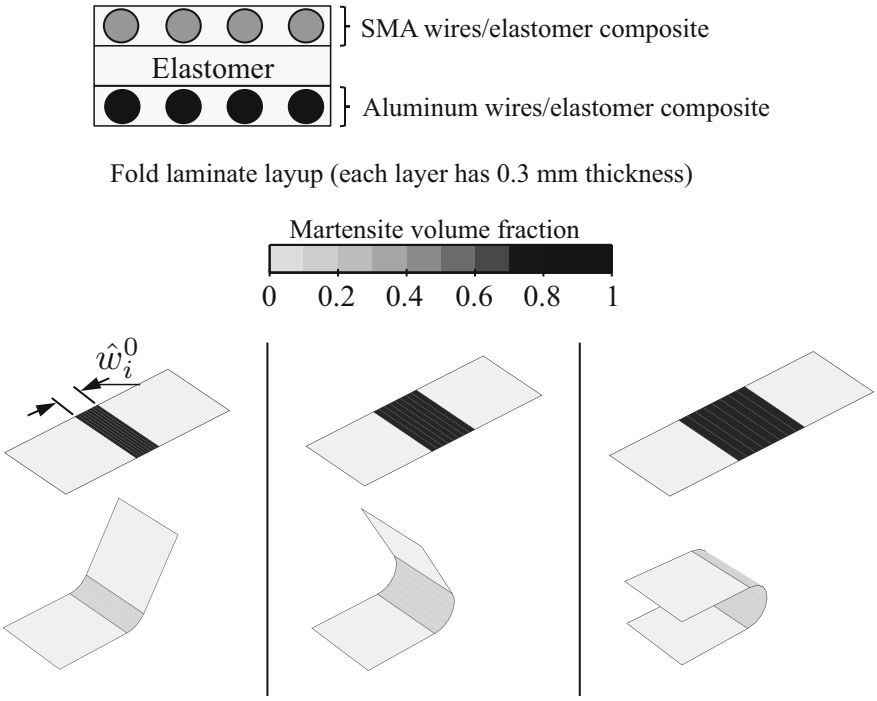


Fig. 8.19 Simulation results of an origami structure having a single smooth fold. The laminate layup is also shown. By increasing the fold width $\hat{w}_i^0$, a higher fold angle magnitude can be achieved when the SMA material is fully transformed from martensite into austenite

layer is located on the top or the bottom of the laminate) is dictated by the sign of the goal fold angle. We do not consider externally applied mechanical loads in the presented examples. The temperature is increased from 300 to 400 K in all the examples, leading to phase transformation in the SMA wires of the laminate shown in Fig. 8.19.

A result obtained using the implemented unfolding polyhedra method for the design of self-folding structures is shown in Fig. 8.21. Unlike the cube shown in Fig. 8.17, not all the goal fold angles in the structure are equal as indicated in the figure; therefore, distinct fold widths are required for the smooth folds in the designed sheet. The simulated folding motion for this example is shown in Fig. 8.22.

More complex examples of design of self-folding structures using the unfolding polyhedra method are shown in Figs. 8.23, 8.24, 8.25, and 8.26, where goal meshes associated with a regular octahedron, a non-regular octahedron, a regular icosahedron, and an ellipsoid are presented, respectively.

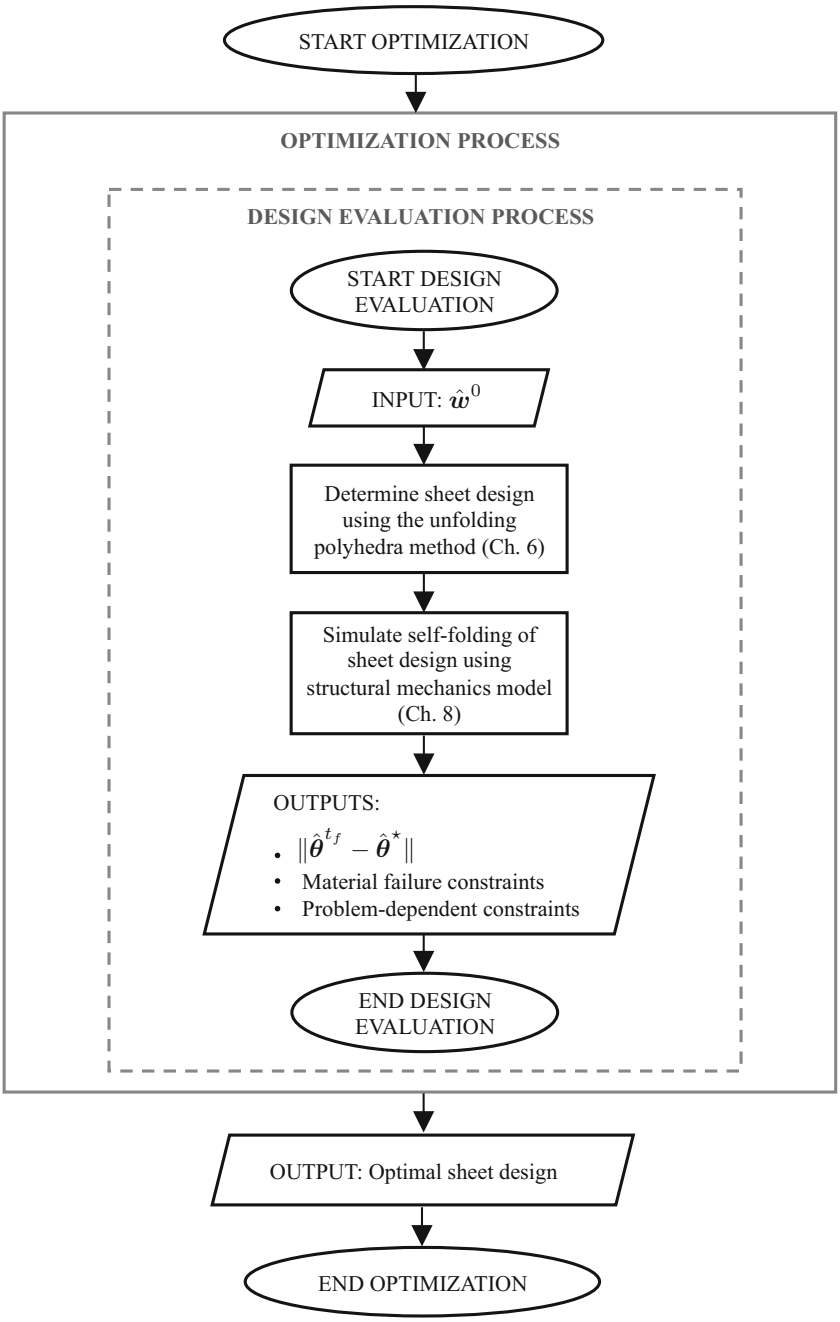


Fig. 8.20 Flowchart showing the steps of the optimization and design evaluation processes in the unfolding polyhedra method for the design of self-folding structures

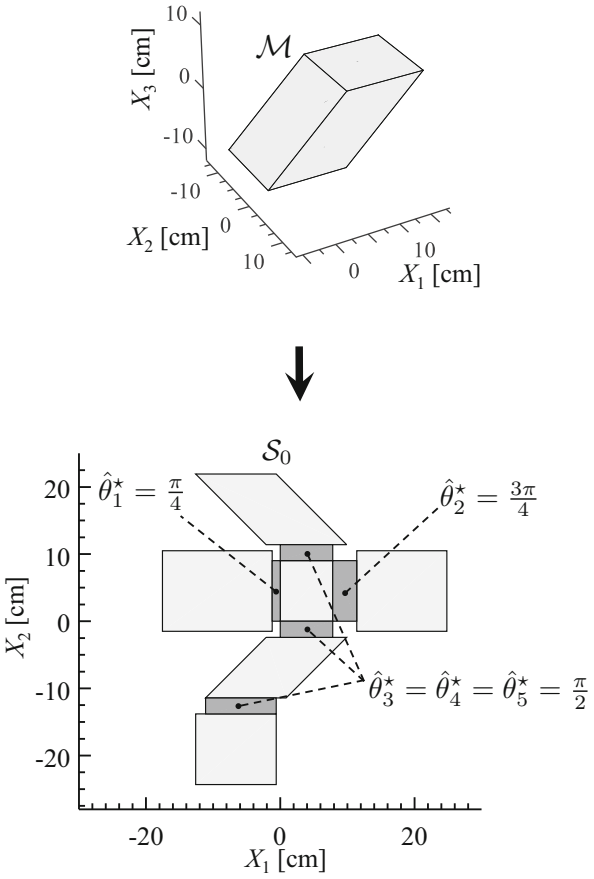


Fig. 8.21 Implementation example of the unfolding polyhedra method for the design of active origami structures: Given a goal mesh $\mathcal{M}$ (parallelepiped), find the mid-surface reference configuration $\mathcal{S}_0$ of an origami structure that approximates $\mathcal{M}$ in a folded configuration. The goal fold angles for each fold $(\hat{\theta}_1^\star, \ldots, \hat{\theta}_5^\star)$ are indicated. Note that larger fold widths are required to achieve higher fold angles

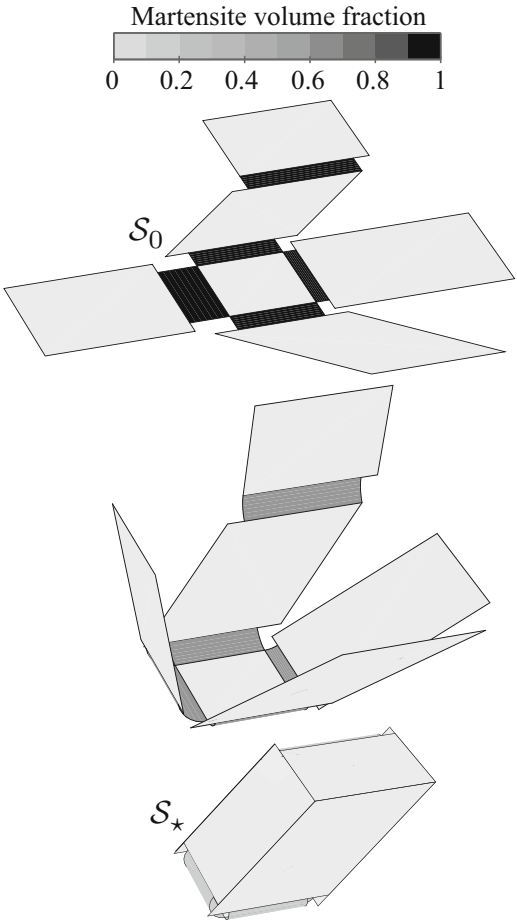


Fig. 8.22 Self-folding of a designed SMA-based origami structure (refer to Fig. 8.21) from the planar reference configuration to the configuration that approximates its associated goal mesh. The temperature is increased uniformly in the structure and all folds are simultaneously active

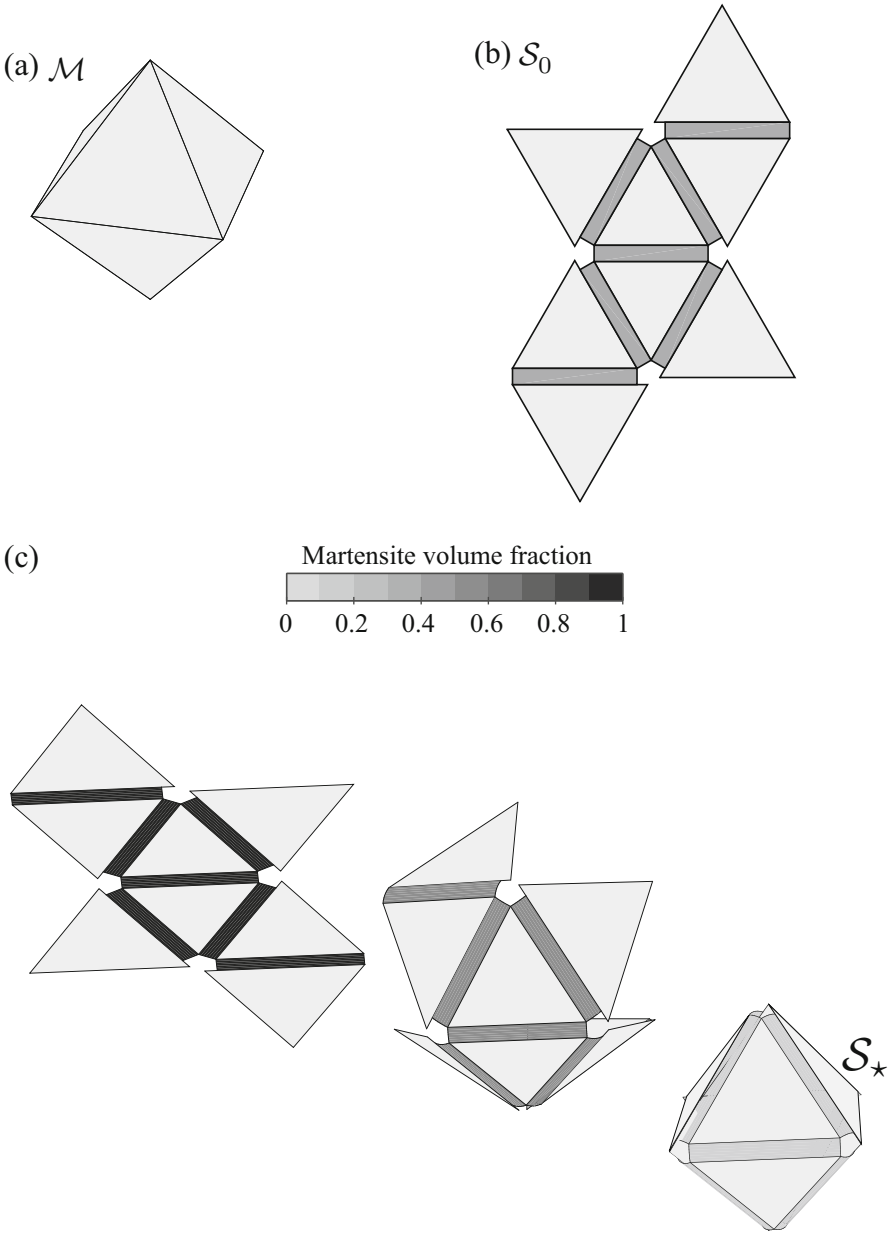


Fig. 8.23 Results of the implemented unfolding polyhedra method for active origami structures associated with a goal mesh of a regular octahedron: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths. All the fold widths are equal because all the folds have associated goal fold angles of $\pi - \cos^{-1}(-\frac{1}{3})$ radians (approximately 70.53°); (**c**) Configurations attained by the self-folding structure during its folding history

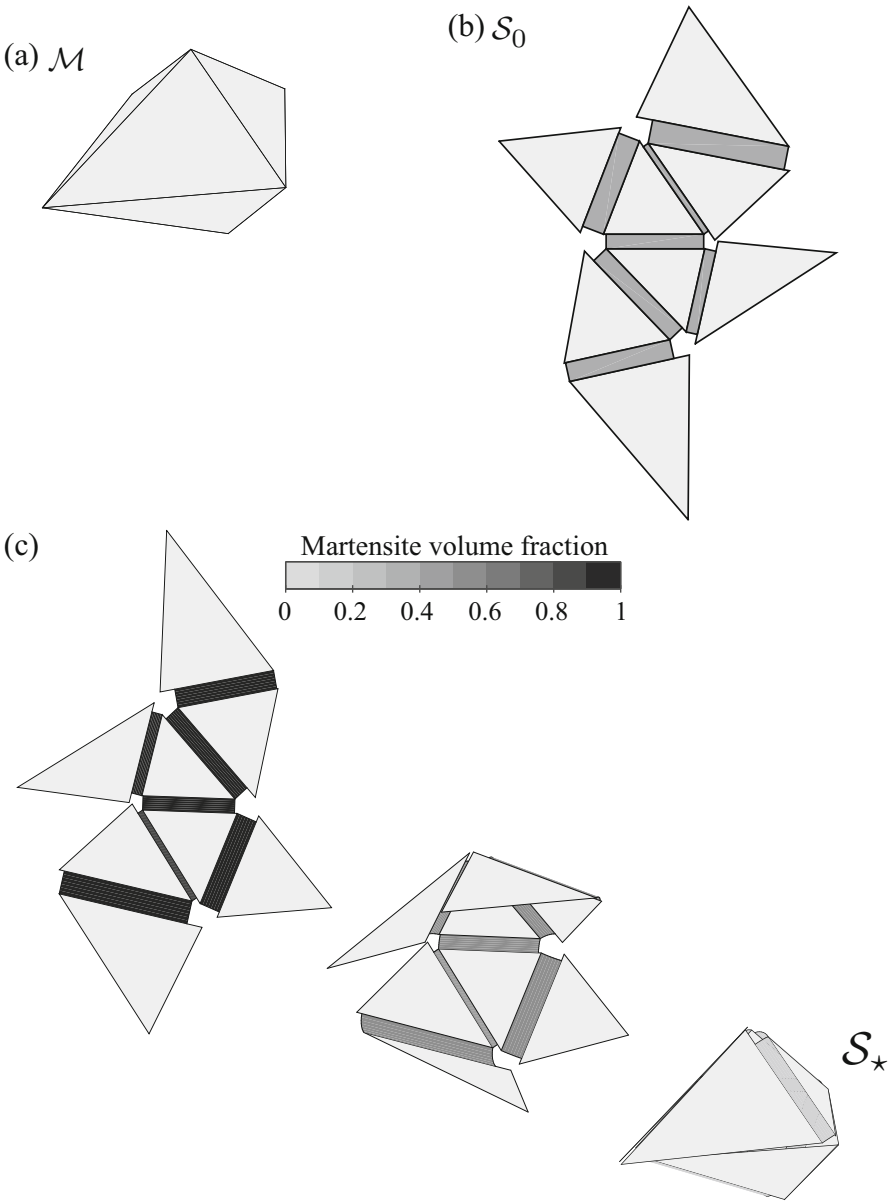


Fig. 8.24 Results of the implemented unfolding polyhedra method for active origami structures associated with a goal mesh of an octahedron: (**a**) Goal mesh generated by randomly perturbing the coordinates of every node of the regular octahedron (Fig. 8.23a); (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the self-folding structure during its folding history

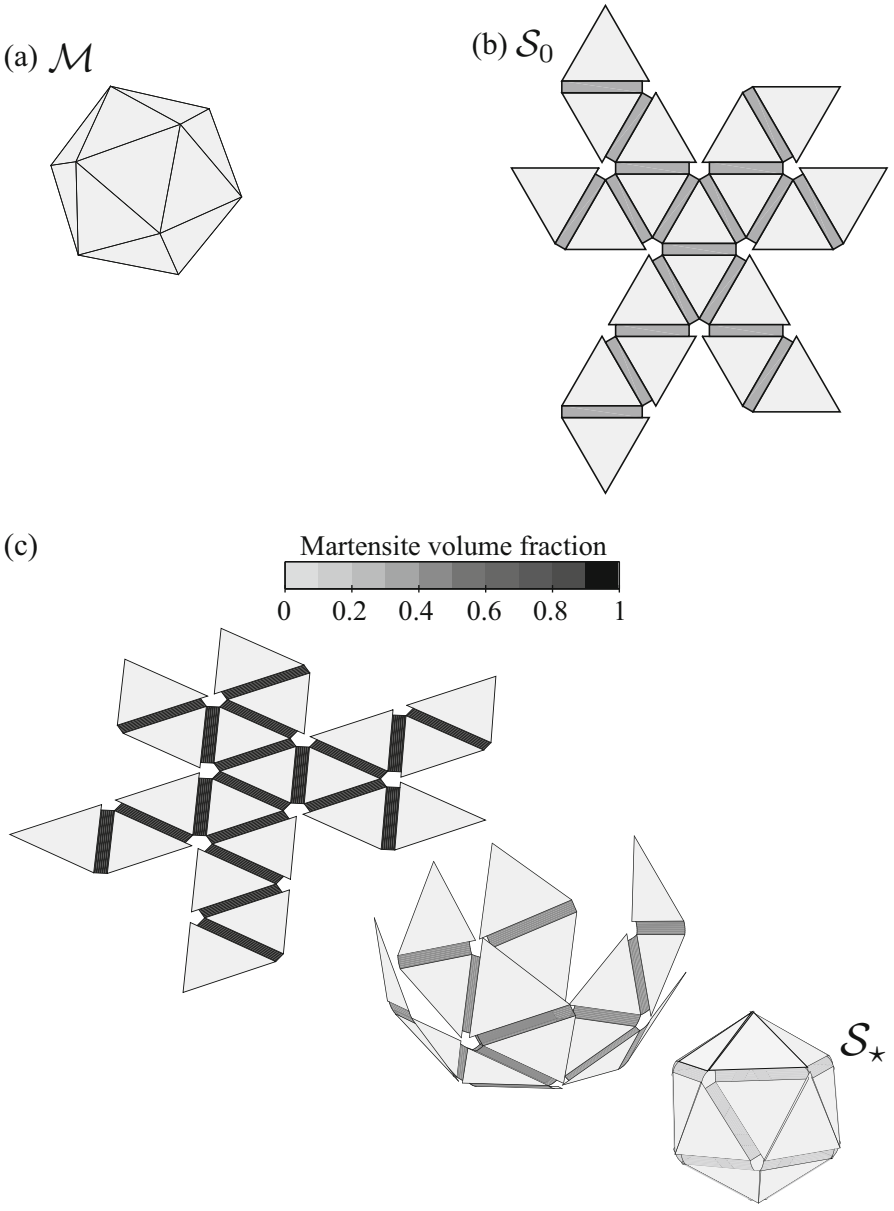


Fig. 8.25 Results of the implemented unfolding polyhedra method for active origami structures associated with a goal mesh of a regular icosahedron: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths. All the fold widths are equal because all the folds have associated goal fold angles of $\pi - \cos^{-1}(-\frac{5^{1/2}}{3})$ radians (approximately 41.81°); (**c**) Configurations attained by the self-folding structure during its folding history

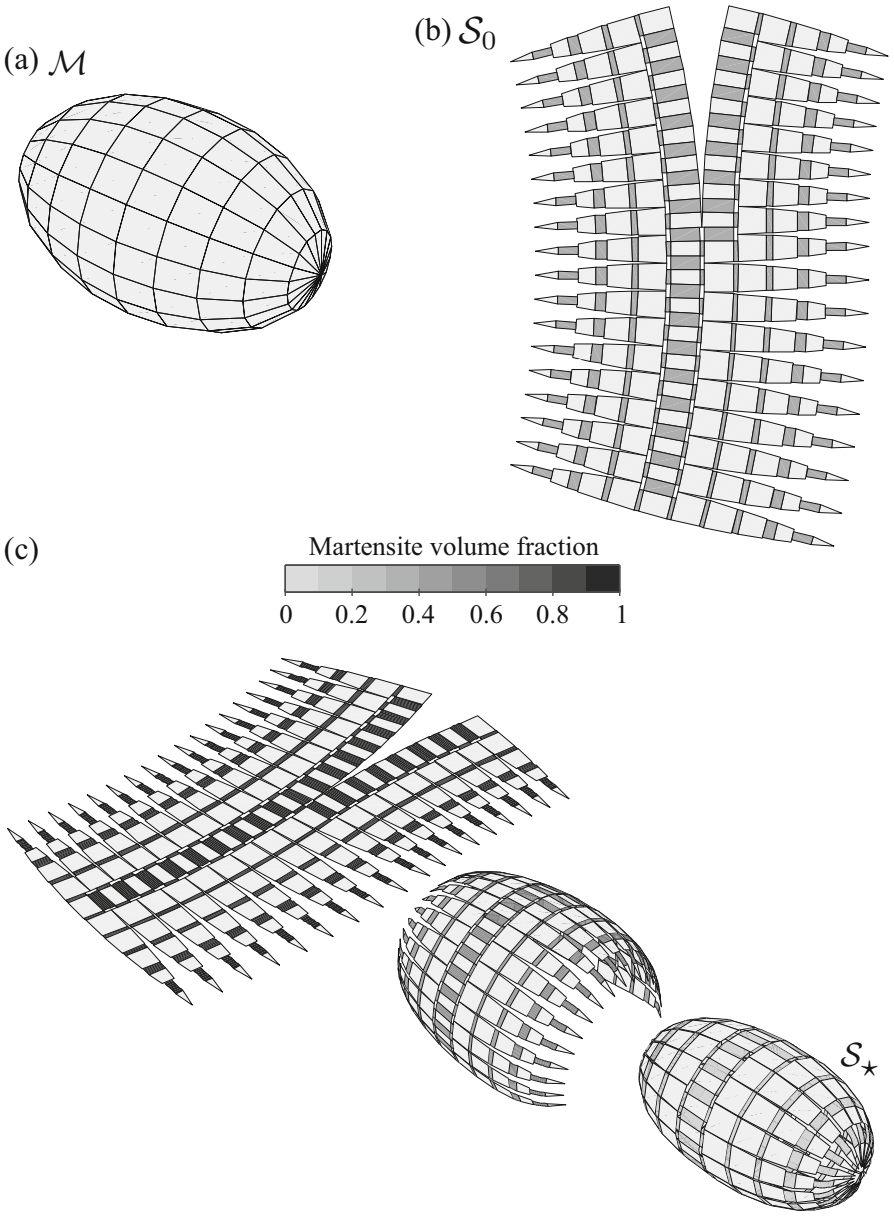


Fig. 8.26 Results of the implemented unfolding polyhedra method for active origami structures associated with a goal mesh of an ellipsoidal shape: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the self-folding structure during its folding history

8.7 Tuck-Folding Method for the Design of Self-Folding Structures

In this section, the *tuck-folding method* for the design of origami structures with smooth folds presented in Chap. 7 (based on kinematics only) is extended to consider the design of self-folding structures morphed through active material actuation. As in Sect. 8.6, the laminate layups for each smooth fold domain are assumed given. The design problem statement considered here is the following:

- *Given*: A goal three-dimensional shape represented as a polygonal mesh (termed as the *goal mesh* $\mathcal{M}$) and laminate layups for the smooth fold domains in the structure,
- *Find*: The geometry of the mid-surface configuration $\mathcal{S}_0$ of a planar origami structure (Fig. 8.27) that morphs towards a configuration approximating $\mathcal{M}$ via active material-driven folding (Fig. 8.28).

The tuck-folding method for origami with smooth folds presented in Chap. 7 is used to solve the aforementioned origami design problem. As in Sect. 8.6, we extend the tuck-folding method from Chap. 7 to consider the *fold widths* as design variables because they dictate the fold angle achieved by the active smooth fold domains under imposition of an activation field (see Fig. 8.19).

The geometry of each *edge module* in a sheet designed using the tuck-folding method (refer to Sect. 7.2.1) is defined by five variables corresponding to the length variable $\hat{W}_i$, the angular variable $\hat{\psi}_i$, and the width of the three smooth folds in the edge module $\hat{w}^0_{3i-2}$, $\hat{w}^0_{3i-1}$, and $\hat{w}^0_{3i}$, $i = 1, \ldots, N^I_{\mathcal{E}}$ (refer to Fig. 7.5a), where $N^I_{\mathcal{E}}$ is the number of interior edges of the goal mesh $\mathcal{M}$. The vectors $\hat{\boldsymbol{W}}, \hat{\boldsymbol{\psi}} \in \mathbb{R}^{N^I_{\mathcal{E}}}$ are constructed by collecting the design variables $\hat{W}_i$ and $\hat{\psi}_i$, $i = 1, \ldots, N^I_{\mathcal{E}}$, respectively, as follows:

$$\hat{\boldsymbol{W}} = [\hat{W}_1 \ \cdots \ \hat{W}_{N^I_{\mathcal{E}}}]^\top, \qquad \hat{\boldsymbol{\psi}} = [\hat{\psi}_1 \ \cdots \ \hat{\psi}_{N^I_{\mathcal{E}}}]^\top, \tag{8.64}$$

and the vector $\hat{\boldsymbol{w}}^0 \in \mathbb{R}^{3N^I_{\mathcal{E}}}$ is constructed by collecting the design variables $\hat{w}^0_i$, $i = 1, \ldots, 3N^I_{\mathcal{E}}$, as follows:

$$\hat{\boldsymbol{w}}^0 = [\hat{w}^0_1 \ \cdots \ \hat{w}^0_{3N^I_{\mathcal{E}}}]^\top. \tag{8.65}$$

The constraints for these design variables allowing for a valid mid-surface reference configuration $\mathcal{S}_0$ are derived in Sect. 7.2.3 [52]. These constraints can be grouped into a vector equation of equality constraints $\boldsymbol{h} = \boldsymbol{0}$ and a vector equation of inequality constraints $\boldsymbol{g} \leq \boldsymbol{0}$ (refer to Sect. 4.2.5). The optimization problem associated with this design framework is then stated as follows:

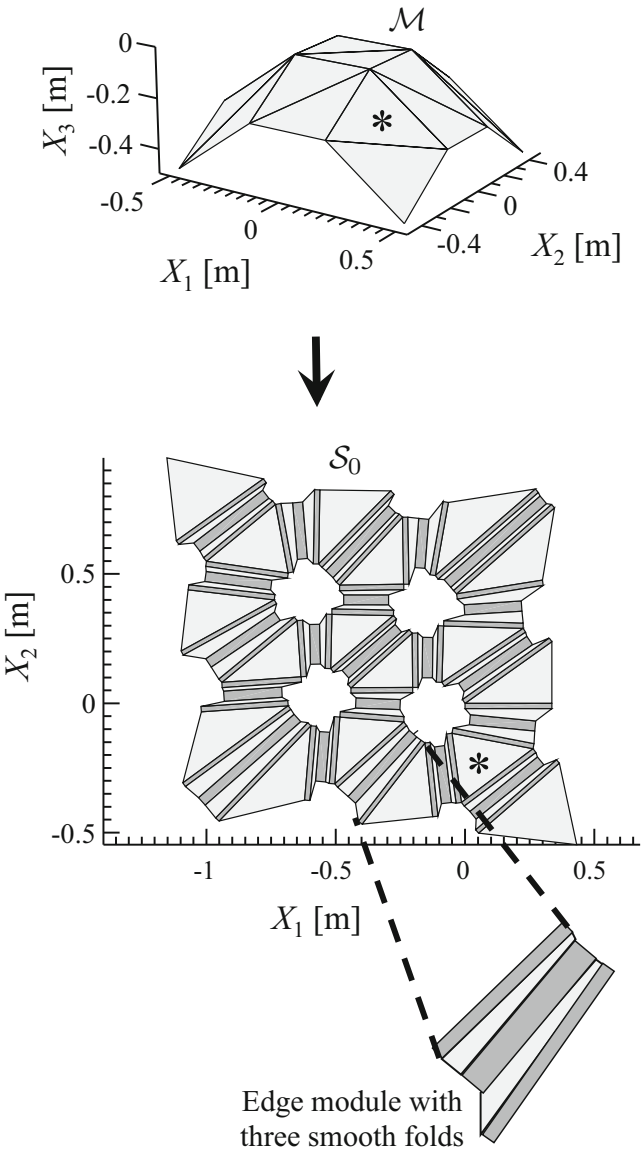


Fig. 8.27 Schematic of the tuck-folding design method for active origami structures: Given a goal mesh $\mathcal{M}$, find the mid-surface reference configuration $\mathcal{S}_0$ of an origami structure that approximates $\mathcal{M}$ in a folded configuration. $\mathcal{S}_0$ is comprised of the faces of $\mathcal{M}$ mapped to the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$ and introduced edge modules having three smooth folds and two rigid faces

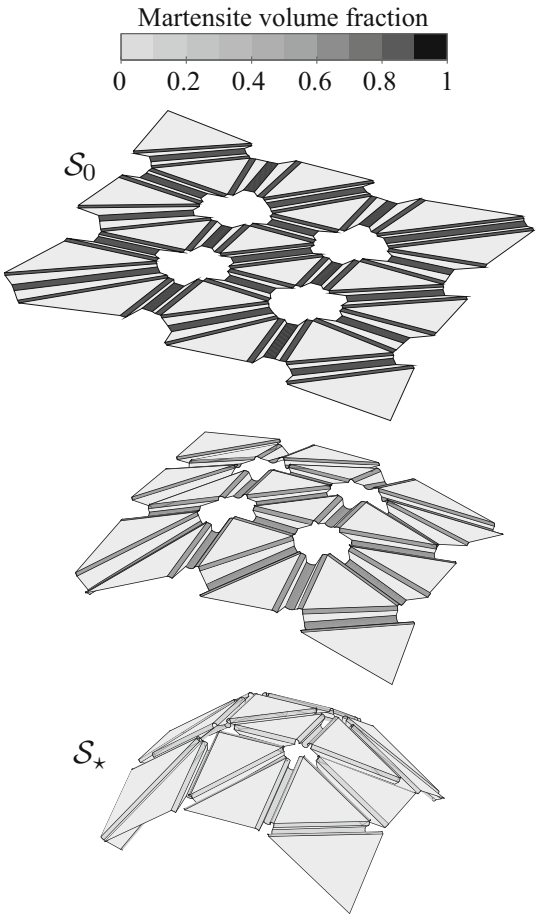


Fig. 8.28 Self-folding of a designed SMA-based origami structure (refer to Fig. 8.27) from the planar reference configuration to the configuration that approximates its associated goal mesh. The temperature is increased uniformly in the structure and all folds are simultaneously active

$$
\begin{aligned}
&\text{Find} && \hat{\boldsymbol{W}}, \hat{\boldsymbol{\psi}}, \hat{\boldsymbol{w}}^0 \\
&\text{That minimizes} && \|\hat{\boldsymbol{\theta}}^{t_f} - \hat{\boldsymbol{\theta}}^\star\| \\
&\text{Subject to} && \boldsymbol{h} = \boldsymbol{0} \\
& && \boldsymbol{g} \leq \boldsymbol{0} \\
& && \hat{\boldsymbol{w}}_L^0 \leq \hat{\boldsymbol{w}}^0 \leq \hat{\boldsymbol{w}}_U^0 \\
& && \text{Material failure constraints} \\
& && \text{Problem-dependent constraints.}
\end{aligned}
\tag{8.66}
$$

A flowchart showing the steps for the optimization and design evaluation processes in the tuck-folding method for the design of active origami structures is provided in Fig. 8.29. The optimization problem (8.66) is solved here using

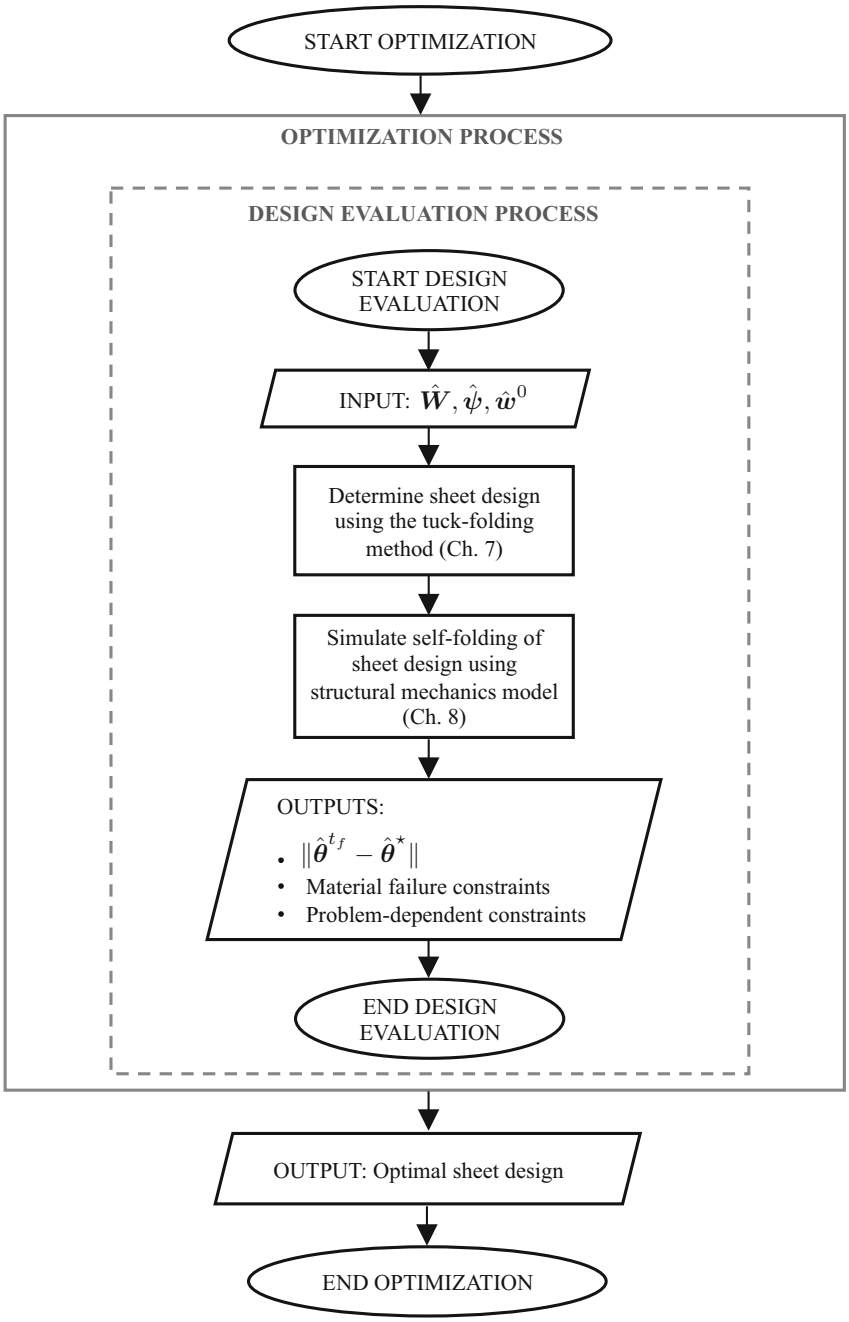


Fig. 8.29 Flowchart showing the steps of the optimization and design evaluation processes in the tuck-folding method for the design of self-folding structures

the gradient-based constrained optimization algorithm `fmincon` in MATLAB. Each design evaluation is executed using the structural mechanics model derived previously in this chapter. The laminate layup shown in Fig. 8.19 is adopted for all the examples presented in this section. No externally applied mechanical loads are considered. In all the examples, the temperature is increased from 300 to 400 K, leading to phase transformation in the SMA wires of the laminate shown in Fig. 8.19.

Figure 8.27 shows the goal mesh $\mathcal{M}$ and determined sheet design $\mathcal{S}_0$ associated with a goal shape of positive Gaussian curvature. Configurations attained by the designed structure during its folding history from $\mathcal{S}_0$ to $\mathcal{S}_\star$ are shown in Fig. 8.28. Figure 8.30 shows the same results as those in Figs. 8.27 and 8.28 but considering a goal mesh associated with a shape of negative Gaussian curvature. Both examples show that the tuck-folding method successfully provides the geometry of the reference configuration $\mathcal{S}_0$ of a single planar sheet that morphs towards a configuration that approximates $\mathcal{M}$ via active material-driven simultaneous folding. Examples of goal meshes having not only triangulated faces are shown in Figs. 8.31, 8.32, and 8.33. A more complex example of a goal mesh associated with the shape of a sphere is shown Fig. 8.34. Although only examples of SMA actuation are provided in this section, the present design method is applicable to structures having smooth fold domains comprised of other active materials so long as strains are small enough such that the linearized strain tensor can be utilized.

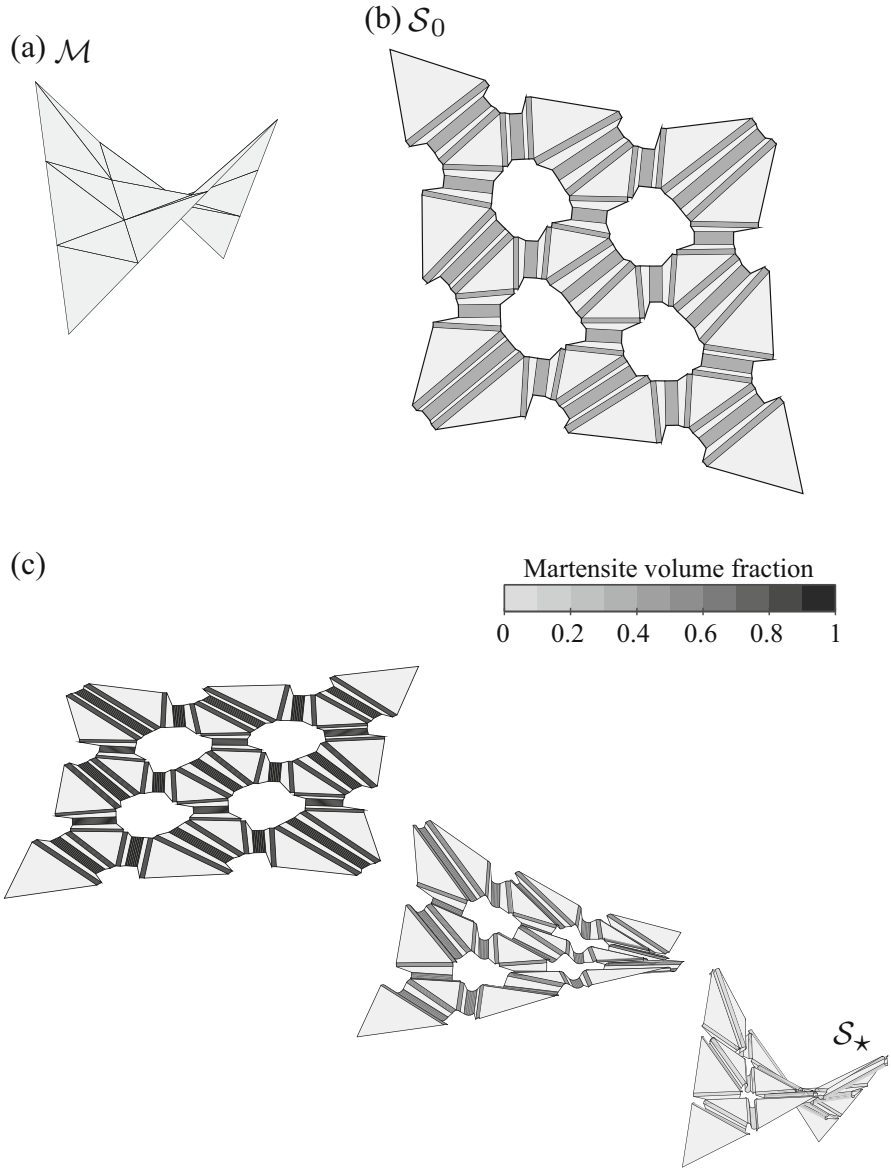


Fig. 8.30 Results of the implemented tuck-folding design method for active origami structures considering a goal mesh associated with a shape of negative Gaussian curvature: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the structure during its folding history

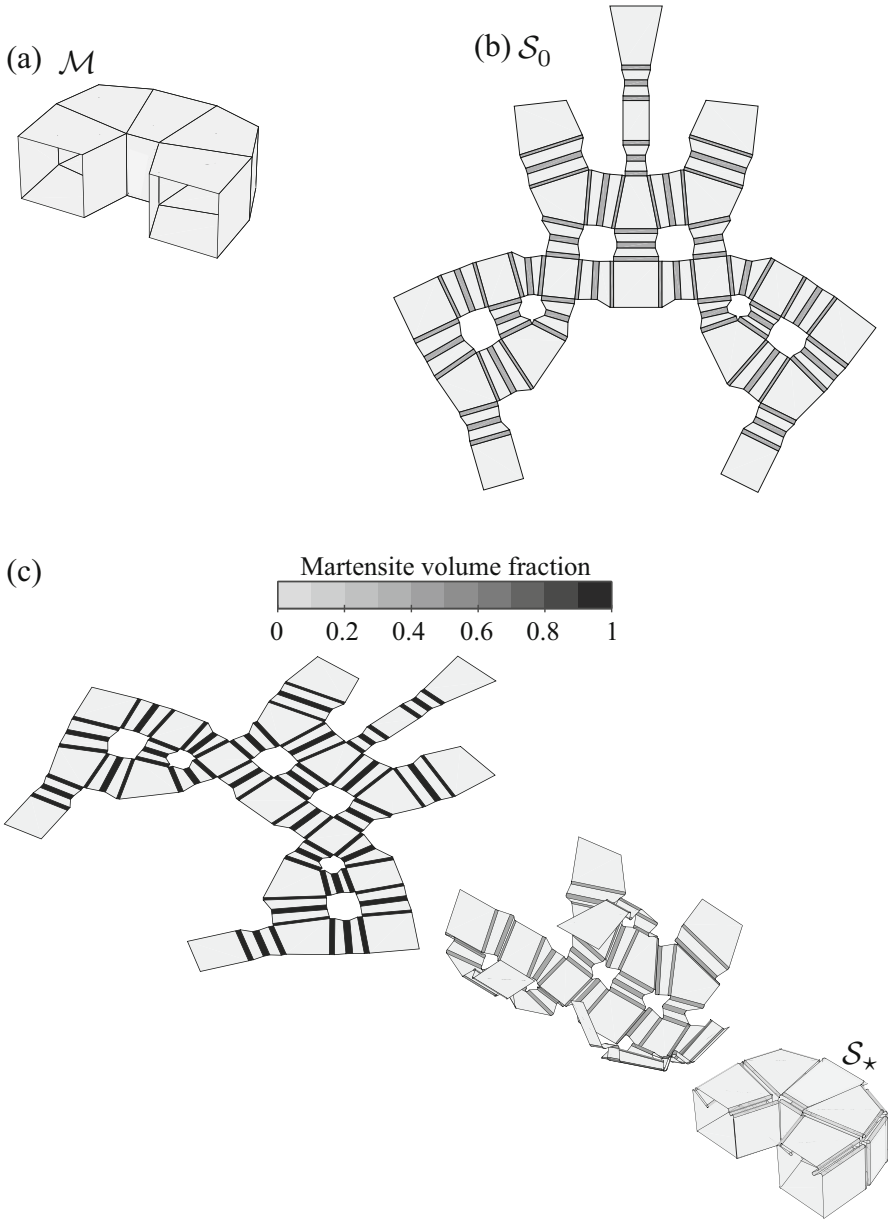


Fig. 8.31 Results of the implemented tuck-folding design method for active origami structures associated with a goal mesh comprised of quadrilateral faces: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the structure during its folding history

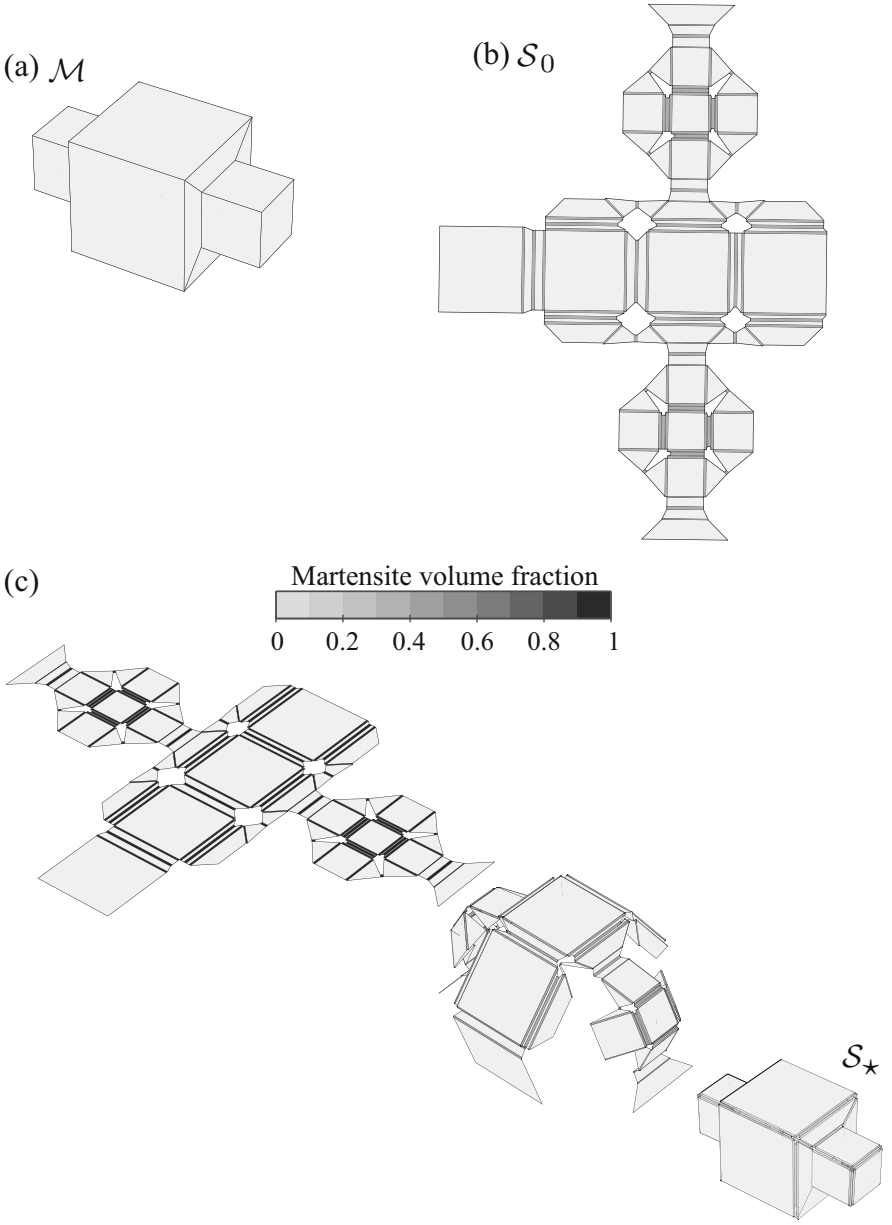


Fig. 8.32 Results of the implemented tuck-folding design method for active origami structures associated with a goal mesh comprised of quadrilateral faces: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the structure during its folding history

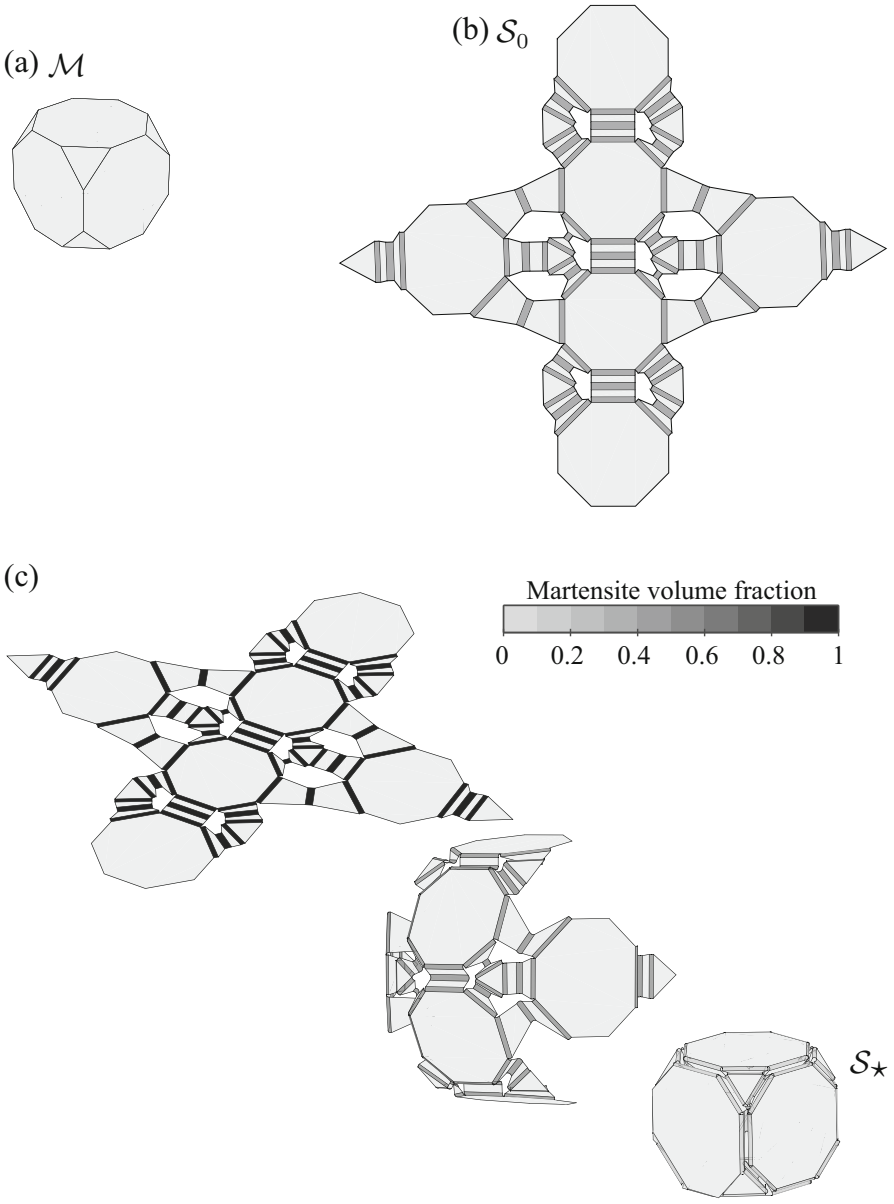


Fig. 8.33 Results of the implemented tuck-folding design method for active origami structures associated with the goal mesh of a truncated cube: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the structure during its folding history

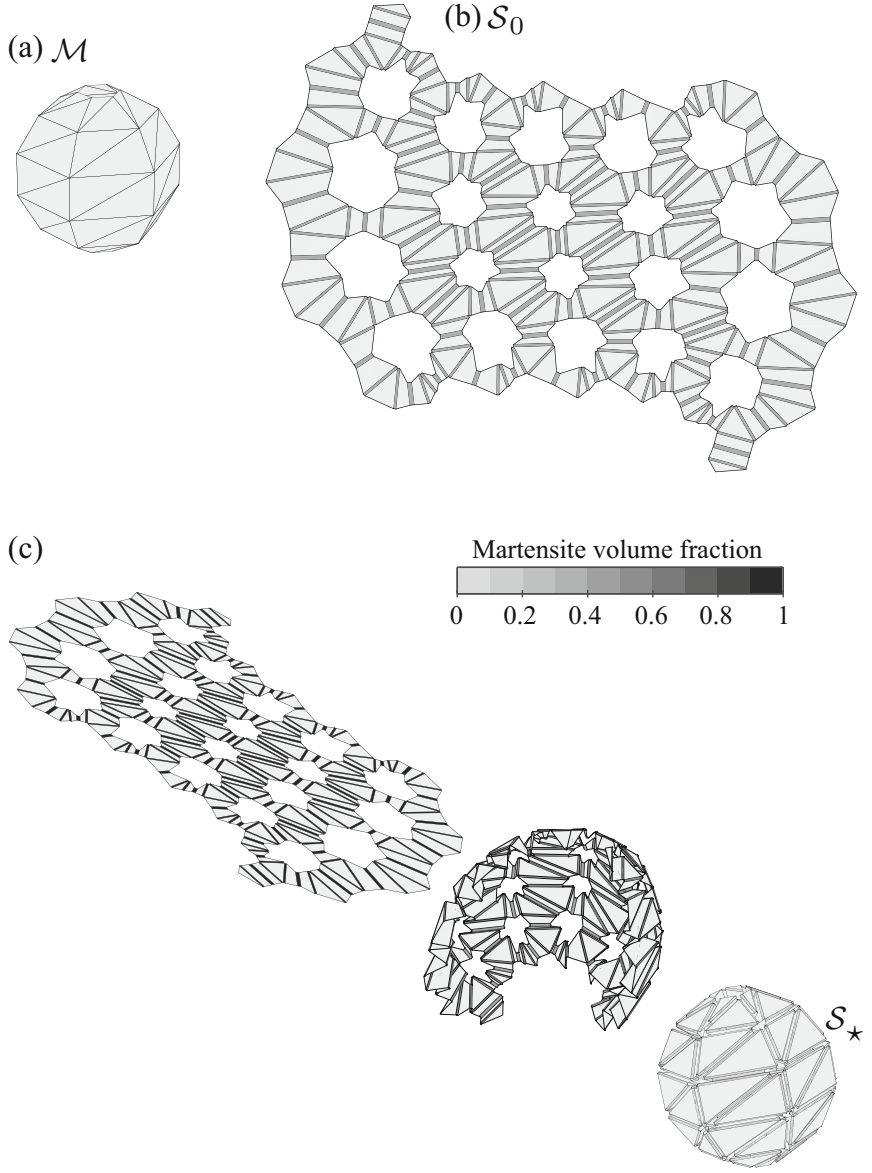


Fig. 8.34 Results of the implemented tuck-folding design method for active origami structures associated with the goal mesh of a sphere: (**a**) Goal mesh; (**b**) Determined sheet design with optimal fold widths; (**c**) Configurations attained by the structure during its folding history

8.7.1 *Design of a Self-Folding Parabolic Antenna Using the Tuck-Folding Method*

Origami structures have found various existing and potential implementations in antenna applications [53]. Tentzeris and coworkers analyzed and designed an origami spring antenna [54]. Such an antenna had a variable height that dictated its operating frequency. Tentzeris and coworkers also studied the performance of a helical antenna that can be folded and unfolded to operate at different resonant frequencies [55–59]. In addition to those designs, they have also explored an origami accordion antenna that can change its operating frequency based on its height [60, 61]. Fuchi and coworkers explored a frequency-selective origami sheet having a Miura-Ori fold pattern that can be tuned by changing its folded configuration [62–64]. Dickey and coworkers reported a study on microstrip origami antennas that can be self-reconfigured via SMP actuation [65]. Nogi and coworkers also investigated foldable antennas at the micro-scale [66]. Toshiyuki and coworkers developed a folding parabolic antenna with flat facets that can be stored in a compact configuration [67]. More recent studies consider several distinct fold patterns for origami antennas including the Kresling pattern [68], "bow-tie" pattern [69], and others [70].

As discussed in Chaps. 2 and 5, two main assumptions are generally made in models and design methods for origami [71, 72]: that folds are rectilinear creases of zeroth-order geometric continuity (*creased folds*), and that planar facets in the sheet are rigid. Often, creases may not be desirable in antenna surfaces. Considering the mechanical response of an antenna, sharp creased folds would induce stress concentrations and hence adversely affect the structural integrity of the antenna. Furthermore, sharp creases may affect the stability of the antenna radiation patterns (consult [73] for a comparison between sharp and round corners in antennas). To overcome these limitations imposed by the assumption of creased folds, models and design approaches for origami with *smooth folds* have been developed and presented in Chaps. 5–7 and the present chapter.

In this section, the *tuck-folding method* for the design of active origami structures developed in Sect. 8.7 is applied to a morphing antenna problem. A parabolic dish antenna is considered for this study. Parabolic shapes are employed in high-gain antennas and have several applications such as in communications and radio telescopes (e.g., high-speed data communication for satellites [74, 75]). The objective is to design a planar sheet with smooth folds that can self-fold via SMA actuation towards the goal parabolic shape. As in several realizations of parabolic antennas in the literature [76], the smooth parabolic shape of the antenna is discretized as a faceted structure. Such a realization is required in the present study for the application of the tuck-folding method presented in this section.

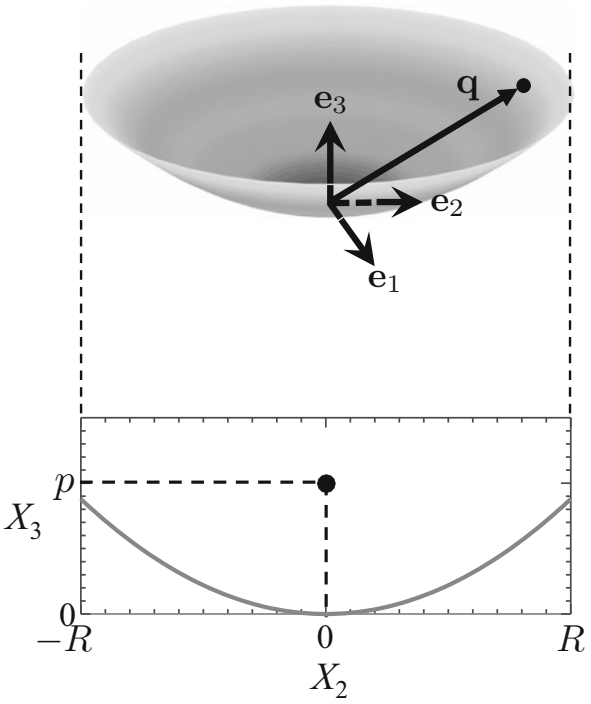


Fig. 8.35 (**a**) Parabolic shape and (**b**) its associated parabolic section and geometric parameters

8.7.1.1 Antenna Design Problem

The objective is to design a planar sheet with a pattern of smooth folds that can self-fold through SMA actuation towards the goal parabolic form. The origami modeling and design approaches employed here are not limited to parabolic goal shapes or to SMA actuation and can be applied to other goal shapes and origami structures comprised of other active materials [48, 77] (refer to the other examples shown in Sect. 8.7). Let us first define the goal shape and its associated discretization into a faceted structure. The center point of the parabolic surface is assumed to be located at the origin of a Cartesian coordinate system. The position vector of a point on the parabolic surface is denoted $\mathbf{q} \in \mathbb{R}^3$ and is given as follows:

$$\mathbf{q}(r,\varphi) = \begin{bmatrix} r\cos(\varphi) \\ r\sin(\varphi) \\ \frac{r^2}{4p} \end{bmatrix} \qquad 0 \le r \le R, \qquad 0 \le \varphi \le 2\pi, \tag{8.67}$$

where r is the coordinate on the radial direction, φ is the coordinate on the azimuthal direction, R is the radius of the parabolic surface, and p is the focal height. A schematic of a parabolic shape is provided in Fig. 8.35.

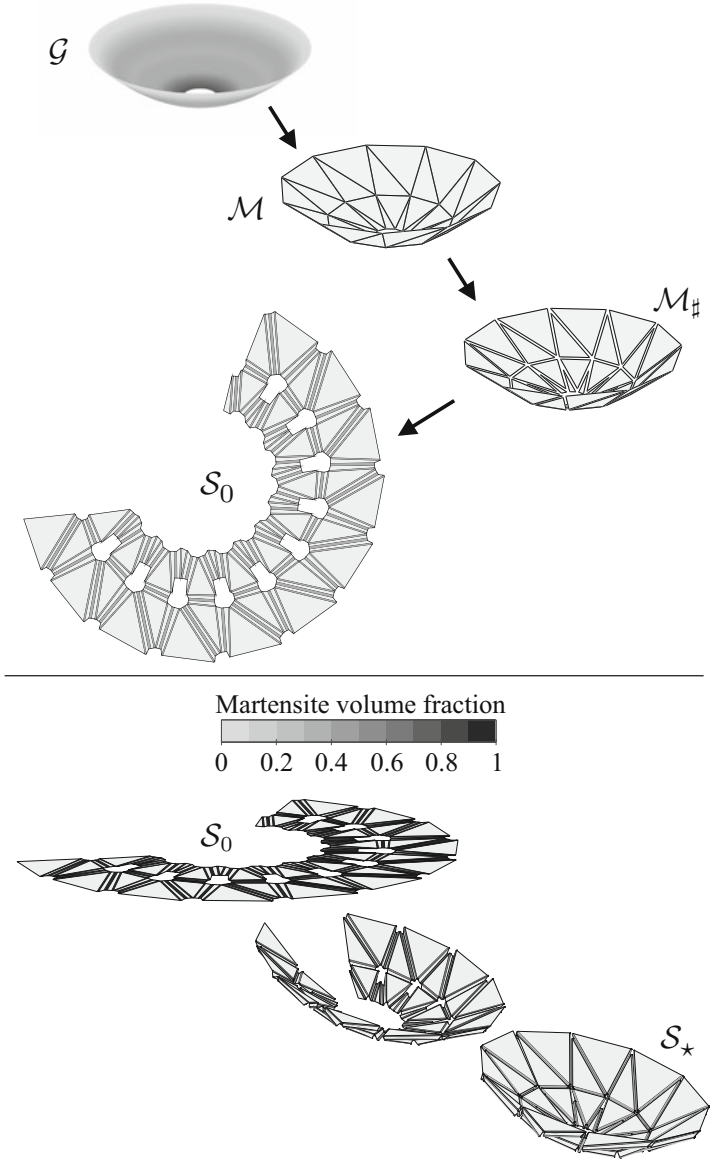


Fig. 8.36 Steps in the origami design and analysis approach employed for the self-folding antenna. A mesh discretization $\mathcal{M}$ of the parabolic goal shape $\mathcal{G}$ is first determined. The faces of $\mathcal{M}$ are trimmed to accommodate the smooth folds resulting in the trimmed mesh $\mathcal{M}_\sharp$. An origami sheet design $\mathcal{S}_0$ is then determined. The active material actuation-driven folding of the planar sheet towards the goal configuration $\mathcal{S}_\star$ that approximates the goal shape is then simulated using the model presented previously in this chapter

To determine a non-dimensional set of coordinates for the parabolic shape that will be useful in the mesh parameterization, the coordinates r and φ are substituted by the normalized coordinates $\eta_1 = r/R$ and $\eta_2 = \varphi/(2\pi)$. The expression for the position vector $\mathbf{q}$ is then given as follows:

$$\mathbf{q}(\eta_1, \eta_2) = \begin{bmatrix} \eta_1 R \cos(2\pi\eta_2) \\ \eta_1 R \sin(2\pi\eta_2) \\ \frac{\eta_1^2 R^2}{4p} \end{bmatrix} \qquad 0 \leq \eta_1 \leq 1, \qquad 0 \leq \eta_2 \leq 1. \tag{8.68}$$

The steps in the tuck-folding design method used to determine a sheet design for the parabolic antenna are illustrated in Fig. 8.36. Given a parabolic goal shape $\mathcal{G}$ with position vector $\mathbf{q}$ as defined in (8.68), a mesh discretization $\mathcal{M}$ of $\mathcal{G}$ is first determined. The mesh $\mathcal{M}$ is then trimmed to accommodate for smooth folds (consult Sect. 7.2.2 for details on the face trimming step). The resulting set of trimmed faces is denoted $\mathcal{M}_\sharp$. Afterwards, a planar sheet with a pattern of smooth folds denoted as $\mathcal{S}_0$ is then determined employing the tuck-folding method. The active material actuation-driven folding of the planar sheet towards the configuration that approximates the parabolic shape ($\mathcal{S}_\star$) is then simulated using the structural mechanics model presented previously in this chapter. The folding motion of the determined designs from $\mathcal{S}_0$ to $\mathcal{S}_\star$ is achieved through uniform changes in temperature in the structure without requiring externally applied forces or moments.

An analysis of the electromagnetic properties of the studied structural configuration $\mathcal{S}_\star$ is not performed in this study and is recommended for future work. The resulting antenna performance of $\mathcal{S}_\star$ is rather taken into account by tailoring certain geometric properties of such a configuration to achieve an expected improved performance. Since the goal shape $\mathcal{G}$ is discretized as a polygonal mesh $\mathcal{M}$, the reflective characteristics of the resulting origami design may not be as those obtained from a smooth parabolic shape. Thus, the following error in the normal vector orientation between the ideal parabolic shape $\mathcal{G}$ and its discretization $\mathcal{M}$ is introduced [53]:

$$E_{\text{Normal}} = \frac{\displaystyle\int_0^1 \int_0^1 J \, \cos^{-1}\left(\mathbf{n}_{\mathcal{G}} \cdot \mathbf{n}_{\mathcal{M}}\right) \, \mathrm{d}\eta_1 \mathrm{d}\eta_2}{\displaystyle\int_0^1 \int_0^1 J \, \mathrm{d}\eta_1 \mathrm{d}\eta_2}, \tag{8.69}$$

where $\mathbf{n}_{\mathcal{M}} \in \mathbb{R}^3$ is the unit normal vector of goal mesh $\mathcal{M}$,[4] $\mathbf{n}_{\mathcal{G}} \in \mathbb{R}^3$ is the unit normal vector of the parabolic goal shape $\mathcal{G}$ given as follows[5,6]:

$$\mathbf{n}_{\mathcal{G}} = \frac{1}{\left(\eta_1^2 R^2 + 4p^2\right)^{1/2}} \begin{bmatrix} -\eta_1 R \cos(2\pi\eta_2) \\ -\eta_1 R \sin(2\pi\eta_2) \\ 2p \end{bmatrix}, \tag{8.70}$$

and J is the surface element of integration given as follows:

$$J = 2\pi R^2 \eta_1 \left(1 + \frac{\eta_1^2 R^2}{4p^2}\right)^{1/2}. \tag{8.71}$$

Since the goal mesh $\mathcal{M}$ is trimmed to accommodate smooth folds resulting in the set of trimmed mesh faces $\mathcal{M}_\sharp$, the effective surface area of the resulting morphing antenna is lower than its untrimmed form. Therefore, the following error introduced by the mesh trimming process is considered:

$$E_{\text{Trim}} = 1 - \frac{\text{Area}(\mathcal{M}_\sharp)}{\text{Area}(\mathcal{M})}. \tag{8.72}$$

The potential effects that the increase in the value of E_{Trim} would have in the performance of a parabolic antenna include decrease of the reflective surface area and larger deviations in the radiation patterns from an ideal parabolic antenna.

The designed origami antenna increases its performance by minimizing the values of E_{Normal} and E_{Trim} (defined in (8.69) and (8.72), respectively). We remark that both of these outputs are dependent on the mesh discretization $\mathcal{M}$. To simplify the meshing problem, a structured triangulated mesh is used to discretize $\mathcal{G}$. The mapping of such a structured mesh is illustrated in Fig. 8.37. The two discrete parameters that define mesh refinement are N_1 and N_2 and, respectively, correspond to the number of triangle pairs in the mesh on the η_1 direction (radial direction) and the η_2 direction (azimuthal direction). For the length distribution assumed for the triangle pairs of the mesh in the direction aligned with η_1, the following conditions must be met:

[4] For each face of the goal mesh $\mathcal{M}$, the position vectors of its nodes are denoted $\tilde{\mathbf{y}}^{j1}, \tilde{\mathbf{y}}^{j2}, \tilde{\mathbf{y}}^{j3} \in \mathbb{R}^3$ and its unit normal vector is calculated via (3.5).

[5] The reader can readily verify the normalization coefficient in (8.70) since $\left((-\eta_1 R\cos(2\pi\eta_2))^2 + (-\eta_1 R \sin(2\pi\eta_2))^2 + (2p)^2\right)^{1/2} = \left(\eta_1^2 R^2(\cos^2(2\pi\eta_2) + \sin^2(2\pi\eta_2)) + 4p^2\right)^{1/2} = \left(\eta_1^2 R^2 + 4p^2\right)^{1/2}$.

[6] The unit normal vector $\mathbf{n}_{\mathcal{G}}$ of the parabolic surface is determined as $\mathbf{n}_{\mathcal{G}} = \left(\frac{\partial \mathbf{q}}{\partial \eta_1} \times \frac{\partial \mathbf{q}}{\partial \eta_2}\right) \| \frac{\partial \mathbf{q}}{\partial \eta_1} \times \frac{\partial \mathbf{q}}{\partial \eta_2} \|^{-1}$.

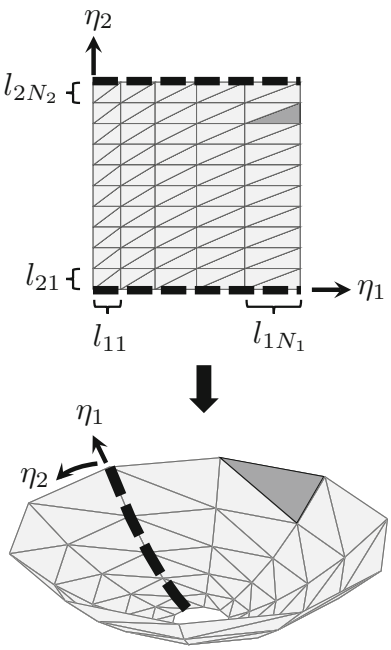


Fig. 8.37 Mapping of a structured mesh for a parabolic shape

$$\sum_{i=1}^{N_1} l_{1i} = 1, \qquad \frac{l_{11}}{l_{1N_1}} = B, \tag{8.73}$$

where B is the mesh size bias ratio. The first condition in (8.73) arises because the range of η_1 is the interval [0, 1] and therefore its subdivisions $l_{11}, \ldots, l_{1N_1}$ must add to 1. The second condition allows for size biasing in the distribution of the triangle pairs of the mesh in the direction aligned with η_1. A linear function for $l_{11}, \ldots, l_{1N_1}$ is assumed here for simplicity. Such a linear function is obtained by considering the conditions of (8.73):

$$l_{1i} = \frac{-2(B-1)i + 2(N_1 B - 1)}{N_1(N_1 - 1)(B+1)}, \qquad i = 1, \ldots, N_1. \tag{8.74}$$

Considering (8.74), a value of $B = 1$ results in a uniform mesh distribution along η_1. A uniform mesh distribution is assumed for the triangle pairs in the direction aligned with η_2 (azimuthal direction of the parabolic shape). For the meshing along this direction, the following conditions must be met (see Fig. 8.37):

$$\sum_{i=1}^{N_2} l_{2i} = 1, \qquad l_{21} = \cdots = l_{2i} = \cdots = l_{2N_2}, \tag{8.75}$$

which simply yields the following values for $l_{21}, \ldots, l_{2N_2}$:

$$l_{2i} = \frac{1}{N_2}, \qquad i = 1, \ldots, N_2. \tag{8.76}$$

The design problem considered in this section entails the determination of the mesh parameters N_1, N_2, and B that minimize E_{Normal} and E_{Trim}, i.e.,

$$\begin{aligned} &\text{Find: } N_1,\ N_2,\ B \\ &\text{That minimizes: } \mathbf{f} = \left[E_{\text{Normal}}(N_1, N_2, B),\ \ E_{\text{Trim}}(N_1, N_2, B)\right]. \end{aligned} \tag{8.77}$$

A full factorial design of experiments (DOE) [78] is used to explore the design space and to construct a Pareto frontier of the designs in the $E_{\text{Normal}} - E_{\text{Trim}}$ space.

8.7.1.2 Results of the Antenna Design Exploration Study

A full factorial DOE is used to explore the design space and to determine a Pareto frontier of designs in the $E_{\text{Normal}} - E_{\text{Trim}}$ space (defined in (8.69) and (8.72), respectively). For the example considered here, values of $R = 2\,\text{m}$ and $p = 1\,\text{m}$ are assumed (see Fig. 8.35). The upper bounds for the levels of N_1 and N_2 in the DOE correspond to those for which the design framework outlined in Sect. 8.7 is able to provide a resulting origami sheet design (i.e., when the faces of the goal mesh are large enough such that SMA-based smooth folds can be accommodated in the sheet design $\mathcal{S}_0$). The levels for the variable N_1 are $\{2, 3, 4, 5, 6\}$, the levels for the variable N_2 are $\{3, 4, 5, 6, 7, 8, 9, 10, 11\}$, and the levels for the variable B are $\{0.50, 0.75, 1.00, 1.25, 1.50, 1.75, 2.00\}$. Thus, a total of 315 designs are analyzed. The locations of all the analyzed designs in the $E_{\text{Normal}} - E_{\text{Trim}}$ space are shown in Fig. 8.38a. From these results, a Pareto frontier is constructed and shown in Fig. 8.38b. Meshes with lower values of N_1 and N_2 have fewer faces and such faces have larger size; therefore, the face trimming process applied for each interior edge does not significantly decrease the total area of $\mathcal{M}$. Thus, for such designs, the value of E_{Trim} is lower. However, since such discretizations are coarse, the error in the normal orientation E_{Normal} is higher. Conversely, meshes with high values of N_1 and N_2 have more faces and such faces are of smaller size. Since such mesh discretizations are finer, the error in the face orientation E_{Normal} is lower. The trade-off between the two performance outputs E_{Normal} and E_{Trim} is clearly observed in Fig. 8.38b.

Figure 8.39 presents results of three designs located in the Pareto frontier of Fig. 8.38. It shows the goal meshes $\mathcal{M}$, the trimmed mesh $\mathcal{M}_\sharp$, and obtained sheet designs $\mathcal{S}_0$. The design of $N_1 = 2$, $N_2 = 4$, and $B = 0.50$ has the lowest value of E_{Trim} of all the analyzed designs. The design of $N_1 = 6$, $N_2 = 11$, $B = 1.00$ has the lowest value of normal orientation error E_{Normal} of all the analyzed designs.

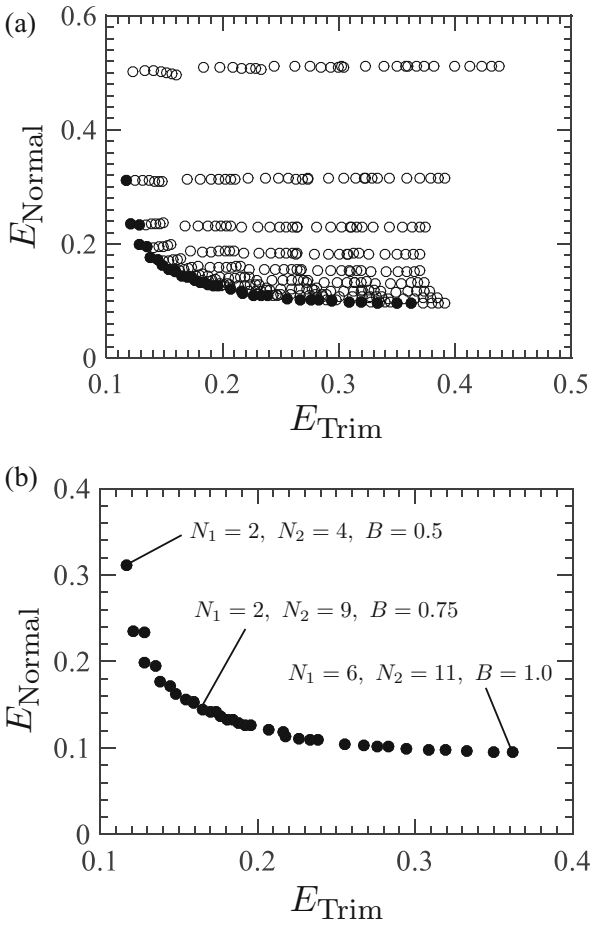


Fig. 8.38 DOE results: (**a**) E_{Normal} vs. E_{Trim} for all designs. The filled dots indicate designs in the Pareto front; (**b**) E_{Normal} vs. E_{Trim} for the designs in the Pareto front. The value of the design variables for the sheets presented in Fig. 8.39 is shown

Finally, Fig. 8.40 shows the SMA actuation-driven folding for the design of $N_1 = 2$, $N_2 = 9$, and $B = 0.75$. It is observed that the origami design framework of Sect. 8.7 successfully generates a sheet design that allows for self-folding driven by uniform temperature increase in the structure.

An analysis of the electromagnetic properties of the determined active antennas was not performed in this study and is suggested for future work. The resulting antenna performance was instead taken into account in this study by tailoring geometric properties of the goal mesh for an expected improved performance. Another aspect to be addressed in future work is the comparison of the trade-offs between the use of creased folds (that may affect the stability of the antenna radiation patterns) and the use of smooth folds (that results in a decrease of the antenna functional area due to the required face trimming, but allows the introduction of active materials for self-folding capabilities). To eliminate the disadvantages introduced by face trimming, an extension of the design method

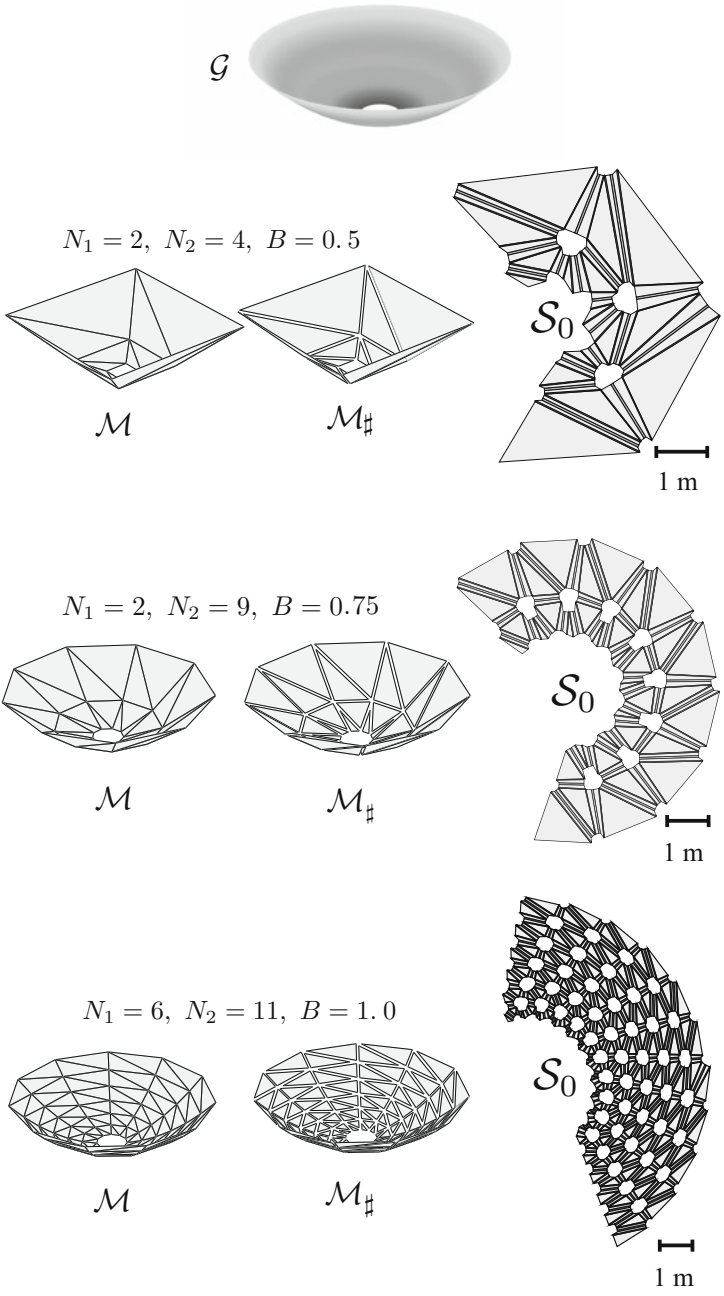


Fig. 8.39 Results of three designs located in the Pareto front shown in Fig. 8.38. The goal meshes $\mathcal{M}$, the trimmed mesh $\mathcal{M}_\sharp$, and determined origami sheet designs $\mathcal{S}_0$ are shown

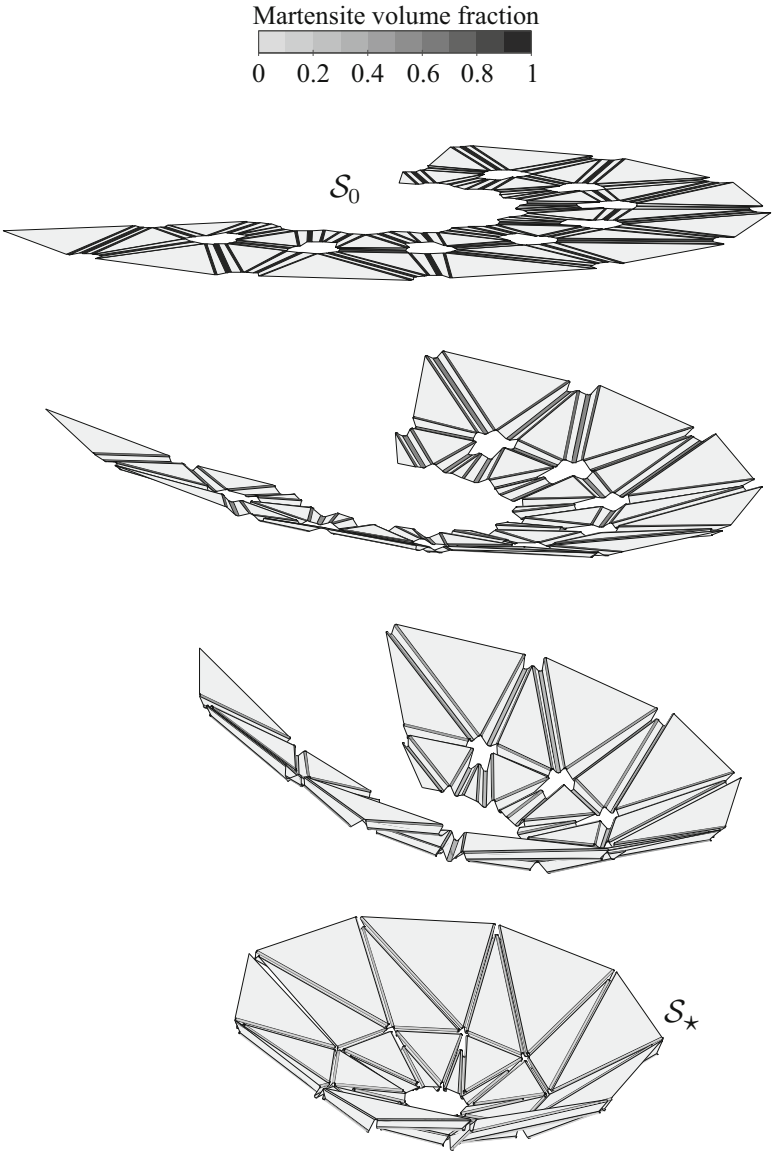


Fig. 8.40 Self-folding motion of the design having mesh parameters $N_1 = 2, N_2 = 9, B = 0.75$ (see Fig. 8.39) from the planar configuration to the parabolic configuration driven by SMA actuation

presented in Sect. 8.7 consists of overlaying the untrimmed goal mesh faces to the determined sheet design $\mathcal{S}_0$. Such an extension is illustrated in Fig. 8.41 for a simple goal mesh having one interior vertex and in Fig. 8.42 for a parabolic dish goal mesh.

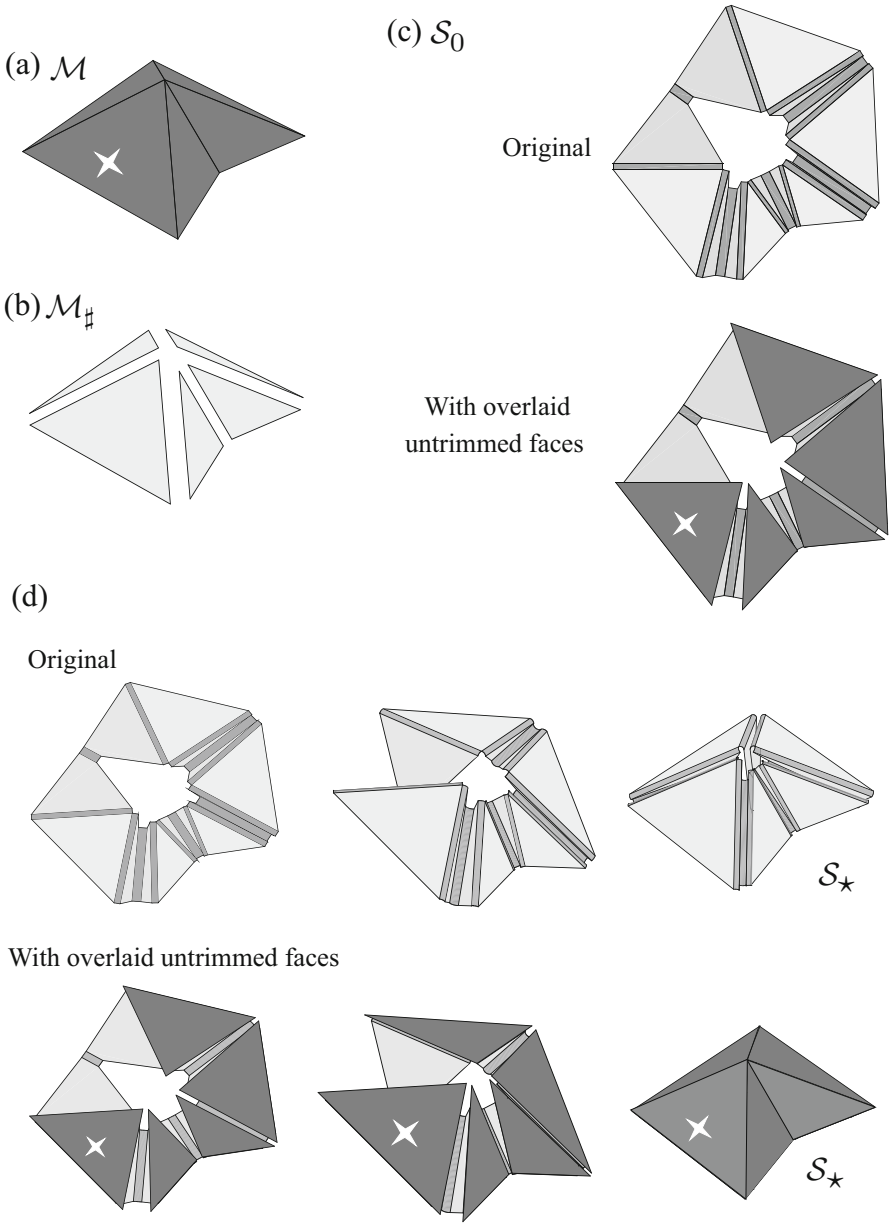


Fig. 8.41 Implementation example of the tuck-folding method considering overlaying of untrimmed faces onto the determined sheet design: (**a**) Goal mesh $\mathcal{M}$ having one interior vertex; (**b**) Trimmed mesh $\mathcal{M}_\sharp$; (**c**) Determined origami sheet designs $\mathcal{S}_0$ with and without overlaid untrimmed faces; (**d**) Folding motion of the origami sheet designs. The sheet with overlaid untrimmed faces does not have gaps or holes in the goal configuration $\mathcal{S}_\star$

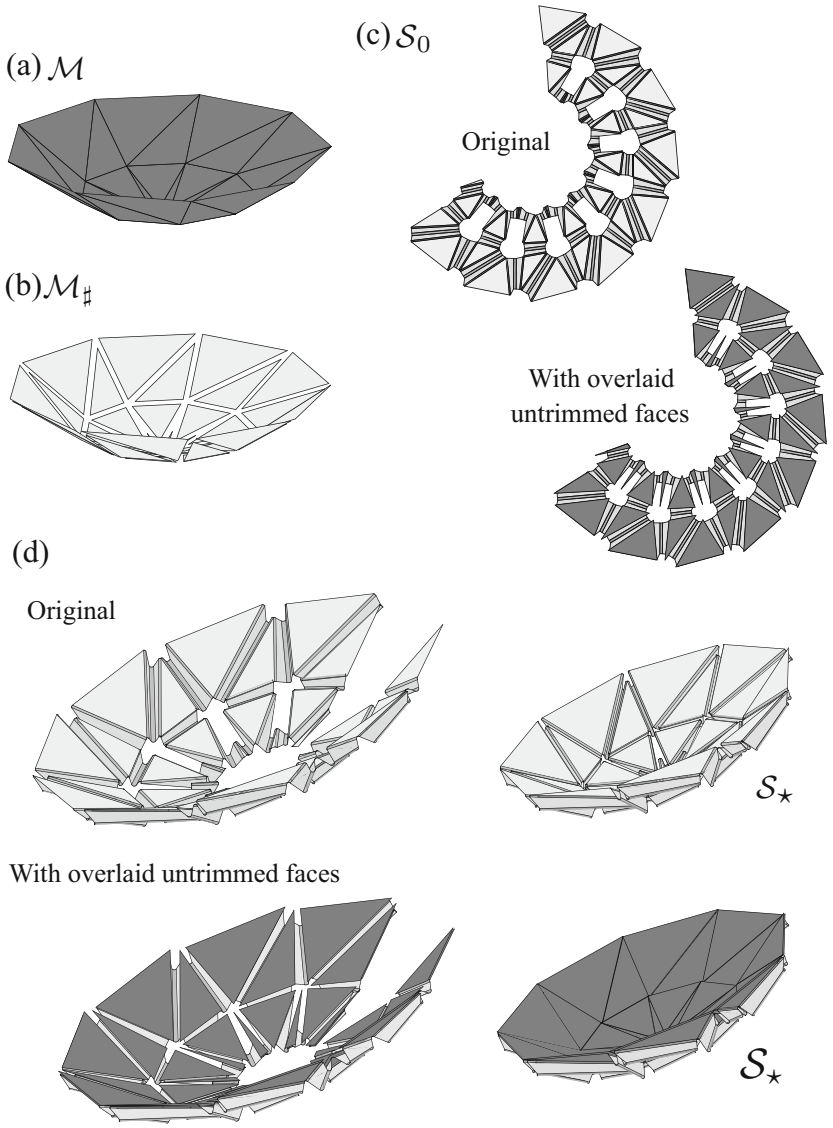


Fig. 8.42 Implementation example of the tuck-folding method considering overlaying of untrimmed faces onto the determined sheet design: (**a**) Goal mesh $\mathcal{M}$ of a parabolic dish; (**b**) Trimmed mesh $\mathcal{M}_\sharp$; (**c**) Determined origami sheet designs $\mathcal{S}_0$ with and without overlaid untrimmed faces; (**d**) Folding motion of the origami sheet designs. The sheet with overlaid untrimmed faces does not have gaps or holes in the goal configuration $\mathcal{S}_\star$

Chapter Summary

In this chapter, we developed a model for the structural mechanics of origami continuum bodies having smooth fold domains of non-zero thickness. Such a structural model integrates the surface kinematics model for origami with smooth folds presented in Chap. 5 and existing plate theories to obtain a structural representation for folds of non-zero thickness. In the structural model, the kinematic variables that fully describe the configuration of the origami structure with smooth fold domains are the fold angles and fold arc-lengths of each fold in the structure. The governing field equations were presented and a variational approach including the consideration of kinematic constraints was used to develop the structural model.

The present model does not make any assumptions on the material response of the origami structures; therefore, it is applicable to a large spectrum of origami structures (ranging from those having elastic folds to those having *active material actuation*). The numerical implementation of the model was described and various implementation examples were presented. The results determined using the present model were numerically validated against higher-fidelity FEA and good agreement is observed for all the examples. Furthermore, the present model requires significantly fewer kinematic variables to describe the configuration of an origami structure as compared to FEA and hence is significantly more computationally efficient. In the structural mechanics model presented here, classical plate theory assumptions were adopted for each smooth fold domain. For future work, higher-order plate theories or layerwise theories may be considered to improve the current model.

Finally, we integrated the design methods of Chaps. 6 and 7 with the mechanics of active origami structures presented herein to develop frameworks for the design of self-folding structures that morph towards targeted shapes under the application of a non-mechanical stimulus (e.g., thermal, chemical, electromagnetic). It is hoped that the curious and motivated reader will use these frameworks to further advance the field of origami engineering and the applications of active origami. In Sect. 8.6, we extended the method of unfolding polyhedra from Chap. 6 for the design of active self-folding structures. The tuck-folding method from Chap. 7 was also extended for the design of active self-folding structures in Sect. 8.7. We finalized the chapter by presenting a study on the analysis and design of an active self-folding antenna in Sect. 8.7.1.

Problems

8.1 Show that the mid-surface strain and signed curvature fields for smooth folds with G^2 continuity provided in (8.10) satisfy the relations stated in (8.8).

8.2 Verify the formulation for the components of the tangent matrices provided in (8.45)–(8.52).

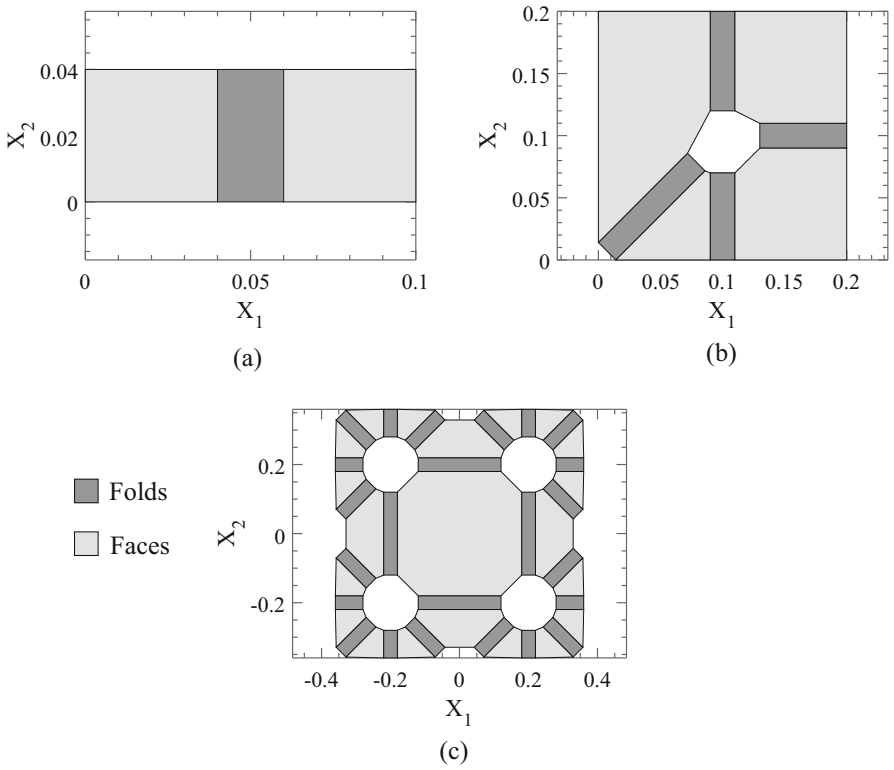


Fig. 8.43 Geometry of the origami sheets for Problems 8.4–8.9. Axes units are in meters

8.3 Verify the derivatives of the axial force and moment resultants with respect to fold angle and fold arc-length provided in (8.53)–(8.56).

8.4 Simulate the origami sheet of the boundary value problem shown in Fig. 8.5 considering smooth folds comprised of linear elastic materials having the following material parameters: (a) Young's modulus $E = 150$ GPa, Poisson's ratio $\nu = 0.3$; (b) $E = 200$ GPa, $\nu = 0.3$. Compare the results of (a) and (b) with those shown in Fig. 8.6. The geometry of the sheet is shown in detail in Fig. 8.43a.

8.5 Repeat Problem 8.4 considering the origami sheet of the boundary value problem shown in Fig. 8.9. The geometry of the sheet is shown in detail in Fig. 8.43b.

8.6 Repeat Problem 8.4 considering the origami sheet of the boundary value problem shown in Fig. 8.13. The geometry of the sheet is shown in detail in Fig. 8.43c.

8.7 Simulate the origami sheet of the boundary value problem shown in Fig. 8.7 but replace the SMA/elastomer composite layer with a thermoelastic material having

the following material parameters: (a) Young's modulus $E = 1$ GPa, Poisson's ratio $\nu = 0.3$, thermoelastic expansion coefficient $\alpha = 0.0005\ \mathrm{K}^{-1}$; (b) $E = 1$ GPa, $\nu = 0.3$, $\alpha = 0.005\ \mathrm{K}^{-1}$. Compare the results of (a) and (b) with those shown in Fig. 8.8. Refer to (C.4) for the formulation of a constitutive model for thermoelastic materials. The geometry of the sheet is shown in detail in Fig. 8.43a.

8.8 Repeat Problem 8.7 considering the origami sheet of the boundary value problem shown in Fig. 8.11. The geometry of the sheet is shown in detail in Fig. 8.43b.

8.9 Repeat Problem 8.7 considering the origami sheet of the boundary value problem shown in Fig. 8.15. The geometry of the sheet is shown in detail in Fig. 8.43c.

8.10 Simulate the response of the piezoelectric-based self-folding strips shown in Fig. 8.44. The laminate layup of the piezoelectric-based folds is shown in Fig. 8.44d. Assume that the aluminum layer has a linear elastic response with material parameters from Table 8.4. The response of the piezoelectric layers can be modeled using (C.6). For the piezoelectric layers, assume the following material parameters: $E_1 = E_2 = 60\,\mathrm{GPa}$, $\nu_{12} = 0.3$, $G_{12} = 23\,\mathrm{GPa}$, $d_{113} = 0$, and $d_{223} = 741\mathrm{E}{-12}$ m/V. For all the folds, assume that the electric potential is initially zero everywhere. Then, increase the electric potential at the top face of the top piezoelectric layer to 1000 V (leading to a decrease of the electric field component $\hat{E}_3$ from 0 to $-1000/0.002$ V/m in the top layer) and simultaneously increase the voltage at the bottom face of the bottom piezoelectric layer to 1000 V (leading to an increase of the electric field component $\hat{E}_3$ from 0 to $1000/0.002$ V/m in the bottom layer). Report the final position of point A (labeled in Fig. 8.44) for cases a, b, and c.

8.11 Implement the unfolding polyhedra method for the design of active origami structures in a computational environment and determine sheet designs for the 13 Archimedean polyhedra.

8.12 Implement the tuck-folding method for the design of active origami structures in a computational environment. Then, determine a sheet design for the goal mesh with one interior node of positive Gaussian curvature shown in Fig. 8.45. Assume that all the folds are comprised of the laminated composite shown in Fig 8.7. The position vectors of the nodes are given as follows (units in meters):

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\ \hat{\mathbf{y}}^7\right]^\top = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.60 & 0.00 & 0.40 \\ 0.30 & 0.52 & 0.40 \\ -0.30 & 0.52 & 0.40 \\ -0.60 & 0.00 & 0.40 \\ -0.30 & -0.52 & 0.40 \\ 0.30 & -0.52 & 0.40 \end{bmatrix}, \tag{8.78}$$

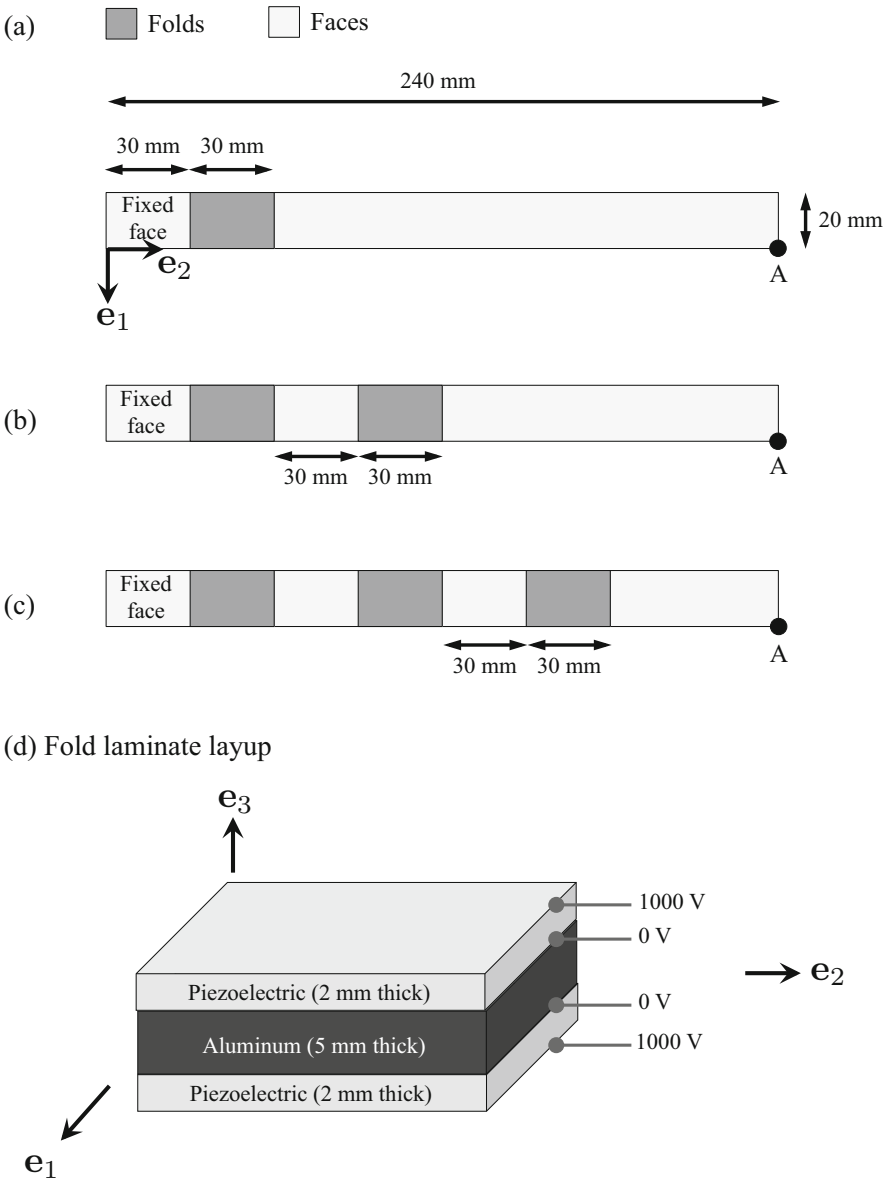


Fig. 8.44 Piezoelectric-based self-folding strips with (**a**) one fold, (**b**) two folds, and (**c**) three-folds. The laminate layup of the piezoelectric-based folds is shown in (**d**)

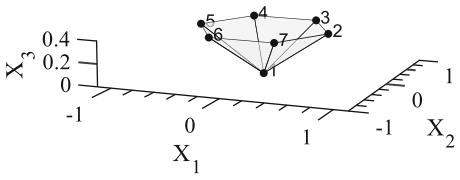


Fig. 8.45 Goal mesh $\mathcal{M}$ for Problem 8.12 (units in meters)

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given as follows:

$$\mathbf{C}^{\mathcal{M}} = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 3 & 4 \\ 1 & 4 & 5 \\ 1 & 5 & 6 \\ 1 & 6 & 7 \\ 1 & 7 & 2 \end{bmatrix}. \tag{8.79}$$

8.13 Implement the tuck-folding method for the design of active origami structures in a computational environment. Then, determine a sheet design for the goal mesh with one interior node of negative Gaussian curvature shown in Fig. 8.46. Assume that all the folds are comprised of the laminated composite shown in Fig. 8.7. The position vectors of the nodes are given as follows:

$$\left[\hat{\mathbf{y}}^1\ \hat{\mathbf{y}}^2\ \hat{\mathbf{y}}^3\ \hat{\mathbf{y}}^4\ \hat{\mathbf{y}}^5\ \hat{\mathbf{y}}^6\ \hat{\mathbf{y}}^7\right]^\top = \begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.80 & 0.00 & -0.18 \\ 0.40 & 0.69 & 0.18 \\ -0.40 & 0.69 & -0.18 \\ -0.80 & 0.00 & 0.18 \\ -0.40 & -0.69 & -0.18 \\ 0.40 & -0.69 & 0.18 \end{bmatrix}, \tag{8.80}$$

and the components of the mesh connectivity matrix $\mathbf{C}^{\mathcal{M}}$ are given in (8.79).

8.14 Implement the tuck-folding method for the design of active origami structures in a computational environment and determine sheet designs for the 13 Archimedean polyhedra. Compare your results with those of Problem 8.11.

8.15 Implement the tuck-folding method for the design of active origami structures in a computational environment and determine sheet designs for two different non-convex goal meshes (your choice).

8.16 Implement the extension of the design method of Sect. 8.7 consisting of overlaying the untrimmed goal mesh faces to the determined sheet design $\mathcal{S}_0$ as illustrated in Figs. 8.41 and 8.42.

8.17 Repeat the case study of the active origami antenna employing a different mesh parameterization than the one used in Sect. 8.7.1.

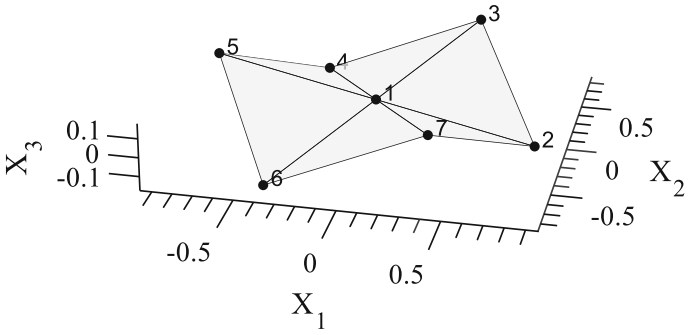


Fig. 8.46 Goal mesh $\mathcal{M}$ for Problem 8.13 (units in meters)

References

1. M. Schenk, S.D. Guest, Origami folding: a structural engineering approach, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (2011), pp. 291–304
2. A.A. Evans, J.L. Silverberg, C.D. Santangelo, Lattice mechanics of origami tessellations. Phys. Rev. E **92**(1), 013205 (2015)
3. G.V. Rodrigues, L.M. Fonseca, M.A. Savi, A. Paiva, Nonlinear dynamics of an adaptive origami-stent system. Int. J. Mech. Sci. **133**, 303–318 (2017)
4. C.M. Wheeler, M.L. Culpepper, Soft origami: classification, constraint, and actuation of highly compliant origami structures. J. Mech. Robot. **8**(5), 051012 (2016)
5. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr., D.C. Lagoudas, Origami-inspired active structures: a synthesis and review. Smart Mater. Struct. **23**(9), 094001 (2014)
6. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Kinematics of origami structures with smooth folds. J. Mech. Robot. **8**(6), 061019 (2016)
7. H. Yasuda, Z. Chen, J. Yang, Multitransformable leaf-out origami with bistable behavior. J. Mech. Robot. **8**(3), 031013 (2016)
8. J. Ma, Z. You, The origami crash box, in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (2011), pp. 277–290
9. J. Ma, Z. You, A novel origami crash box with varying profiles, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2013–13495 (American Society of Mechanical Engineers, New York, 2013), pp. V06BT07A048.
10. J. Ma, D. Hou, Y. Chen, Z. You, Quasi-static axial crushing of thin-walled tubes with a kite-shape rigid origami pattern: numerical simulation. Thin-Walled Struct. **100**,38–47 (2016)
11. J. Ma, Z. You, Energy absorption of thin-walled beams with a pre-folded origami pattern. Thin-Walled Struct. **73**, 198–206 (2013)
12. Y. Li, Z. You, Thin-walled open-section origami beams for energy absorption, in *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2014–35204 (American Society of Mechanical Engineers, New York, 2014), pp. V003T01A014
13. D. Hou, Y. Chen, J. Ma, Z. You, Axial crushing of thin-walled tubes with kite-shape pattern, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2015–46671 (American Society of Mechanical Engineers, New York, 2015), pp. V05BT08A037

14. K. Yang, S. Xu, J. Shen, S. Zhou, Y.M. Xie, Energy absorption of thin-walled tubes with pre-folded origami patterns: numerical simulation and experimental verification. Thin-Walled Struct. **103**, 33–44 (2016)
15. E.A. Peraza Hernandez, D.J. Hartl, R.J. Malak Jr., Design and numerical analysis of an SMA mesh-based self-folding sheet. Smart Mater. Struct. **22**(9), 094008 (2013)
16. E. Peraza Hernandez, D. Hartl, E. Galvan, R. Malak, Design and optimization of a shape memory alloy-based self-folding sheet. J. Mech. Des. **135**(11), 111007 (2013)
17. R.W. Mailen, M.D. Dickey, J. Genzer, M.A. Zikry, A fully coupled thermo-viscoelastic finite element model for self-folding shape memory polymer sheets. J. Polym. Sci. B Polym. Phys. **55**(16), 1207–1219 (2017)
18. S. Ahmed, C. Lauff, A. Crivaro, K. McGough, R. Sheridan, M. Frecker, P. von Lockette, Z. Ounaies, T. Simpson, J.-M. Lien, R. Strzelec, Multi-field responsive origami structures: preliminary modeling and experiments, in *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference IDETC/CIE*, No. DETC2013–12405, Portland (2013), pp. V06BT07A028
19. T. Hull, *Project Origami: Activities for Exploring Mathematics* (CRC Press, Boca Raton, 2012)
20. W.S. Slaughter, *The Linearized Theory of Elasticity* (Birkhäuser, Boston, 2002)
21. J.N. Reddy, *Mechanics of Laminated Composite Plates: Theory and Analysis* (CRC Press, Boca Raton, 1997)
22. N.J. Pagano, Exact solutions for composite laminates in cylindrical bending. J. Compos. Mater. **3**(3), 398–411 (1969)
23. P. Heyliger, S. Brooks, Exact solutions for laminated piezoelectric plates in cylindrical bending. J. Appl. Mech. **63**(4), 903–910 (1996)
24. P.V. Nimbolkar, I.M. Jain, Cylindrical bending of elastic plates. Proc. Math. Sci. **10**, 793–802 (2015)
25. K. Fuchi, T.H. Ware, P.R. Buskohl, G.W. Reich, R.A. Vaia, T.J. White, J.J. Joo, Topology optimization for the design of folding liquid crystal elastomer actuators. Soft Matter **11**(37), 7288–7295 (2015)
26. G.A. Holzapfel, *Nonlinear Solid Mechanics: A Continuum Approach for Engineering* (Wiley, Chichester, 2000)
27. J.N. Reddy, *Energy Principles and Variational Methods in Applied Mechanics* (Wiley, Hoboken, 2002)
28. J.N. Reddy, *An Introduction to Nonlinear Finite Element Analysis: With Applications to Heat Transfer, Fluid Mechanics, and Solid Mechanics* (Oxford University Press, Oxford, 2014)
29. D.C. Lagoudas (ed.), *Shape Memory Alloys: Modeling and Engineering Applications* (Springer Science + Business Media, LLC, New York, 2008)
30. M. Budimir, Piezoelectric anisotropy and free energy instability in classic perovskites. Technical report, Materiaux, Ecole Polytechnique Fédérale de Lausane, 2006
31. J. Lee, J.G. Boyd IV, D.C. Lagoudas, Effective properties of three-phase electro-magneto-elastic composites. Int. J. Eng. Sci. **43**(10), 790–825 (2005)
32. P. Tan, L. Tong, Modeling for the electro-magneto-thermo-elastic properties of piezoelectric-magnetic fiber reinforced composites. Compos. A: Appl. Sci. Manuf. **33**(5), 631–645 (2002)
33. C.Y.K. Chee, L. Tong, G.P. Steven, A review on the modelling of piezoelectric sensors and actuators incorporated in intelligent structures. J. Intell. Mater. Syst. Struct. **9**(1), 3–19 (1998)
34. C.M. Wayman, H.K.D.H. Bhadeshia, Phase transformations, nondiffusive. Phys. Metall. **2**, 1507–1554 (1983)
35. D.A. Porter, K.E. Easterling, M. Sherif, *Phase Transformations in Metals and Alloys (Third Edition)* (CRC Press, Boca Raton, 2009)
36. R. Kainuma, Y. Imano, W. Ito, Y. Sutou, H. Morito, S. Okamoto, O. Kitakami, K. Oikawa, A. Fujita, T. Kanomata, K. Ishida, Magnetic-field-induced shape recovery by reverse phase transformation. Nature **439**(7079), 957–960 (2006)
37. B. Kiefer, D.C. Lagoudas, Magnetic field-induced martensitic variant reorientation in magnetic shape memory alloys. Philos. Mag. **85**(33–35), 4289–4329 (2005)

38. I. Karaman, B. Basaran, H.E. Karaca, A.I. Karsilayan, Y.I. Chumlyakov, Energy harvesting using martensite variant reorientation mechanism in a NiMnGa magnetic shape memory alloy. Appl. Phys. Lett. **90**(17), 172505 (2007)
39. A. Lendlein, S. Kelch, Shape-memory polymers. Angew. Chem. Int. Ed. **41**(12), 2034–2057 (2002)
40. P.T. Mather, X. Luo, I.A. Rousseau, Shape memory polymer research. Annu. Rev. Mater. Res. **39**, 445–471 (2009)
41. Y. Liu, H. Du, L. Liu, J. Leng, Shape memory polymers and their composites in aerospace applications: a review. Smart Mater. Struct. **23**(2), 023001 (2014)
42. M.D. Hager, S. Bode, C. Weber, U.S. Schubert, Shape memory polymers: past, present and future developments. Prog. Polym. Sci. **49–50**, 3–33 (2015)
43. J.N. Reddy, *An Introduction to the Finite Element Method*, vol. 2 (McGraw-Hill, New York, 1993)
44. W. Cheney, D. Kincaid, *Numerical Analysis. Mathematics of Scientific Computing* (Brooks & Cole Publishing Company, Pacific Grove, 1996)
45. J. Solomon, *Numerical Algorithms: Methods for Computer Vision, Machine Learning, and Graphics* (CRC Press, Boca Raton, 2015)
46. E.A. Peraza Hernandez, B. Kiefer, D.J. Hartl, A. Menzel, D.C. Lagoudas, Analytical investigation of structurally stable configurations in shape memory alloy-actuated plates. Int. J. Solids Struct. **69**, 442–458 (2015)
47. D.J. Hartl, D.C. Lagoudas, Constitutive modeling and structural analysis considering simultaneous phase transformation and plastic yield in shape memory alloys. Smart Mater. Struct. **18**(10), 104017 (2009)
48. E. Peraza Hernandez, D. Hartl, E. Akleman, D. Lagoudas, Modeling and analysis of origami structures with smooth folds. Comput. Aided Des. **78**, 93–106 (2016)
49. E.A. Peraza Hernandez, D.J. Hartl, A. Kotz, R.J. Malak, Design and optimization of an SMA-based self-folding structural sheet with sparse insulating layers, in *Proceedings of the ASME 2014 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, SMASIS2014–7540 (American Society of Mechanical Engineers, New York, 2014)
50. E. Peraza Hernandez, D. Hartl, R. Malak, D. Lagoudas, Analysis and optimization of a shape memory alloy-based self-folding sheet considering material uncertainties, in *Proceedings of the ASME 2015 Conference on Smart Materials, Adaptive Structures and Intelligent Systems SMASIS*, No. SMASIS2015–9001 (American Society of Mechanical Engineers, New York, 2015), pp. V001T01A013
51. MathWorks, MATLAB's `fmincon`. http://www.mathworks.com/help/optim/ug/fmincon.html
52. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Design and simulation of origami structures with smooth folds. Proc. R. Soc. A **473**(2200), 20160716 (2017)
53. E.A. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Analysis and design of an active self-folding antenna, in *ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, page V05BT08A049 (American Society of Mechanical Engineers, New York, 2017)
54. S. Yao, X. Liu, S.V. Georgakopoulos, M.M. Tentzeris, A novel reconfigurable origami spring antenna, in *Proceedings of the 2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)* (IEEE, Piscataway, 2014), pp. 374–375
55. X. Liu, S. Yao, S.V. Georgakopoulos, B.S. Cook, M.M. Tentzeris, Reconfigurable helical antenna based on an origami structure for wireless communication system, in *Proceedings of the 2014 IEEE MTT-S International Microwave Symposium (IMS)* (IEEE, Piscataway, 2014), pp. 1–4
56. X. Liu, S. Yao, B.S. Cook, M.M. Tentzeris, S.V. Georgakopoulos, An origami reconfigurable axial-mode bifilar helical antenna. IEEE Trans. Antennas Propag. **63**(12), 5897–5903 (2015)
57. X. Liu, S.V. Georgakopoulos, M. Tentzeris, A novel mode and frequency reconfigurable origami quadrifilar helical antenna, in *2015 IEEE 16th Annual Wireless and Microwave Technology Conference (WAMICON)* (IEEE, Piscataway, 2015), pp. 1–3

58. X. Liu, S. Yao, S. V. Georgakopoulos, M. Tentzeris, Origami quadrifilar helix antenna in UHF band, in *Proceedings of the 2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)* (IEEE, Piscataway, 2014), pp. 372–373
59. X. Liu, S. Yao, S.V. Georgakopoulos, A frequency tunable origami spherical helical antenna, in *2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting* (IEEE, Piscataway, 2017), pp. 1361–1362
60. S. Yao, S.V. Georgakopoulos, B. Cook, M. Tentzeris, A novel reconfigurable origami accordion antenna, in *Proceedings of the 2014 IEEE MTT-S International Microwave Symposium (IMS)* (IEEE, Piscataway, 2014), pp. 1–4
61. S. Yao, X. Liu, S.V. Georgakopoulos, M.M. Tentzeris, A novel tunable origami accordion antenna, in *Proceedings of the 2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)* (IEEE, Piscataway, 2014), pp. 370–371
62. K. Fuchi, A.R. Diaz, E.J. Rothwell, R.O. Ouedraogo, J. Tang, An origami tunable metamaterial. J. Appl. Physiol. **111**(8), 084905 (2012)
63. K. Fuchi, J. Tang, B. Crowgey, A.R. Diaz, E.J. Rothwell, R.O. Ouedraogo, Origami tunable frequency selective surfaces. IEEE Antennas Wirel. Propag. Lett. **11**, 473–475 (2012)
64. S.R. Seiler, G. Bazzan, K. Fuchi, E.J. Alanyak, A.S. Gillman, G.W. Reich, P.R. Buskohl, S. Pallampati, D. Sessions, D. Grayson, G.H. Huff, Physical reconfiguration of an origami-inspired deployable microstrip patch antenna array, in *2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting* (IEEE, Piscataway, 2017), pp. 2359–2360
65. G.J. Hayes, Y. Liu, J. Genzer, G. Lazzi, M.D. Dickey, Self-folding origami microstrip antennas. IEEE Trans. Antennas Propag. **62**(10), 5416–5419 (2014)
66. M. Nogi, N. Komoda, K. Otsuka, K. Suganuma, Foldable nanopaper antennas for origami electronics. Nanoscale **5**(10), 4395–4399 (2013)
67. I. Toshiyuki, O. Naokazu, H. Takaya, A folding parabola antenna with flat facets. J. Natl. Inst. Inf. Commun. Technol. **50**(3–4), 177–181 (2003)
68. X. Liu, S. Yao, S.V. Georgakopoulos, Mode reconfigurable bistable spiral antenna based on kresling origami, in *2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting* (IEEE, Piscataway, 2017), pp. 413–414
69. W. Su, R. Bahr, S.A. Nauroze, M.M. Tentzeris, Novel 3D-printed "Chinese fan" bow-tie antennas for origami/shape-changing configurations, in *2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting* (IEEE, Piscataway, 2017), pp. 1245–1246
70. K. Fuchi, G. Bazzan, A.S. Gillman, G.H. Huff, P.R. Buskohl, E.J. Alyanak, Frequency tuning through physical reconfiguration of a corrugated origami frequency selective surface, in *2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting* (IEEE, Piscataway, 2017), pp. 411–412
71. T. Tachi, Simulation of rigid origami, in *Origami 4, Fourth International Meeting of Origami Science, Mathematics, and Education* (2009), pp. 175–187
72. S.-M. Belcastro, T.C. Hull, Modelling the folding of paper into three dimensions using affine transformations. Linear Algebra Appl. **348**(1–3), 273–282 (2002)
73. S.-W. Qu, C.-L. Ruan, Effect of round corners on bowtie antennas. Prog. Electromagn. Res. **57**, 179–195 (2006)
74. J.F. Sauder, M.W. Thomson, Ka-band parabolic deployable antenna (KaPDA) enabling high speed data communication for CubeSats, in *AIAA SPACE 2015 Conference and Exposition* (2015), p. 4425
75. J.F. Sauder, N. Chahat, B. Hirsch, R. Hodges, Y. Rahmat-Samii, E. Peral, M.W. Thomson, From prototype to flight: qualifying a Ka-band parabolic deployable antenna (KaPDA) for CubeSats, in *4th AIAA Spacecraft Structures Conference* (2017), p. 0620

76. P. Agrawal, M. Anderson, M. Card, Preliminary design of large reflectors with flat facets. IEEE Trans. Antennas Propag. **29**(4), 688–694 (1981)
77. E. Peraza Hernandez, D.J. Hartl, D.C. Lagoudas, Modeling and design of shape memory alloy-based origami structures with smooth folds, in *25th AIAA/AHS Adaptive Structures Conference* (2017), p. 1875
78. L. Eriksson, E. Johansson, N. Kettaneh-Wold, C. Wikström, S. Wold, *Design of Experiments: Principles and Applications* (Umetrics Academy, Umeå, 2000)

Appendix A
Notation and Useful Formulas

This appendix serves as a guide to and review of the notation, mathematical concepts, and formulas used in the development of the models and design methods presented in this book. It is assumed that the reader has previously completed an introductory linear algebra course and is familiar with the properties of operators such as dot product, cross product, determinant, etc. For the sake of brevity, we do not provide derivations of any particular formula reviewed in this appendix. Such derivations can be found in other sources [1–3].

A.1 Vectors in Three-Dimensional Space

First, let us review mathematical concepts and formulas concerning vectors in the three-dimensional space of real numbers ($\mathbb{R}^3$). The vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \in \mathbb{R}^3$ form the basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ that defines the Cartesian coordinate system of $\mathbb{R}^3$. These vectors are defined as follows:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{A.1}$$

The axes aligned with $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ are respectively denoted as the 1-*axis*, the 2-*axis*, and the 3-*axis* as indicated in Fig. A.1. The components of a vector $\mathbf{y} \in \mathbb{R}^3$ in the coordinate system defined by $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ are denoted y_1, y_2, and y_3 (see Fig. A.1):

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2

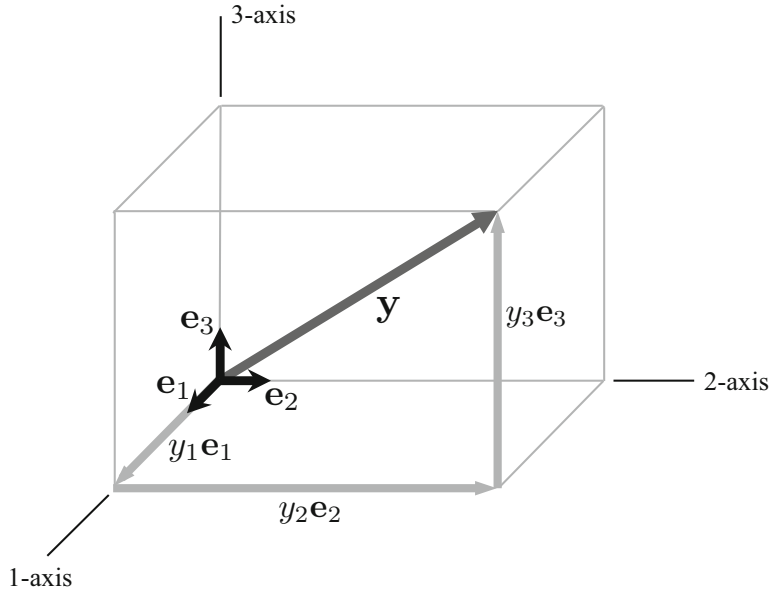


Fig. A.1 Vector $\mathbf{y} \in \mathbb{R}^3$ in the coordinate system defined by $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$

$$\begin{aligned}\mathbf{y} &= y_1\mathbf{e}_1 + y_2\mathbf{e}_2 + y_3\mathbf{e}_3 \\ &= y_1\begin{bmatrix}1\\0\\0\end{bmatrix} + y_2\begin{bmatrix}0\\1\\0\end{bmatrix} + y_3\begin{bmatrix}0\\0\\1\end{bmatrix} = \begin{bmatrix}y_1\\y_2\\y_3\end{bmatrix}.\end{aligned} \tag{A.2}$$

The *dot product* of two vectors $\mathbf{y} \in \mathbb{R}^3$ and $\mathbf{z} \in \mathbb{R}^3$ is denoted $\mathbf{y} \cdot \mathbf{z} \in \mathbb{R}$ and is determined as follows:

$$\mathbf{y} \cdot \mathbf{z} = \begin{bmatrix}y_1\\y_2\\y_3\end{bmatrix} \cdot \begin{bmatrix}z_1\\z_2\\z_3\end{bmatrix} = y_1z_1 + y_2z_2 + y_3z_3. \tag{A.3}$$

The *magnitude* of a vector $\mathbf{y} \in \mathbb{R}^3$ is denoted $\|\mathbf{y}\| \in \mathbb{R}$ and is determined as follows:

$$\|\mathbf{y}\| = (\mathbf{y} \cdot \mathbf{y})^{\frac{1}{2}} = \left(y_1^2 + y_2^2 + y_3^2\right)^{\frac{1}{2}}. \tag{A.4}$$

A vector $\mathbf{y} \in \mathbb{R}^3$ is said to be a *unit vector* if its magnitude is 1 (i.e., if $\|\mathbf{y}\| = 1$).

Example A.1
Statement: Consider the vectors $\mathbf{y} \in \mathbb{R}^3$ and $\mathbf{z} \in \mathbb{R}^3$ having components given as follows:

$$\mathbf{y} = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}, \qquad \mathbf{z} = \begin{bmatrix} 2 \\ -2 \\ 3 \end{bmatrix}. \tag{A.5}$$

Determine (a) the dot product $\mathbf{y} \cdot \mathbf{z}$ and (b) the magnitude of $\mathbf{y}$ and $\mathbf{z}$.
Solution:

(a) The dot product $\mathbf{y} \cdot \mathbf{z}$ is determined using (A.3):

$$\mathbf{y} \cdot \mathbf{z} = y_1 z_1 + y_2 z_2 + y_3 z_3 = (1)(2) + (2)(-2) + (2)(3) = 4. \tag{A.6}$$

(b) The magnitude of $\mathbf{y}$ is determined using (A.4):

$$\|\mathbf{y}\| = \left(y_1^2 + y_2^2 + y_3^2\right)^{\frac{1}{2}} = \left(1^2 + 2^2 + 2^2\right)^{\frac{1}{2}} = 3, \tag{A.7}$$

and the magnitude of $\mathbf{z}$ is also determined using (A.4):

$$\|\mathbf{z}\| = \left(z_1^2 + z_2^2 + z_3^2\right)^{\frac{1}{2}} = \left(2^2 + (-2)^2 + 3^2\right)^{\frac{1}{2}} = 17^{\frac{1}{2}}. \tag{A.8}$$

The angle between two vectors $\mathbf{y} \in \mathbb{R}^3$ and $\mathbf{z} \in \mathbb{R}^3$ is denoted $\phi \in [0, \pi]$ and can be determined implicitly using the following formula:

$$\cos(\phi) = \frac{\mathbf{y} \cdot \mathbf{z}}{\|\mathbf{y}\| \, \|\mathbf{z}\|}. \tag{A.9}$$

The angle ϕ is illustrated in Fig. A.2. The vectors $\mathbf{y}$ and $\mathbf{z}$ are said to be *orthogonal* if their dot product is 0 (i.e., if $\mathbf{y} \cdot \mathbf{z} = 0$). Geometrically, this means that the vectors $\mathbf{y}$ and $\mathbf{z}$ are perpendicular as illustrated in Fig. A.2b (i.e., they meet at an angle of $\phi = \pi/2$, c.f. (A.9)).

The *cross product* of two vectors $\mathbf{y} \in \mathbb{R}^3$ and $\mathbf{z} \in \mathbb{R}^3$ is denoted $\mathbf{y} \times \mathbf{z} \in \mathbb{R}^3$, which is a vector orthogonal to $\mathbf{y}$ and $\mathbf{z}$ that has components given as follows:

$$\mathbf{y} \times \mathbf{z} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \times \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} y_2 z_3 - y_3 z_2 \\ y_3 z_1 - y_1 z_3 \\ y_1 z_2 - y_2 z_1 \end{bmatrix}. \tag{A.10}$$

The reader can verify that $\mathbf{y} \times \mathbf{z}$ is orthogonal to $\mathbf{y}$ and $\mathbf{z}$ (i.e., $(\mathbf{y} \times \mathbf{z}) \cdot \mathbf{y} = 0$ and $(\mathbf{y} \times \mathbf{z}) \cdot \mathbf{z} = 0$).

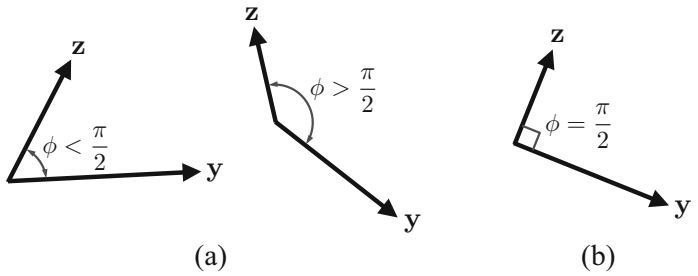


Fig. A.2 (**a**) Vectors **y** and **z** and angle ϕ between them. (**b**) Special case where the vectors **y** and **z** are *orthogonal*

Example A.2
Statement: Consider the vectors $\mathbf{y} \in \mathbb{R}^3$ and $\mathbf{z} \in \mathbb{R}^3$ having components given as follows:

$$\mathbf{y} = \begin{bmatrix} -1 \\ 0 \\ 3\frac{1}{2} \end{bmatrix}, \qquad \mathbf{z} = \begin{bmatrix} 4 \\ 0 \\ 0 \end{bmatrix}. \tag{A.11}$$

Determine (a) the angle between **y** and **z** and (b) the cross product $\mathbf{y} \times \mathbf{z}$.
Solution:

(a) The angle ϕ between **y** and **z** is determined using (A.9):

$$\cos(\phi) = \frac{\mathbf{y} \cdot \mathbf{z}}{\|\mathbf{y}\| \, \|\mathbf{z}\|} = \frac{-4}{(2)(4)} = -\frac{1}{2}, \tag{A.12}$$

and thus:

$$\phi = \cos^{-1}\left(-\frac{1}{2}\right) = \frac{2\pi}{3}. \tag{A.13}$$

(b) The cross product $\mathbf{y} \times \mathbf{z}$ is determined using (A.10):

$$\mathbf{y} \times \mathbf{z} = \begin{bmatrix} y_2 z_3 - y_3 z_2 \\ y_3 z_1 - y_1 z_3 \\ y_1 z_2 - y_2 z_1 \end{bmatrix} = \begin{bmatrix} (0)(0) - (3\frac{1}{2})(0) \\ (3\frac{1}{2})(4) - (-1)(0) \\ (-1)(0) - (0)(4) \end{bmatrix} = \begin{bmatrix} 0 \\ 4(3\frac{1}{2}) \\ 0 \end{bmatrix}. \tag{A.14}$$

The *span* of vectors $\mathbf{z}^1, \ldots, \mathbf{z}^M \in \mathbb{R}^3$ is denoted $\text{span}(\mathbf{z}^1, \ldots, \mathbf{z}^M) \subseteq \mathbb{R}^3$ and is the set of all linear combinations of such vectors:

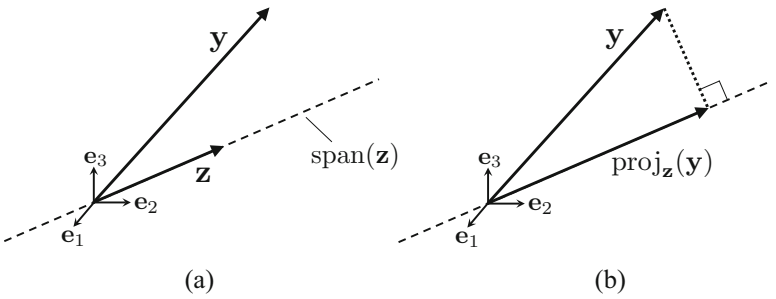


Fig. A.3 Illustration of the orthogonal vector projection of a vector $\mathbf{y} \in \mathbb{R}^3$ onto span($\mathbf{z}$). (**a**) A vector $\mathbf{y}$ and span($\mathbf{z}$). (**b**) Orthogonal vector projection $\text{proj}_{\mathbf{z}}(\mathbf{y}) \in \mathbb{R}^3$ of $\mathbf{y}$ onto span($\mathbf{z}$)

$$\text{span}(\mathbf{z}^1, \ldots, \mathbf{z}^M) = \left\{ \sum_{i=1}^{M} a_i \mathbf{z}^i \;\middle|\; a_1, \ldots, a_M \in \mathbb{R} \right\}. \tag{A.15}$$

The span of a single vector $\mathbf{z} \in \mathbb{R}^3$ can be visualized as a straight line in three-dimensional space that intersects the origin and is aligned with $\mathbf{z}$ (see Fig. A.3a). The *orthogonal vector projection* of a vector $\mathbf{y} \in \mathbb{R}^3$ onto span($\mathbf{z}$) is denoted $\text{proj}_{\mathbf{z}}(\mathbf{y}) \in \mathbb{R}^3$ and can be determined using the following formula (see Fig. A.3b):

$$\text{proj}_{\mathbf{z}}(\mathbf{y}) = \frac{\mathbf{y} \cdot \mathbf{z}}{\|\mathbf{z}\|} \frac{\mathbf{z}}{\|\mathbf{z}\|} = \frac{\mathbf{y} \cdot \mathbf{z}}{\mathbf{z} \cdot \mathbf{z}} \mathbf{z}. \tag{A.16}$$

The span of two orthogonal vectors $\mathbf{z}^1 \in \mathbb{R}^3$ and $\mathbf{z}^2 \in \mathbb{R}^3$ ($\mathbf{z}^1 \cdot \mathbf{z}^2 = 0$) can be visualized as a plane in three-dimensional space that intersects the origin and has a normal vector $\mathbf{z}^1 \times \mathbf{z}^2$. The *orthogonal vector projection* of a vector $\mathbf{y} \in \mathbb{R}^3$ onto span($\mathbf{z}^1, \mathbf{z}^2$) is denoted $\text{proj}_{\mathbf{z}^1, \mathbf{z}^2}(\mathbf{y}) \in \mathbb{R}^3$ and can be determined as follows:

$$\begin{aligned} \text{proj}_{\mathbf{z}^1, \mathbf{z}^2}(\mathbf{y}) &= \frac{\mathbf{y} \cdot \mathbf{z}^1}{\|\mathbf{z}^1\|} \frac{\mathbf{z}^1}{\|\mathbf{z}^1\|} + \frac{\mathbf{y} \cdot \mathbf{z}^2}{\|\mathbf{z}^2\|} \frac{\mathbf{z}^2}{\|\mathbf{z}^2\|} \\ &= \frac{\mathbf{y} \cdot \mathbf{z}^1}{\mathbf{z}^1 \cdot \mathbf{z}^1} \mathbf{z}^1 + \frac{\mathbf{y} \cdot \mathbf{z}^2}{\mathbf{z}^2 \cdot \mathbf{z}^2} \mathbf{z}^2. \end{aligned} \tag{A.17}$$

Figure A.4 illustrates the orthogonal vector projection of a vector $\mathbf{y} \in \mathbb{R}^3$ onto span($\mathbf{e}_1, \mathbf{e}_2$) that can be determined by substituting $\mathbf{z}^1 = \mathbf{e}_1$ and $\mathbf{z}^2 = \mathbf{e}_2$ in (A.17).

Example A.3
Statement: Consider the vectors $\mathbf{y} \in \mathbb{R}^3$, $\mathbf{z}^1 \in \mathbb{R}^3$, and $\mathbf{z}^2 \in \mathbb{R}^3$ having components given as follows:

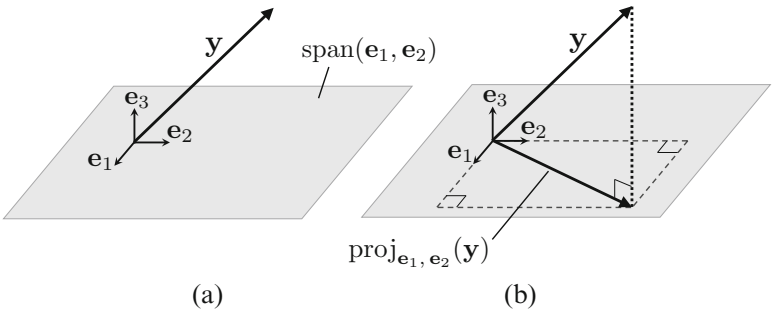


Fig. A.4 Illustration of the orthogonal vector projection of a vector $\mathbf{y} \in \mathbb{R}^3$ onto $\text{span}(\mathbf{e}_1, \mathbf{e}_2)$. (**a**) Vector $\mathbf{y}$ and $\text{span}(\mathbf{e}_1, \mathbf{e}_2)$. (**b**) Orthogonal vector projection $\text{proj}_{\mathbf{e}_1, \mathbf{e}_2}(\mathbf{y}) \in \mathbb{R}^3$ of $\mathbf{y}$ onto $\text{span}(\mathbf{e}_1, \mathbf{e}_2)$

$$\mathbf{y} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \qquad \mathbf{z}^1 = \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix}, \qquad \mathbf{z}^2 = \begin{bmatrix} 0 \\ -2^{\frac{1}{2}} \\ 2^{\frac{1}{2}} \end{bmatrix}. \tag{A.18}$$

Determine (a) the orthogonal vector projection of $\mathbf{y}$ onto $\text{span}(\mathbf{z}^1)$ and (b) the orthogonal vector projection of $\mathbf{y}$ onto $\text{span}(\mathbf{z}^1, \mathbf{z}^2)$.
Solution:

(a) The orthogonal vector projection of $\mathbf{y}$ onto $\text{span}(\mathbf{z}^1)$ denoted $\text{proj}_{\mathbf{z}^1}(\mathbf{y})$ is determined using (A.16):

$$\text{proj}_{\mathbf{z}^1}(\mathbf{y}) = \frac{\mathbf{y} \cdot \mathbf{z}^1}{\mathbf{z}^1 \cdot \mathbf{z}^1} \mathbf{z}^1 = -\frac{1}{9} \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix}. \tag{A.19}$$

(b) The orthogonal vector projection of $\mathbf{y}$ onto $\text{span}(\mathbf{z}^1, \mathbf{z}^2)$ denoted $\text{proj}_{\mathbf{z}^1, \mathbf{z}^2}(\mathbf{y})$ is determined using (A.17):

$$\begin{aligned} \text{proj}_{\mathbf{z}^1, \mathbf{z}^2}(\mathbf{y}) &= \frac{\mathbf{y} \cdot \mathbf{z}^1}{\mathbf{z}^1 \cdot \mathbf{z}^1} \mathbf{z}^1 + \frac{\mathbf{y} \cdot \mathbf{z}^2}{\mathbf{z}^2 \cdot \mathbf{z}^2} \mathbf{z}^2 \\ &= -\frac{1}{9} \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix} + \frac{2^{\frac{3}{2}}}{4} \begin{bmatrix} 0 \\ -2^{\frac{1}{2}} \\ 2^{\frac{1}{2}} \end{bmatrix} \\ &= -\frac{1}{9} \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} = \frac{1}{9} \begin{bmatrix} 1 \\ -11 \\ 7 \end{bmatrix}. \end{aligned} \tag{A.20}$$

A.2 Vectors of Arbitrary Dimensions

Let us now review mathematical concepts and formulas concerning vectors in the space $\mathbb{R}^N$ (where N is an arbitrary positive integer) that are extensively used in the various chapters of this book. A vector $\mathbf{y} \in \mathbb{R}^N$ having components $y_1, \ldots, y_N$ can be expressed as follows:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}. \tag{A.21}$$

The *zero vector* is denoted $\mathbf{0}_N \in \mathbb{R}^N$ and all of its components are zeros:

$$\mathbf{0}_N = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{A.22}$$

The formulations for the *dot product* and *magnitude* of vectors in $\mathbb{R}^3$ provided in (A.3) and (A.4) can be generalized for any two vectors $\mathbf{y} \in \mathbb{R}^N$ and $\mathbf{z} \in \mathbb{R}^N$:

$$\mathbf{y} \cdot \mathbf{z} \;=\; y_1 z_1 + y_2 z_2 + \cdots + y_N z_N \;=\; \sum_{i=1}^{N} y_i z_i, \tag{A.23}$$

$$\begin{aligned} \|\mathbf{y}\| &= (\mathbf{y} \cdot \mathbf{y})^{\frac{1}{2}} \\ &= \left(y_1^2 + y_2^2 + \cdots + y_N^2\right)^{\frac{1}{2}} \;=\; \left(\sum_{i=1}^{N} y_i^2\right)^{\frac{1}{2}}. \end{aligned} \tag{A.24}$$

The concept of orthogonal vector projection introduced for three-dimensional vectors in Sect. A.1 can also be generalized for vectors in $\mathbb{R}^N$. Let $\mathcal{V}$ be a subspace of $\mathbb{R}^N$ and let $\text{proj}_{\mathcal{V}}(\mathbf{y}) \in \mathbb{R}^N$ be the *orthogonal vector projection* of a vector $\mathbf{y} \in \mathbb{R}^N$ onto $\mathcal{V}$. The vector $\mathbf{y}$ can be decomposed as follows:

$$\mathbf{y} \;=\; \text{proj}_{\mathcal{V}}(\mathbf{y}) \;+\; \big(\mathbf{y} \;-\; \text{proj}_{\mathcal{V}}(\mathbf{y})\big). \tag{A.25}$$

The *orthogonal vector projection* $\text{proj}_{\mathcal{V}}(\mathbf{y})$ satisfies the following criteria:

$$\text{proj}_{\mathcal{V}}(\mathbf{y}) \in \mathcal{V}, \tag{A.26}$$

$$\big(\mathbf{y} \;-\; \text{proj}_{\mathcal{V}}(\mathbf{y})\big) \cdot \mathbf{z} = 0 \qquad \mathbf{z} \in \mathcal{V}, \tag{A.27}$$

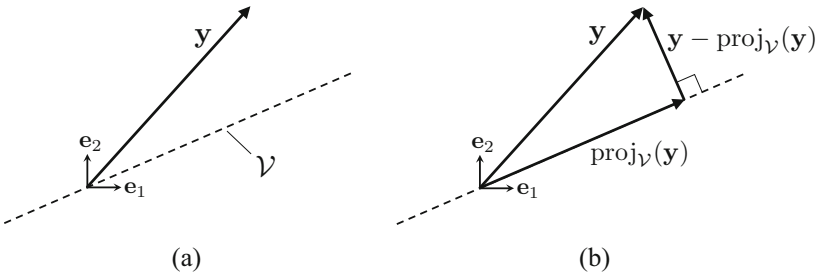


Fig. A.5 Illustration of the orthogonal vector projection of a vector $\mathbf{y} \in \mathbb{R}^2$ onto a subspace $\mathcal{V} \subset \mathbb{R}^2$. (**a**) Vector $\mathbf{y}$ and subspace $\mathcal{V}$. (**b**) Orthogonal vector projection $\text{proj}_{\mathcal{V}}(\mathbf{y}) \in \mathbb{R}^2$ of $\mathbf{y}$ onto $\mathcal{V}$

i.e., $\text{proj}_{\mathcal{V}}(\mathbf{y})$ *is contained in* $\mathcal{V}$ and $(\mathbf{y} - \text{proj}_{\mathcal{V}}(\mathbf{y}))$ *is orthogonal to any vector* $\mathbf{z} \in \mathcal{V}$. The concept of orthogonal vector projection is illustrated in Fig. A.5 for a vector $\mathbf{y} \in \mathbb{R}^2$.

If the subspace $\mathcal{V} \subseteq \mathbb{R}^N$ is represented as the span of pairwise orthogonal vectors $\mathbf{z}^1, \ldots, \mathbf{z}^M$ (i.e., if $\mathcal{V} = \text{span}(\mathbf{z}^1, \ldots, \mathbf{z}^M)$ where $M \leq N$), $\text{proj}_{\mathcal{V}}(\mathbf{y})$ can be determined using the following formula:

$$\text{proj}_{\mathcal{V}}(\mathbf{y}) = \text{proj}_{\mathbf{z}^1, \ldots, \mathbf{z}^M}(\mathbf{y}) = \sum_{i=1}^{M} \frac{\mathbf{y} \cdot \mathbf{z}^i}{\|\mathbf{z}^i\|} \frac{\mathbf{z}^i}{\|\mathbf{z}^i\|} = \sum_{i=1}^{M} \frac{\mathbf{y} \cdot \mathbf{z}^i}{\mathbf{z}^i \cdot \mathbf{z}^i} \mathbf{z}^i, \tag{A.28}$$

cf. (A.16) and (A.17).

A.3 Matrices of Arbitrary Dimensions

A matrix $\mathbf{Y} \in \mathbb{R}^{M \times N}$ with components Y_{ij}, $i = 1, \ldots, M$, $j = 1, \ldots, N$, can be expressed as follows:

$$\mathbf{Y} = \begin{bmatrix} Y_{11} & \cdots & Y_{1N} \\ \vdots & \ddots & \vdots \\ Y_{M1} & \cdots & Y_{MN} \end{bmatrix}. \tag{A.29}$$

A matrix $\mathbf{Y} \in \mathbb{R}^{M \times N}$ is said to be a *square matrix* if $M = N$ (i.e., $\mathbf{Y} \in \mathbb{R}^{M \times M}$). The *determinant* of a square matrix $\mathbf{Y} \in \mathbb{R}^{M \times M}$ is denoted $\det(\mathbf{Y}) \in \mathbb{R}$.

The *transpose* of a matrix $\mathbf{Y} \in \mathbb{R}^{M \times N}$ is denoted $\mathbf{Y}^\top \in \mathbb{R}^{N \times M}$ and has components given as follows:

$$\left[\mathbf{Y}^\top\right]_{ji} = Y_{ij} \qquad i = 1, \ldots, M, \quad j = 1, \ldots, N, \tag{A.30}$$

therefore it can be expressed as follows:

$$\mathbf{Y}^{\top} = \begin{bmatrix} Y_{11} & \cdots & Y_{M1} \\ \vdots & \ddots & \vdots \\ Y_{1N} & \cdots & Y_{MN} \end{bmatrix}. \tag{A.31}$$

A matrix $\mathbf{Y}$ is said to be *symmetric* if $\mathbf{Y} = \mathbf{Y}^{\top}$.

The *matrix multiplication* of a matrix $\mathbf{Y} \in \mathbb{R}^{M \times N}$ and a matrix $\mathbf{Z} \in \mathbb{R}^{N \times P}$ is denoted $\mathbf{Y}\,\mathbf{Z} \in \mathbb{R}^{M \times P}$ and has components given as follows:

$$[\mathbf{Y}\,\mathbf{Z}]_{ik} = \sum_{j=1}^{N} Y_{ij} Z_{jk} \qquad i = 1, \ldots, M, \quad k = 1, \ldots, P. \tag{A.32}$$

The *identity matrix* is a square matrix denoted $\mathbf{I}_M \in \mathbb{R}^{M \times M}$ and its components are defined as follows:

$$[\mathbf{I}_M]_{ij} = \begin{cases} 1; & i = j \\ 0; & i \neq j \end{cases} \qquad i = 1, \ldots, M, \quad j = 1, \ldots, M, \tag{A.33}$$

therefore it can be expressed as follows:

$$\mathbf{I}_M = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{A.34}$$

For any matrix $\mathbf{Y} \in \mathbb{R}^{M \times N}$, we have that:

$$\mathbf{I}_M\,\mathbf{Y} \;=\; \mathbf{Y}\,\mathbf{I}_N \;=\; \mathbf{Y}. \tag{A.35}$$

The *inverse* of a square matrix $\mathbf{Y} \in \mathbb{R}^{M \times M}$ is denoted $\mathbf{Y}^{-1} \in \mathbb{R}^{M \times M}$ and satisfies the following criteria:

$$\mathbf{Y}\,\mathbf{Y}^{-1} \;=\; \mathbf{Y}^{-1}\,\mathbf{Y} \;=\; \mathbf{I}_M. \tag{A.36}$$

A square matrix $\mathbf{Y} \in \mathbb{R}^{M \times M}$ has an inverse if and only if $\det(\mathbf{Y}) \neq 0$.

The *Moore-Penrose pseudoinverse* of a matrix $\mathbf{Y} \in \mathbb{R}^{M \times N}$ is denoted $\mathbf{Y}^{\dagger} \in \mathbb{R}^{N \times M}$ and satisfies the following four criteria:

$$\begin{aligned} \mathbf{Y}\,\mathbf{Y}^{\dagger}\,\mathbf{Y} &= \mathbf{Y}, \\ \mathbf{Y}^{\dagger}\,\mathbf{Y}\,\mathbf{Y}^{\dagger} &= \mathbf{Y}^{\dagger}, \\ (\mathbf{Y}^{\dagger}\,\mathbf{Y})^{\top} &= \mathbf{Y}^{\dagger}\,\mathbf{Y}, && \mathbf{Y}^{\dagger}\,\mathbf{Y} \text{ is symmetric}, \\ (\mathbf{Y}\,\mathbf{Y}^{\dagger})^{\top} &= \mathbf{Y}\,\mathbf{Y}^{\dagger}, && \mathbf{Y}\,\mathbf{Y}^{\dagger} \text{ is symmetric}. \end{aligned} \tag{A.37}$$

The description of methods for the calculation of the pseudoinverse of a matrix $\mathbf{Y}$ is beyond the scope of this appendix. For such information, the reader is referred to [2, 4].

The multiplication of a matrix $\mathbf{Y} \in \mathbb{R}^{M\times N}$ and a vector $\mathbf{z} \in \mathbb{R}^{N}$ having components $z_1, \ldots, z_N$ is denoted $\mathbf{Y}\,\mathbf{z} \in \mathbb{R}^{M}$ and has components given as follows:

$$[\mathbf{Y}\,\mathbf{z}]_i = \sum_{j=1}^{N} Y_{ij} z_j \qquad i = 1, \ldots, M. \tag{A.38}$$

Two important vector subspaces associated with a matrix $\mathbf{Y} \in \mathbb{R}^{M\times N}$ are its *range* and *null space*. The *range* of a matrix $\mathbf{Y} \in \mathbb{R}^{M\times N}$ is denoted $\mathrm{Ran}(\mathbf{Y}) \subseteq \mathbb{R}^{M}$ and is defined as follows:

$$\mathrm{Ran}(\mathbf{Y}) = \{\mathbf{b} \in \mathbb{R}^{M} \mid \text{ there exists } \mathbf{z} \in \mathbb{R}^{N} \text{ such that } \mathbf{b} = \mathbf{Y}\,\mathbf{z}\}. \tag{A.39}$$

The *null space* of a matrix $\mathbf{Y} \in \mathbb{R}^{M\times N}$ is denoted $\mathrm{Nul}(\mathbf{Y}) \subseteq \mathbb{R}^{N}$ and is defined as follows:

$$\mathrm{Nul}(\mathbf{Y}) = \{\mathbf{z} \in \mathbb{R}^{N} \mid \mathbf{Y}\,\mathbf{z} = \mathbf{0}_M\}. \tag{A.40}$$

A.4 Block Matrices

A matrix $\mathbf{Y} \in \mathbb{R}^{M\times N}$ can be expressed as an $R \times S$ *block partitioned matrix*, called *block matrix* for simplicity, as follows:

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{Y}^{11} & \ldots & \boldsymbol{Y}^{1S} \\ \vdots & \ddots & \vdots \\ \boldsymbol{Y}^{R1} & \ldots & \boldsymbol{Y}^{RS} \end{bmatrix} \in \mathbb{R}^{M\times N}, \tag{A.41}$$

where: $\boldsymbol{Y}^{ij} \in \mathbb{R}^{M_i \times N_j}, \qquad i = 1, \ldots, R, \qquad j = 1, \ldots, S,$

$\sum_{i=1}^{R} M_i = M, \qquad \sum_{j=1}^{S} N_j = N.$

The ij block component of the block matrix $\mathbf{Y}$ in (A.41) is denoted as follows:

$$[\mathbf{Y}]_{ij\,\text{block}} = \boldsymbol{Y}^{ij}. \tag{A.42}$$

Let us consider two block matrices $\mathbf{Y} \in \mathbb{R}^{M\times N}$ and $\mathbf{Z} \in \mathbb{R}^{N\times P}$. The matrix $\mathbf{Y}$ is partitioned as an $R \times S$ block matrix as provided in (A.41) and the matrix $\mathbf{Z}$ is partitioned as an $S \times U$ block matrix where $[\mathbf{Z}]_{jk\,\text{block}} = \boldsymbol{Z}^{jk} \in \mathbb{R}^{N_j\times P_k}$ (where $j = 1, \ldots, S$, $k = 1, \ldots, U$, and $\sum_{j=1}^{S} N_j = N$, $\sum_{k=1}^{U} P_k = P$). The block components of the matrix multiplication of $\mathbf{Y}$ and $\mathbf{Z}$ are given as follows:

$$[\mathbf{Y}\,\mathbf{Z}]_{ik\,\text{block}} = \sum_{j=1}^{S} \boldsymbol{Y}^{ij}\,\boldsymbol{Z}^{jk} \in \mathbb{R}^{M_i\times P_k} \qquad i = 1, \ldots, R, \quad k = 1, \ldots, U. \tag{A.43}$$

Example A.4
Statement: Consider the following matrices $\mathbf{Y} \in \mathbb{R}^{4\times 4}$ and $\mathbf{Z} \in \mathbb{R}^{4\times 4}$ partitioned as 2×2 block matrices:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^\top & d \end{bmatrix}, \qquad \mathbf{Z} = \begin{bmatrix} \mathbf{E} & \mathbf{f} \\ \mathbf{g}^\top & h \end{bmatrix}, \tag{A.44}$$

where $\mathbf{A} \in \mathbb{R}^{3\times 3}$, $\mathbf{E} \in \mathbb{R}^{3\times 3}$, $\mathbf{b} \in \mathbb{R}^{3\times 1}$, $\mathbf{c} \in \mathbb{R}^{3\times 1}$, $\mathbf{f} \in \mathbb{R}^{3\times 1}$, $\mathbf{g} \in \mathbb{R}^{3\times 1}$, $d \in \mathbb{R}^{1\times 1}$, and $h \in \mathbb{R}^{1\times 1}$. Determine $\mathbf{Y}\,\mathbf{Z}$.
Solution: Using (A.43), we determine the block components of $\mathbf{Y}\,\mathbf{Z}$:

$$\begin{aligned}
[\mathbf{Y}\,\mathbf{Z}]_{11\,\text{block}} &= \mathbf{A}\,\mathbf{E} + \mathbf{b}\,\mathbf{g}^\top \in \mathbb{R}^{3\times 3}, \\
[\mathbf{Y}\,\mathbf{Z}]_{12\,\text{block}} &= \mathbf{A}\,\mathbf{f} + \mathbf{b}\,h \in \mathbb{R}^{3\times 1}, \\
[\mathbf{Y}\,\mathbf{Z}]_{21\,\text{block}} &= \mathbf{c}^\top\,\mathbf{E} + d\,\mathbf{g}^\top \in \mathbb{R}^{1\times 3}, \\
[\mathbf{Y}\,\mathbf{Z}]_{22\,\text{block}} &= \mathbf{c}^\top\,\mathbf{f} + d\,h \in \mathbb{R}^{1\times 1}.
\end{aligned} \tag{A.45}$$

Thus, $\mathbf{Y}\,\mathbf{Z}$ is given as follows:

$$\mathbf{Y}\,\mathbf{Z} = \begin{bmatrix} \mathbf{A}\,\mathbf{E} + \mathbf{b}\,\mathbf{g}^\top & \mathbf{A}\,\mathbf{f} + \mathbf{b}\,h \\ \mathbf{c}^\top\mathbf{E} + d\,\mathbf{g}^\top & \mathbf{c}^\top\mathbf{f} + d\,h \end{bmatrix}. \tag{A.46}$$

A.5 Systems of Linear Equations

In the computational implementation of the models and design methods presented in this book, we often encounter the problem of solving simultaneous linear algebraic equations of the following form:

$$\begin{aligned} A_{11}\, y_1 \;+\; A_{12}\, y_2 \;+\cdots+\; A_{1N}\, y_N \;&=\; b_1 \\ A_{21}\, y_1 \;+\; A_{22}\, y_2 \;+\cdots+\; A_{2N}\, y_N \;&=\; b_2 \\ &\vdots \\ A_{M1}\, y_1 + A_{M2}\, y_2 + \cdots + A_{MN}\, y_N &= b_M, \end{aligned} \tag{A.47}$$

where the scalars A_{ij} and b_i are given and the scalars y_j are sought ($i = 1, \ldots, M$, $j = 1, \ldots, N$). Thus, (A.47) is a *system of M linear equations in N unknowns*. The system of linear equations (A.47) can be formulated as follows:

$$\mathbf{A}\,\mathbf{y} = \mathbf{b}, \tag{A.48}$$

where:

$$\mathbf{A} = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{M1} & \cdots & A_{MN} \end{bmatrix} \in \mathbb{R}^{M\times N}, \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \in \mathbb{R}^{N}, \qquad \mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_M \end{bmatrix} \in \mathbb{R}^{M}. \tag{A.49}$$

If the number of linear equations in the system (A.48) is equal to the number of unknowns (i.e., if $M = N$ in (A.47)) and such equations are independent (i.e., if $\det(\mathbf{A}) \neq 0$), we determine the inverse of $\mathbf{A}$ and solve for $\mathbf{y}$ as follows:

$$\begin{aligned} \mathbf{A}\,\mathbf{y} &= \mathbf{b} \\ \mathbf{A}^{-1}\,\mathbf{A}\,\mathbf{y} &= \mathbf{A}^{-1}\,\mathbf{b} && \text{multiplying by } \mathbf{A}^{-1} \text{ from the left} \\ \mathbf{I}_N\,\mathbf{y} &= \mathbf{A}^{-1}\,\mathbf{b} \\ \mathbf{y} &= \mathbf{A}^{-1}\,\mathbf{b}. \end{aligned} \tag{A.50}$$

In many cases encountered in the computational implementation of the models and design methods presented in this book, the number of independent equations in the system (A.48) is not equal to the number of unknowns N. In such cases, we seek a vector $\mathbf{y}$ that minimizes the *residual error* that is defined as follows:

$$\|\mathbf{A}\,\mathbf{y} - \mathbf{b}\|. \tag{A.51}$$

The set of all the vectors that minimize the residual error (A.51) is denoted $\mathcal{Q} \subseteq \mathbb{R}^N$ and is defined as follows:

$$\mathcal{Q} = \left\{ \mathbf{y} \in \mathbb{R}^N \;\middle|\; \|\mathbf{A}\,\mathbf{y} - \mathbf{b}\| = \min_{\mathbf{z} \in \mathbb{R}^N} \|\mathbf{A}\,\mathbf{z} - \mathbf{b}\| \right\}. \tag{A.52}$$

The set $\mathcal{Q}$ is always non-empty but may contain more than one vector $\mathbf{y}$. To determine a unique vector that minimizes the residual error (A.51), we calculate the vector $\mathbf{y} \in \mathcal{Q}$ that has the lowest magnitude, which is named the *minimum norm solution* of the system (A.48). Thus, the minimum norm solution of the system (A.48) is the unique vector $\mathbf{y} \in \mathcal{Q}$ such that $\|\mathbf{y}\| = \min_{\mathbf{z} \in \mathcal{Q}} \|\mathbf{z}\|$. It is a known result in linear algebra that the minimum norm solution of the system (A.48) can be determined as follows [2, 4]:

$$\mathbf{y} = \mathbf{A}^{\dagger}\,\mathbf{b}, \tag{A.53}$$

where $\mathbf{A}^{\dagger} \in \mathbb{R}^{N \times M}$ is the *Moore-Penrose pseudoinverse* of $\mathbf{A}$ (refer to (A.37)).

Another application of the pseudoinverse that is utilized in this book is its use in the determination of the *orthogonal vector projection* of a vector $\mathbf{y} \in \mathbb{R}^N$ onto the null space of a matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$, which is denoted $\mathrm{Nul}(\mathbf{A})$ and is defined in (A.40). Let $\mathbf{y}^* \in \mathbb{R}^N$ be the orthogonal vector projection of $\mathbf{y} \in \mathbb{R}^N$ onto $\mathrm{Nul}(\mathbf{A})$ (refer to (A.25)–(A.27)). The vector $\mathbf{y}^*$ must satisfy the following two criteria:

$$\mathbf{y}^* \in \mathrm{Nul}(\mathbf{A}) \qquad \text{thus: } \mathbf{A}\,\mathbf{y}^* = \mathbf{0}_M, \tag{A.54}$$

$$\left(\mathbf{y} - \mathbf{y}^*\right) \cdot \mathbf{z} = 0 \qquad \mathbf{z} \in \mathrm{Nul}(\mathbf{A}), \tag{A.55}$$

cf. (A.26) and (A.27). The concept of orthogonal vector projection onto $\mathrm{Nul}(\mathbf{A})$ is illustrated in Fig. A.6 for a vector $\mathbf{y} \in \mathbb{R}^2$ and a matrix $\mathbf{A} \in \mathbb{R}^{M \times 2}$.

Given a vector $\mathbf{y} \in \mathbb{R}^N$ and a matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$, the *orthogonal vector projection* $\mathbf{y}^* \in \mathbb{R}^N$ of $\mathbf{y}$ onto $\mathrm{Nul}(\mathbf{A})$ can be determined using the following formula [4]:

$$\begin{aligned} \mathbf{y}^* &= \mathbf{y} - \mathbf{A}^{\dagger}\,\mathbf{A}\,\mathbf{y} \\ &= \left(\mathbf{I}_N - \mathbf{A}^{\dagger}\,\mathbf{A}\right)\mathbf{y}. \end{aligned} \tag{A.56}$$

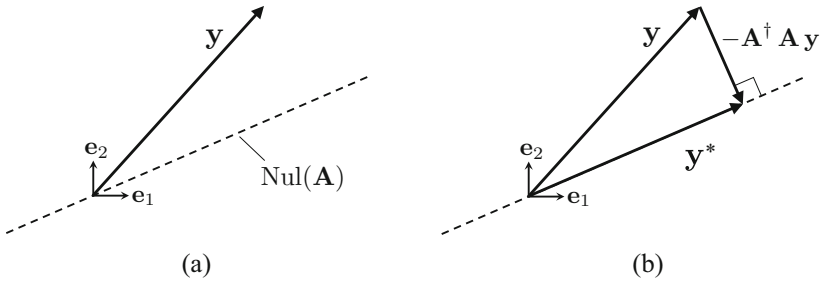


Fig. A.6 Illustration of the orthogonal vector projection of a vector $\mathbf{y} \in \mathbb{R}^2$ onto the null space of a matrix $\mathbf{A} \in \mathbb{R}^{M\times 2}$. (**a**) Vector $\mathbf{y}$ and Nul($\mathbf{A}$) $\subset \mathbb{R}^2$. (**b**) Orthogonal vector projection $\mathbf{y}^* \in \mathbb{R}^2$ of $\mathbf{y}$ onto Nul($\mathbf{A}$)

Problems

A.1 Consider the vectors $\mathbf{y}^1 \in \mathbb{R}^3$, $\mathbf{y}^2 \in \mathbb{R}^3$, and $\mathbf{y}^3 \in \mathbb{R}^3$ having components given as follows:

$$\mathbf{y}^1 = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}, \qquad \mathbf{y}^2 = \begin{bmatrix} 1 \\ -1 \\ -2\frac{1}{2} \end{bmatrix}, \qquad \mathbf{y}^3 = \begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix}. \tag{A.57}$$

Determine (a) the dot product of $\mathbf{y}^1$ and $\mathbf{y}^2$, (b) the dot product of $\mathbf{y}^2$ and $\mathbf{y}^3$, (c) the dot product of $\mathbf{y}^3$ and $\mathbf{y}^1$, (d) the magnitude of $\mathbf{y}^1$, $\mathbf{y}^2$, and $\mathbf{y}^3$, and (e) which pair(s) of vectors are orthogonal.

A.2 Repeat Problem A.1 considering the vectors $\mathbf{y}^1 \in \mathbb{R}^3$, $\mathbf{y}^2 \in \mathbb{R}^3$, and $\mathbf{y}^3 \in \mathbb{R}^3$ having components given as follows:

$$\mathbf{y}^1 = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}, \qquad \mathbf{y}^2 = \begin{bmatrix} 0 \\ -4 \\ 3 \end{bmatrix}, \qquad \mathbf{y}^3 = \begin{bmatrix} 3 \\ 1 \\ -5 \end{bmatrix}. \tag{A.58}$$

A.3 Consider the vectors $\mathbf{y}^1 \in \mathbb{R}^3$, $\mathbf{y}^2 \in \mathbb{R}^3$, and $\mathbf{y}^3 \in \mathbb{R}^3$ having components given as follows:

$$\mathbf{y}^1 = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}, \qquad \mathbf{y}^2 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \qquad \mathbf{y}^3 = \begin{bmatrix} 4 \\ 4 \\ 0 \end{bmatrix}. \tag{A.59}$$

Determine (a) the angle between $\mathbf{y}^1$ and $\mathbf{y}^2$, (b) the angle between $\mathbf{y}^2$ and $\mathbf{y}^3$, (c) the angle between $\mathbf{y}^3$ and $\mathbf{y}^1$, (d) the cross product $\mathbf{y}^1 \times \mathbf{y}^2$, (e) the cross product $\mathbf{y}^2 \times \mathbf{y}^3$, and (f) the cross product $\mathbf{y}^3 \times \mathbf{y}^1$.

A.4 Consider the vectors $\mathbf{y}^1 \in \mathbb{R}^3$, $\mathbf{y}^2 \in \mathbb{R}^3$, and $\mathbf{y}^3 \in \mathbb{R}^3$ defined in (A.59). Determine (a) the orthogonal vector projection of $\mathbf{y}^1$ onto span($\mathbf{y}^2$), (b) the orthogonal vector projection of $\mathbf{y}^1$ onto span($\mathbf{y}^3$), and (c) the orthogonal vector projection of $\mathbf{y}^1$ onto span($\mathbf{y}^2$, $\mathbf{y}^3$).

A.5 Consider the matrices $\mathbf{Y} \in \mathbb{R}^{3\times 3}$ and $\mathbf{Z} \in \mathbb{R}^{3\times 3}$ having components given as follows:

$$\mathbf{Y} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \qquad \mathbf{Z} = \begin{bmatrix} 2 & 0 & 0 \\ 4 & -1 & 3 \\ 3 & 5 & 2 \end{bmatrix}. \tag{A.60}$$

Determine (a) $\mathbf{Y}\,\mathbf{Z}$, (b) $\mathbf{Z}\,\mathbf{Y}$, (c) $\mathbf{Y}^\top\mathbf{Z}$, (d) $\mathbf{Y}\,\mathbf{Z}^\top$, and (e) $\mathbf{Y}^\top\mathbf{Z}^\top$.

A.6 Consider the matrices $\mathbf{A} \in \mathbb{R}^{2\times 3}$, $\mathbf{B} \in \mathbb{R}^{3\times 2}$ having components given as follows:

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 4 \\ 4 & 2 & 3 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 1 & 4 \\ -1 & 2 \\ 4 & 7 \end{bmatrix}. \tag{A.61}$$

In MATLAB®, the pseudoinverse of matrix $\mathbf{A}$ can be calculated using the command `pinv` as follows:

```
A = [1 -1  4;
     4  2  3]; % Defining matrix A
pseudoinverse_of_A = pinv(A); % Calculating the pseudoinverse of A
```

Determine the pseudoinverse of matrices $\mathbf{A}$ and $\mathbf{B}$ provided in (A.61) using MATLAB.

A.7 Consider the vector $\mathbf{y} \in \mathbb{R}^3$ having components given as follows:

$$\mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \tag{A.62}$$

Using MATLAB, determine the orthogonal vector projection of $\mathbf{y}$ onto the null space of the matrix $\mathbf{A}$ defined in (A.61).

Appendix B
Examples of Implementation Codes

In this appendix, we briefly describe the inputs and outputs of the MATLAB® scripts that represent the implementations of the models and design methods for origami presented in the chapters of this book. The full set of MATLAB scripts can be obtained from the Supplemental Material associated with each chapter.[1] It is hoped that the curious and engaged reader will experiment with these codes and will consider the evolution of new fold patterns into new three-dimensional forms, further advancing the theory and applicability of active origami.

B.1 Implementation of the Kinematic Model for Origami with Creased Folds (Chap. 2)

The MATLAB scripts associated with the kinematic model for origami with creased folds as presented in Chap. 2 are described here. To simulate the folding motion of an origami sheet using such a model, the user opens and executes in MATLAB the corresponding `.m` file with name starting with `Input`. As an example, consider the code presented in this section, which is an input MATLAB script used to simulate the folding motion of the origami sheet shown in Fig. 2.25 (the script is named `Input_Square1Intersection.m` in the Supplemental Material). Note that comments within the script serve to guide the user, though a complete contextual treatment of every program variable is beyond the scope of this appendix.

In the following code, we initialize the MATLAB environment in line 20 by clearing all the previously defined variables and functions from memory, closing all the open figures, and clearing the command window. In line 36, we select the increments for which the configurations are plotted for visualization. The

[1]The scripts provided in the Supplemental Material were developed and tested in MATLAB versions R2015a–R2017a. Compatibility with previous and subsequent versions is not guaranteed.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2

parameters required to define the fold pattern are provided in lines 43–97. In lines 101–144 and 166–183, we provide the algorithmic parameters required to simulate the folding motion. In lines 148–160, we provide the increments for the fold angles. In line 190, we call the function `Origami_Solver.m` that manages all the other functions required to simulate the motion of an origami sheet. Such functions include:

1. `Fold_Pattern_Data.m`: Determines the data associated with the fold pattern
2. `Reference_Plots.m`: Generates plots associated with the reference configuration and the fold pattern (Fig. B.1a, b)
3. `Kinematic_Simulation.m`: Executes the procedure for kinematic simulation presented in Sect. 2.6
4. `Rotation_Constraint.m`: Evaluates the kinematic constraint (2.39)
5. `Face_Mapping.m`: Determines the mapping between reference and current configurations for the faces using the formulation presented in Sect. 2.5
6. `Fold_Angle_Plots.m`: Generates plots of fold angle values vs. increment number (Fig. B.1c)
7. `Folded_Configurations.m`: Generates plots of configurations for the requested increments (Fig. B.1d–f)

The main numeric output variables include:

1. `O.FoldAngles`: Array of fold angle values for each increment
2. `O.exitFlag`: Indicates the reason for stopping the iterative correction procedure at each increment (1: Tolerance in residual achieved; 2: Tolerance in change in fold angles achieved; 3: Maximum number of iterations reached)
3. `O.ResidualNorm`: Magnitude of the residual vector for each increment

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % Supplemental Material
3  %
4  % ACTIVE ORIGAMI
5  % MODELING, DESIGN, AND APPLICATIONS
6  %
7  % Edwin A. Peraza Hernandez, Darren J. Hartl, Dimitris C. Lagoudas
8  %
9  % Springer
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11 %
12 % Code for Kinematic Simulation of Origami with Creased Folds
13 % Code authors: Cullen Nauck, Edwin A. Peraza Hernandez
14 % 03-2017
15 %
16 % Refer to Appendix B of the book for further information about the codes
17 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
18
19 % Initializing environment
20 clear all; close all; clc
21
22 % Main structures
23 % I: Structure with general inputs
24 % PD: Structure with geometric pattern data
```

```
% O: Structure with general outputs
% KI: Structure with kinematic simulation inputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User inputs for fold pattern
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.PlotFoldAngles: Fold angles to plot vs. increment number
I.PlotFoldAngles = [1;  2; 3; 4; 5; 6; 7; 8];

% I.PlotConfigurations: Increment number of configurations to be plotted
I.PlotConfigurations = [0; 25; 50];

% I.ReferencePoints: Reference point coordinates (in reference
%       configuration). These reference points are used as vertices or
%       additional corners for the rigid faces
% NOTE: Reference points will be labeled (1,2,3,etc.) corresponding to
%       their row index in this array
I.ReferencePoints = [
    -20 -20
    0   -20
    20  -20
    -20  0
    0    0
    20   0
    -20  20
    0    20
    20   20];

% I.FoldVectorPoints: Indices of reference points that correspond to
%       start/end vertices of each fold vector
% NOTE: Vector will run in a positive direction from the first point
%       (start) to the second point (end)
% NOTE: Vectors will be labeled (1,2,3,etc.) corresponding to their row
%       index in this array
I.FoldVectorPoints = [
    5   1
    5   2
    5   3
    5   4
    5   6
    5   7
    5   8
    5   9
    ];

%Num_Folds: Number of fold inputs
Num_Folds=size(I.FoldVectorPoints,1);

% I.IntersectionFoldIndices: Matrix with fold indices associated with each
%       interior fold intersection (in counterclockwise order)
% NOTE: Positive sign if fold vector points away from intersection;
%       Negative sign if fold vector points towards intersection
I.IntersectionFoldIndices = [
    1  2  3   5     8   7   6   4];

% I.FaceBoundary: Indices of folds and reference points defining the
%       boundary of each face (ccw order)
% NOTE: Signed depending on if fold vector points CCW (positive) around
%       face or Clockwise (negative) around face
% NOTE: If individual reference points are used to define corners of a face
%       at the boundary, the points should be referred as # of folds +
%       reference point index #)
I.FaceBoundary = [
    4   -1
    1   -2
    2   -3
```

```
92      3    -5
93      5    -8
94      8    -7
95      7    -6
96      6    -4
97      ];
98
99  % I.FixedFace: Index of face selected to be fixed throughout simulation
100 %              (one required)
101 I.FixedFace = 7;
102
103 % I.FoldAngleUpperBound: Highest allowable fold angle for each fold
104 I.FoldAngleUpperBound = [
105     pi;   pi;   pi;   pi;   pi; ...
106     pi;   pi;   pi ];
107
108 % I.FoldAngleLowerBound: Lowest allowable fold angle for each fold
109 I.FoldAngleLowerBound = [
110     -pi;  -pi;  -pi;  -pi;  -pi; ...
111     -pi;  -pi;  -pi];
112
113 % I.FoldAngleInitialValue: Initial value for each fold
114 I.FoldAngleInitialValue = [
115     0;    0;    0;    0;    0; ...
116     0;    0;    0];
117
118 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
119 % User inputs for kinematic simulation
120 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
121
122 % KI.WeightRotationConstraint: Weight for residuals from rotation
123 %       constraints
124 KI.WeightRotationConstraint = 1;
125
126 % KI.WeightFoldAngleBounds: Weight for residuals from fold angle bound
127 %       constraints
128 KI.WeightFoldAngleBounds = 1e-5;
129
130 % KI.ToleranceResidual: Tolerance for norm of residual vector
131 KI.ToleranceResidual = 1e-8;
132
133 % KI.ToleranceFoldAngle: Tolerance for correction of fold angles
134 KI.ToleranceFoldAngle = 1e-8;
135
136 % KI.MaxIterations: Maximum number of iterations allowed for each increment
137 KI.MaxIterations = 15;
138
139 % KI.MaxFoldAngleCorrection: Maximum correction allowed for individual fold
140 %       angles in each iteration
141 KI.MaxFoldAngleCorrection = 5*pi/180;
142
143 % Num_Increments: Number of increments
144 NumIncrements = 50;
145
146 % KI.FoldAngleChangePerIncrement: Initialize matrix that contains each fold
147 %       angle change for every increment
148 KI.FoldAngleChangePerIncrement = zeros( size(I.FoldVectorPoints,1), ...
        NumIncrements);
149
150 % ReferenceFoldAngle: Reference angle for filling fold angle change per
151 %       increment matrix
152 ReferenceFoldAngle = 90*pi/180;
153
154 % Fold angle change per increment matrix components
155 for i = 1:NumIncrements
156    KI.FoldAngleChangePerIncrement(1:end,i) = ...
157         ReferenceFoldAngle/NumIncrements*[
```

```
158          -1; 2; -1; 2; 2;
159          -1; 2; -1];
160  end
161
162  % KI.IncrementCorrection: Input to determine one of two methods for
163  %       calculating initial fold angle increment. One uses the previous
164  %       Jacobian (corrected; Select 1) and the other one does not (not
165  %       corrected; Select 2)
166  KI.IncrementCorrection = 1;
167
168  % KI.DisplayIncrementInfo: Input to allow for the display of iteration
169  %       information for each increment (1 to display, 0 to suppress)
170  KI.DisplayIncrementInfo = 1;
171
172  % KI.FinDiffStep: Finite difference step in fold angles for calculation of
173  %       derivatives
174  KI.FinDiffStep = 2*pi/180;
175
176  %  User input for initial camera angle for folded configuration plots
177  % I.InitialCameraHorizontalRotation: Horizontal rotation about the 3-axis
178  %       from negative 2-axis (degrees)
179  I.InitialCameraHorizontalRotation=135;
180
181  % I.InitialCameraVerticalElevation: Vertical elevation of viewpoint
182  %       (degrees)
183  I.InitialCameraVerticalElevation=30;
184
185  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
186  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
187  % Call main script for simulation
188  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
189  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
190  [PD,O] = Origami_Solver(I,KI);
```

B.2 Implementation of the Unfolding Polyhedra Method for Origami with Creased Folds (Chap. 3)

Here, we describe the MATLAB scripts that represent the implementation of the unfolding polyhedra method for origami with creased folds presented in Chap. 3. The full set of MATLAB scripts can be obtained from the Supplemental Material associated with that chapter.

To determine a net for a given goal mesh, the user opens and executes in MATLAB the corresponding `.m` file with name starting with `Input`. As an example, the code presented in this section is an input MATLAB script (named `Input_Unfolding_Cube.m` in the Supplemental Material) used to determine a net for a cube (Fig. 3.23b).

In the following code, we initialize the MATLAB environment in line 20 by clearing all the previously defined variables and functions from memory, closing all the open figures, and clearing the command window. In line 34, we provide the index of the face that corresponds to the reference face. In lines 37–45, we provide the coordinates of the nodes of the goal mesh. In lines 48–54, we provide connectivity information and indicate which nodes are associated with each face of the goal mesh. In line 61, we call the function `Polyhedra_Solver.m` that

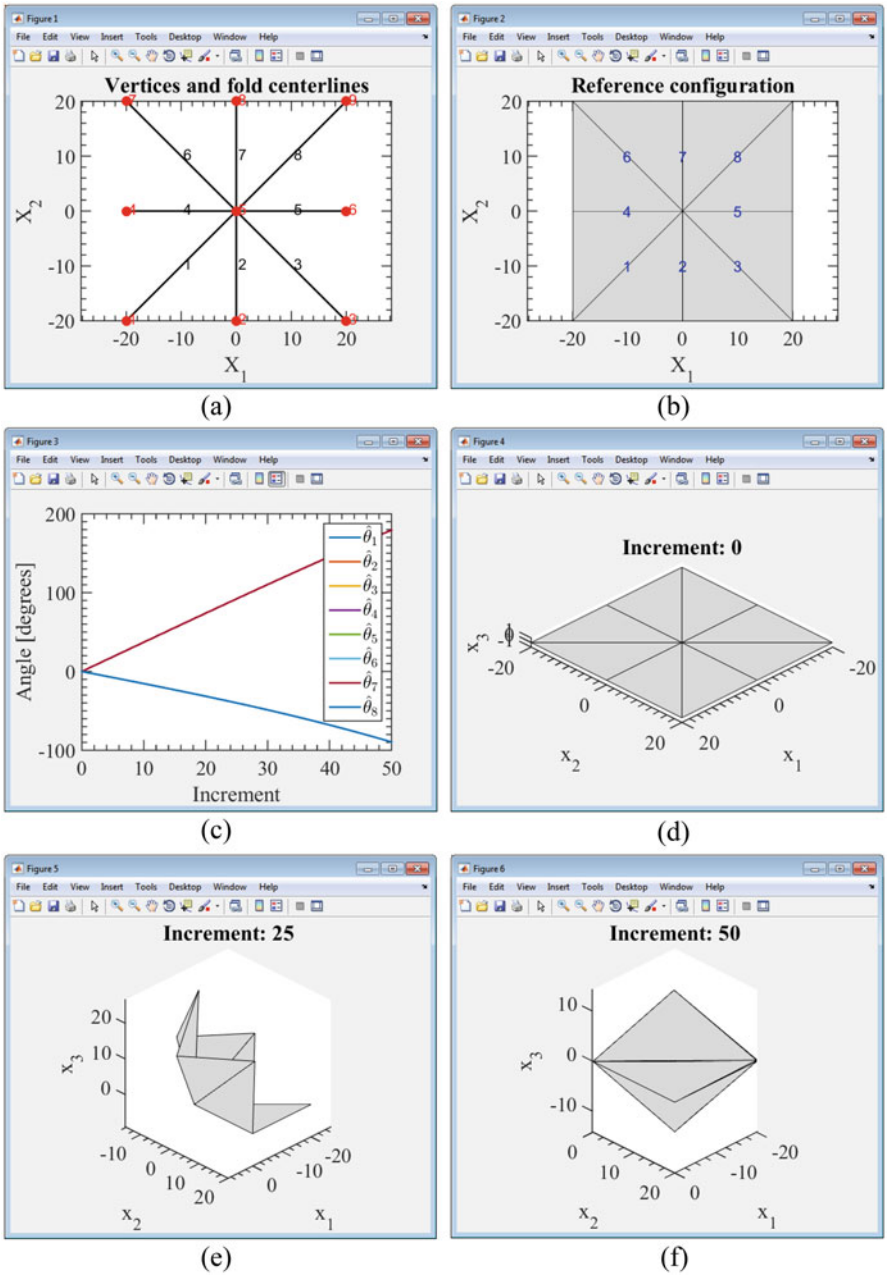


Fig. B.1 Screenshots of the output plots generated by the MATLAB scripts for the kinematic model of origami with creased folds presented in Chap. 2. (**a**) Location of the vertices and the fold lines. (**b**) Reference configuration. (**c**) Fold angles vs. increment number. (**d**)–(**f**) Plots of configurations for the requested increments

manages all the other functions required to determine a net for the given goal mesh. Such functions include:

1. `Goal_Mesh_Data.m`: Determines the geometric parameters associated with the goal mesh and the spanning tree
2. `Initial_Plots.m`: Generates plots associated with the goal mesh and the spanning tree (Fig. B.2a, b)
3. `Configuration_Mapping.m`: Maps each face of the goal mesh towards its corresponding position in the net using the formulation presented in Sect. 3.2.4
4. `Reference_Configuration_Plot.m`: Generates a plot of the net (Fig. B.2c)
5. `Simulation_Input.m`: Generates the data required to simulate the folding motion of the determined net back toward the three-dimensional goal mesh using the kinematic model developed in Chap. 2

The main numerical output from such files is the MATLAB variable `IN` (saved in line 64 of the following code), which is a structure containing the data required to simulate the folding motion of the determined net using the kinematic model presented in Chap. 2.

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % Supplemental Material
3  %
4  % ACTIVE ORIGAMI
5  % MODELING, DESIGN, AND APPLICATIONS
6  %
7  % Edwin A. Peraza Hernandez, Darren J. Hartl, Dimitris C. Lagoudas
8  %
9  % Springer
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11 %
12 % Code for Unfolding Polyhedra of Origami with Creased Folds
13 % Code authors: Cullen Nauck, Edwin A. Peraza Hernandez
14 % 04-2017
15 %
16 % Refer to Appendix B of the book for further information about the codes
17 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
18
19 % Initializing environment
20 clear all; close all; clc
21
22 % Main Structures
23 % I: Structure with general inputs
24 % MD: Structure with goal mesh data
25 % MO: Structure with mapping outputs
26 % IN: Structure with outputs corresponding to Kinematic Origami simulation
27 %       required inputs
28
29 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
30 % User inputs for goal mesh geometry
31 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
32
33 % I.ReferenceFace: Index of reference face for geometry
34 I.ReferenceFace = 1;
35
36 % I.NodeCoordinates: Coordinates of each node in goal mesh
37 I.NodeCoordinates = [...
38     1   1   1 % Coordinates of node 1
```

```
      1   1   -1 % Coordinates of node 2
      1   -1  1 % Coordinates of node 3
      1   -1  -1
      -1  1   1
      -1  1   -1
      -1  -1  1
      -1  -1  -1];

% I.FaceNodeIndices: Indices of nodes defining each face (ccw order)
I.FaceNodeIndices = [...
      2   1   3   4 % Node indices for face 1
      4   3   7   8 % Node indices for face 2
      8   7   5   6 % Node indices for face 3
      6   5   1   2
      5   7   3   1
      2   4   8   6];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Call main script for unfolding
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[I,MD, MO, IN] = Polyhedra_Solver(I);

% Save structure IN for use in kinematic origami simulation code
save('IN','IN')
```

After executing an input script, such as the one previously shown, and determining a net for a given goal mesh, the user can execute the MATLAB script `Kinematic_Simulation_Input.m`. The script loads the MATLAB variable `IN` saved by the input script in order to simulate the folding motion of the determined net. The user can specify the algorithmic parameters required to simulate the folding motion in the script `Kinematic_Simulation_Input.m` just as in the script described in Sect. B.1. After the execution of the script `Kinematic_Simulation_Input.m`, an arbitrary number of configurations of the determined net from the planar reference configuration to the configuration that matches the goal mesh are obtained as shown in Fig. B.3.

B.3 Implementation of the Tuck-Folding Method for Origami with Creased Folds (Chap. 4)

Here, we describe the MATLAB scripts representing the implementation of the tuck-folding method for origami with creased folds presented in Chap. 4. The full set of MATLAB scripts can be obtained from the Supplemental Material associated with that chapter.

To determine an origami sheet design for a given goal mesh using the tuck-folding method, the user opens and executes in MATLAB the corresponding `.m` file with name starting with `Input`. For example, the code presented in this section is an input MATLAB script used to determine an origami sheet design for the pyramid shown in Fig. 4.5 (the script is named

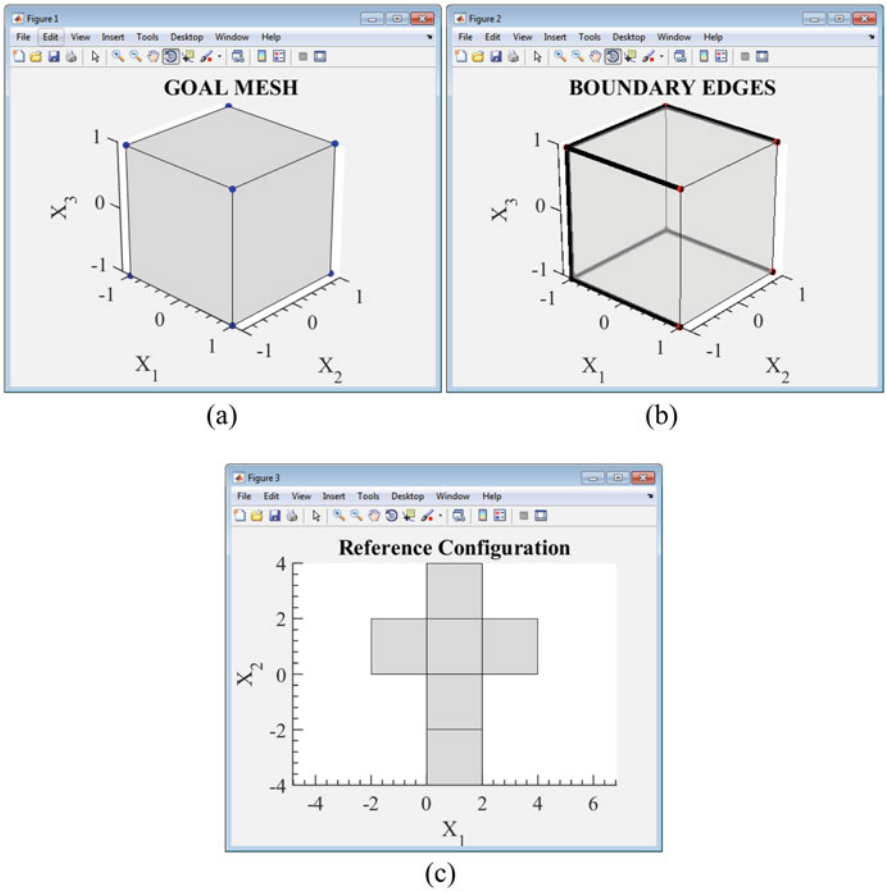


Fig. B.2 Screenshots of the output plots generated by the MATLAB scripts for the unfolding polyhedra method presented in Chap. 3

`Input_Origami_Single_Vertex_5_Edges.m` in the Supplemental Material).

In the following code, we initialize the MATLAB environment in line 20 by clearing all the previously defined variables and functions from memory, closing all the open figures, and clearing the command window. In lines 35–42, we provide the coordinates of the nodes of the goal mesh. We define connectivity and indicate which nodes are associated with each face of the goal mesh in lines 45–51. In line 53, we indicate the pairs of nodes defining the edges of the goal mesh that are assigned as boundary edges by the user (i.e., if these are not evident from the goal mesh, as in the case of closed shapes). In line 58, we provide the initial guess for the length variable of the edge modules ($\hat{W}_i$, refer to Fig. 4.6a). In line 62, we provide the initial guess for the angular variable of the edge modules ($\hat{\psi}_i$, refer

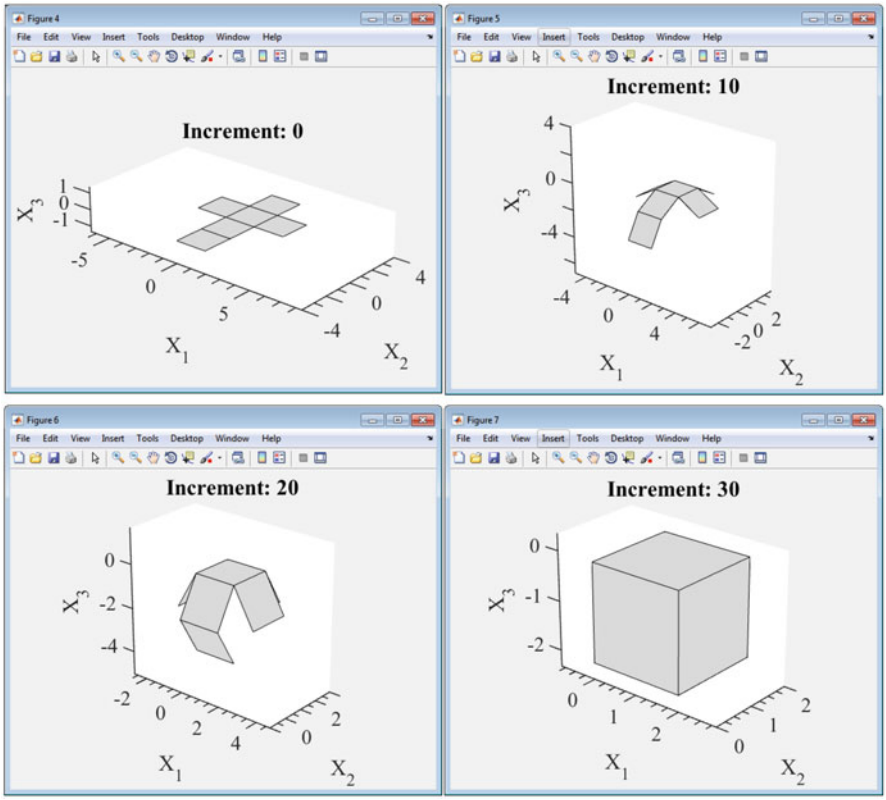


Fig. B.3 Screenshots of the output plots generated by the script `Kinematic_Simulation_Input.m` that simulates the folding motion of a net using the kinematic model presented in Chap. 2

to Fig. 4.6a). In lines 65–96, we provide the algorithmic parameters required in the numerical implementation of the tuck-folding method. In line 103, we call the function `Tuck_Fold_Solver.m` that manages all the other functions required to determine a sheet design for the given goal mesh. Such functions include:

1. `Goal_Mesh_Data.m`: Determines the geometric parameters associated with the goal mesh
2. `Initial_Plots.m`: Generates plots associated with the goal mesh (Fig. B.4a, b)
3. `Design_Parameter_Data.m`: Determines the values of the edge module variables using the procedure outlined in Sect. 4.2.5
4. `Angle_Equality.m`: Evaluates the constraint (4.6)
5. `Length_Equality.m`: Evaluates the constraint (4.7)
6. `Angle_Inequality.m`: Evaluates the constraint (4.11)
7. `Length_Inequality.m`: Evaluates the constraints (4.12) and (4.13)

8. `Configuration_Mapping.m`: Maps each face of the goal mesh towards its corresponding position in the sheet design
9. `Reference_Configuration_Plot.m`: Generates a plot of the origami sheet design (Fig. B.4c)
10. `Simulation_Input.m`: Generates the data required to simulate the folding motion of the determined sheet design back toward the goal mesh using the kinematic model presented in Chap. 2

The main output from such files is the MATLAB variable `IN` (saved in line 106 of the following code), which is a structure containing the data required to simulate the folding motion of the determined sheet design using the kinematic model developed in Chap. 2.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Supplemental Material
%
% ACTIVE ORIGAMI
% MODELING, DESIGN, AND APPLICATIONS
%
% Edwin A. Peraza Hernandez, Darren J. Hartl, Dimitris C. Lagoudas
%
% Springer
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Code for Tuck-Folding Method of Origami with Creased Folds
% Code authors: Edwin A. Peraza Hernandez, Cullen Nauck
% 05-2017
%
% Refer to Appendix B of the book for further information about the codes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Initializing environment
clear all; close all; clc

% Main Structures
% I: Structure with general inputs
% MD: Structure with goal mesh data
% O: Structure with outputs from solving for design parameters
% MO: Structure with mapping outputs
% IN: Structure with outputs corresponding to Kinematic Origami simulation
%       required inputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User inputs for goal mesh geometry
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.NodeCoordinates: Coordinates of each node in goal mesh
I.NodeCoordinates = [ ...
    10  10  9 % Coordinates of node 1
    17  10  2 % Coordinates of node 2
    0   -2  0 % Coordinates of node 3
    20  0   0
    0   20  0
    20  20  0
    ];

% I.FaceNodeIndices: Indices of nodes defining each face (ccw order)
I.FaceNodeIndices = [ ...
    1 3 4 % Node indices for face 1
    4 2 1 % Node indices for face 2
    1 2 6 % Node indices for face 3
```

```
     1 6 5
     3 1 5
    ];

% I.ExtraBoundaryEdges: Extra boundary edges defined by user
I.ExtraBoundaryEdges = [];

% I.WInitialGuess: Initial guess for distance between the two midpoints of
%       the outer folds of each edge module
I.WInitialGuess = 6;

% I.OuterFoldAngleGuess: Initial guess for angle between outer folds of
%       each edge module
I.OuterFoldAngleGuess = 0;

% I.ExtraAngleShrinkage: Extra trim angle shrinkage
I.ExtraAngleShrinkage = 0*pi/180;

% I.ExtraAngleBoundary: Extra trim angle boundary
I.ExtraAngleBoundary = 0*pi/180;

% I.AllowTrimMaterial: 1: allow trim to add more material, otherwise: ...
    disallow
I.AllowTrimMaterial = 0;

% I.ReferenceFace: Index of reference face
I.ReferenceFace = 3;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User algorithmic inputs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.MaxIterations: Maximum number of iterations
I.MaxIterations = 400;

% I.ToleranceResidual: Tolerance for norm of residual vector
I.ToleranceResidual = 1e-5;
% I.ToleranceChangeVariables: Tolerance in change of variables
I.ToleranceChangeVariables = 0;

% I.AngleBoundaryShift: Boundary shift to convert strict angle ...
    inequalities to relaxed inequalities
I.AngleBoundaryShift = 10*pi/180;
% I.LengthBoundaryShift: Boundary shift to convert strict length ...
    inequalities to relaxed inequalities
I.LengthBoundaryShift = 0*1;

% I.AngleFinDiffStep: Finite difference step for angles
I.AngleFinDiffStep = 1*pi/180;
% I.LengthFinDiffStep: Finite difference step for lengths
I.LengthFinDiffStep = 1e-2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Call main module for simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[I,MD,O, MO, IN] = Tuck_Fold_Solver(I);

% Save structure IN for use in kinematic origami simulation code
save('IN','IN')
```

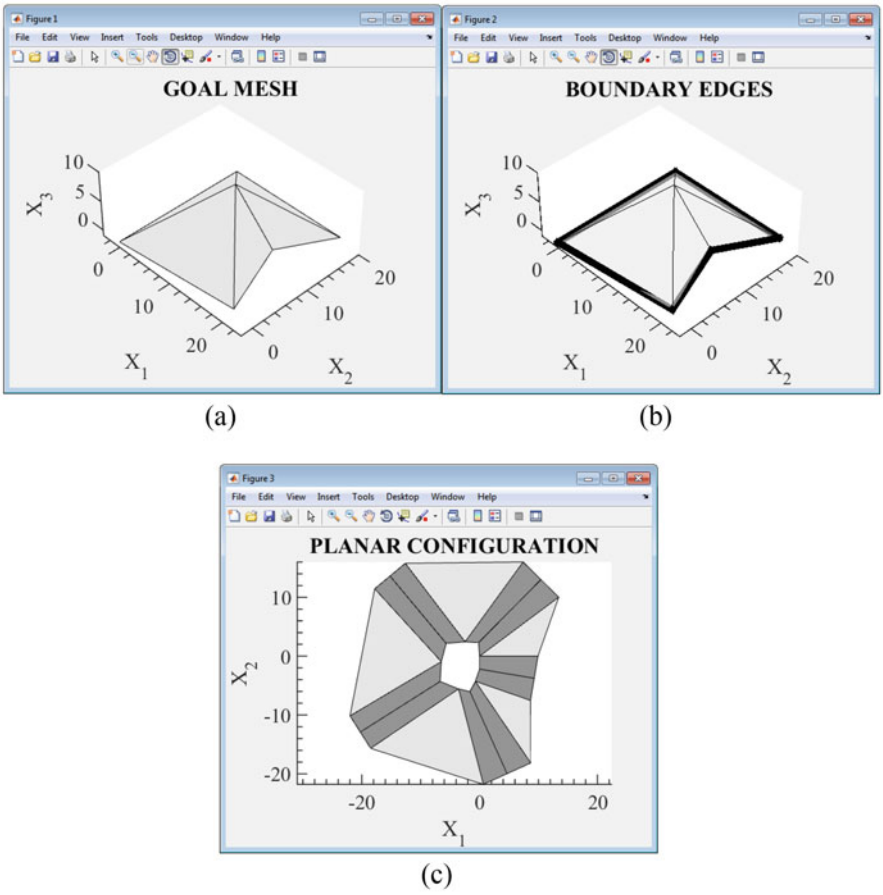


Fig. B.4 Screenshots of the output plots generated by the MATLAB scripts for the tuck-folding method presented in Chap. 4

After executing an input script, such as the one previously shown, and determining a sheet design for a given goal mesh, the user can execute the MATLAB script `Kinematic_Simulation_Input.m`. The script loads the MATLAB variable `IN` saved by the input script in order to simulate the folding motion of the determined sheet design. The user can specify the algorithmic parameters required to simulate the folding motion in the script `Kinematic_Simulation_Input.m` just as in the script described in Sect. B.1. After the execution of the script `Kinematic_Simulation_Input.m`, an arbitrary number of configurations of the sheet design from the planar reference configuration to the configuration that matches the goal mesh are obtained as shown in Fig. B.5.

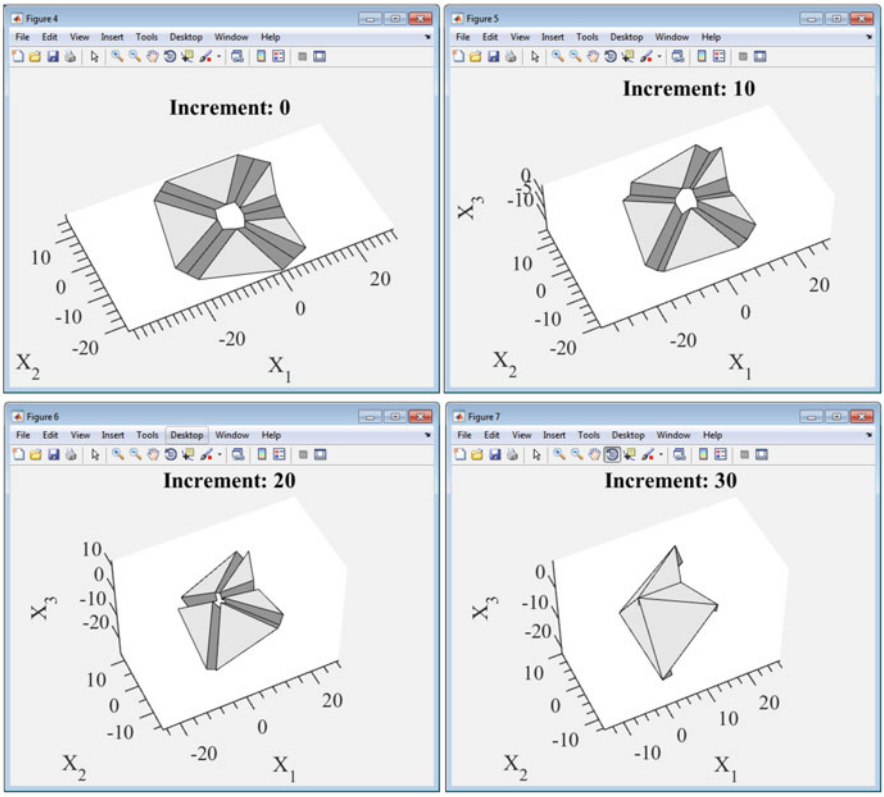


Fig. B.5 Screenshots of the output plots generated by the script `Kinematic_Simulation_Input.m` that simulates the folding motion of a determined origami sheet design using the kinematic model presented in Chap. 2

B.4 Implementation of the Kinematic Model for Origami with Smooth Folds (Chap. 5)

The MATLAB scripts associated with the kinematic model for origami with smooth folds as presented in Chap. 5 are described here. To simulate the folding motion of an origami sheet using such a model, the user opens and executes in MATLAB the corresponding `.m` file with name starting with `Input`. As an example, consider the code presented in this section, which is an input MATLAB script used to simulate the folding motion of the origami sheet shown in Fig. 5.28 (the script is named `Input_Rectangle1Intersection.m` in the Supplemental Material).

In the following code, we initialize the MATLAB environment in line 20 by clearing all the previously defined variables and functions from memory, closing all the open figures, and clearing the command window. In line 36, we select the increments for which the configurations are plotted for visualization. The

parameters required to define the fold pattern are provided in lines 43–114, with the fold widths defined in lines 75 and 76. In lines 118–165 and 191–208, we provide the algorithmic parameters required to simulate the folding motion. In lines 169–181, we provide the increments for the fold angles. The order of geometric continuity of the smooth folds is provided in line 185 (1: first-order; 2: second-order). In line 215, we call the function `Origami_Solver.m` that manages all the other functions required to simulate the motion of an origami sheet. Such functions include:

1. `Fold_Pattern_Data.m`: Determines the data associated with the fold pattern
2. `Reference_Plots.m`: Generates plots associated with the reference configuration and the fold pattern (Fig. B.6a, b)
3. `Kinematic_Simulation.m`: Executes the procedure for kinematic simulation presented in Sect. 5.7
4. `Fold_Width_Straight.m`: Determines the variable $\hat{w}_i$
5. `Beta_1.m`: Determines the variable $\hat{\beta}_1^i$
6. `Beta_2.m`: Determines the variable $\hat{\beta}_2^i$
7. `Rotation_Constraint.m`: Evaluates the kinematic constraint (5.66)
8. `Translation_Constraint.m`: Evaluates the kinematic constraint (5.67)
9. `Face_Mapping.m`: Determines the mapping between reference and current configurations for the faces using the formulation presented in Sect. 5.6
10. `Fold_Angle_Plots.m`: Generates plots of fold angle values vs. increment number (Fig. B.6c)
11. `Folded_Configurations.m`: Generates plots of configurations for the requested increments (Fig. B.6d–f)

The main numeric output variables include:

1. `O.FoldAngles`: Array of fold angle values for each increment
2. `O.exitFlag`: Indicates the reason for stopping the iterative correction procedure at each increment (1: Tolerance in residual achieved; 2: Tolerance in change in fold angles achieved; 3: Maximum number of iterations reached)
3. `O.ResidualNorm`: Magnitude of the residual vector for each increment

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % Supplemental Material
3  %
4  % ACTIVE ORIGAMI
5  % MODELING, DESIGN, AND APPLICATIONS
6  %
7  % Edwin A. Peraza Hernandez, Darren J. Hartl, Dimitris C. Lagoudas
8  %
9  % Springer
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11 %
12 % Code for Kinematic Simulation of Origami with Smooth Folds
13 % Code authors: Cullen Nauck, Edwin A. Peraza Hernandez
14 % 03-2017
15 %
16 % Refer to Appendix B of the book for further information about the codes
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Initializing environment
clear all; close all; clc

% Main structures
% I: Structure with general inputs
% PD: Structure with geometric pattern data
% O: Structure with general outputs
% KI: Structure with kinematic simulation inputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User inputs for fold pattern
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.PlotFoldAngles: Fold angles to plot vs. increment number
I.PlotFoldAngles = [1; 2; 3];

% I.PlotConfigurations: Increment number of configurations to be plotted
I.PlotConfigurations = [0; 10; 20];

% I.ReferencePoints: Reference point coordinates (in reference
%       configuration). These reference points are used as vertices or
%       additional corners for the rigid faces
% NOTE: Reference points will be labeled (1,2,3,etc.) corresponding to
%       their row index in this array
I.ReferencePoints = [
    -30 -20
    0   -20
    30  -20
    -30 0
    0   0;
    30  0
    -30 20
    0   20
    30  20];

% FoldVectorPointsI: Indices of reference points that correspond to
%       start/end points of each fold vector
% NOTE: Vector will run in a positive direction from the first point
%       (start) to the second point (end)
% NOTE: Vectors will be labeled (1,2,3,etc.) corresponding to their row
%       index in this array
I.FoldVectorPoints = [
    5   6
    5   9
    5   8
    5   7
    5   4;
    5   1
    5   2
    5   3
    ];

% Num_Folds: Number of fold inputs
Num_Folds=size(I.FoldVectorPoints,1);

% I.FoldWidth: Width of each fold in simulation
I.FoldWidth = [3; 3; 3; 3; 3; ...
               3; 3; 3];

% I.r_parameters: Length parameters associated with each fold
%   (perpendicular distance between vertex and fold boundary, 1st column
%   r1, 2nd column r2)
I.r_parameters = ...
    [6   4
     6   6
```

```
      6   2.4
      6   6
      6   4
      6   6
      6   2.4
      6   6];

% I.IntersectionFoldIndices: Matrix with fold indices associated with each
%       interior fold intersection (in counterclockwise order)
% NOTE: Positive sign if fold vector points away from intersection;
%       Negative sign if fold vector points towards intersection
I.IntersectionFoldIndices = [
    1   2   3   4   5   6   7   8];

% I.FaceBoundary: Indices of folds and reference points defining the
%       boundary of each face (ccw order)
% NOTE: Signed depending on if fold vector points CCW (positive) around
%       face or Clockwise (negative) around face
% NOTE: If individual reference points are used to define corners of a face
%       at the boundary, the points should be referred as # of folds +
%       reference point index #)
I.FaceBoundary = [
    1   -2
    2   -3
    3   -4
    4   -5
    5   -6
    6   -7
    7   -8
    8   -1
    ];

% I.FixedFace: Index of face selected to be fixed throughout simulation
%              (one required)
I.FixedFace = 1;

% I.FoldAngleUpperBound: Highest allowable fold angle for each fold
I.FoldAngleUpperBound = [
    pi;   pi;   pi;   pi;   pi; ...
    pi;   pi;   pi ];

% I.FoldAngleLowerBound: Lowest allowable fold angle for each fold
I.FoldAngleLowerBound = [
    -pi;  -pi;  -pi;  -pi;  -pi; ...
    -pi;  -pi;  -pi];

% I.FoldAngleInitialValue: Initial value for each fold
I.FoldAngleInitialValue = [
    0;    0;    0;    0;    0; ...
    0;    0;    0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User inputs for kinematic simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% KI.WeightRotationConstraint: Weight for residuals from rotation
%       constraints
KI.WeightRotationConstraint = 1;

% KI.WeightTranslationConstraint: Weight for residuals from translation
%       constraints
KI.WeightTranslationConstraint = 1e-3;

% KI.WeightFoldAngleBounds: Weight for residuals from fold angle bound
%       constraints
KI.WeightFoldAngleBounds = 1e-5;

```

```
% KI.ToleranceResidual: Tolerance for norm of residual vector
KI.ToleranceResidual = 1e-11;

% KI.ToleranceFoldAngle: Tolerance for correction of fold angles
KI.ToleranceFoldAngle = 1e-8;

% KI.MaxIterations: Maximum number of iterations allowed for each increment
KI.MaxIterations = 15;

% KI.MaxFoldAngleCorrection: Maximum correction allowed for individual fold
%       angles in each iteration
KI.MaxFoldAngleCorrection = 5*pi/180;

% Num_Increments: Number of increments
NumIncrements = 20;

% KI.FoldAngleChangePerIncrement: Initialize matrix that contains each fold
%       angle change for every increment
KI.FoldAngleChangePerIncrement = zeros( size(I.FoldVectorPoints,1), ...
     NumIncrements);

% ReferenceFoldAngle: Reference angle for filling fold angle change per
%       increment matrix
ReferenceFoldAngle = 150*pi/180;

% Fold angle change per increment matrix components
for i = 1:NumIncrements
   KI.FoldAngleChangePerIncrement(1:end,i) = ...
        ReferenceFoldAngle/NumIncrements*[
         1;  -1;   1;   -1;  1;
        -1;   1;  -1];
end

% KI.Continuity: Input to determine order of continuity. Input 1 for G^1
%       continuity and input 2 for G^2 continuity
KI.Continuity = 2;

% KI.IncrementCorrection: Input to determine one of two methods for
%       calculating initial fold angle increment. One uses the previous
%       Jacobian (corrected; Select 1) and the other one does not (not
%       corrected; Select 2)
KI.IncrementCorrection = 1;

% KI.DisplayIncrementInfo: Input to allow for the display of iteration
%       information for each increment (1 to display, 0 to suppress)
KI.DisplayIncrementInfo = 1;

% KI.FinDiffStep: Finite difference step in fold angles for calculation of
%       derivatives
KI.FinDiffStep = 2*pi/180;

%  User input for initial camera angle for folded configuration plots
% I.InitialCameraHorizontalRotation: Horizontal rotation about the 3-axis
%       from negative 2-axis (degrees)
I.InitialCameraHorizontalRotation=135;

% I.InitialCameraVerticalElevation: Vertical elevation of viewpoint
%       (degrees)
I.InitialCameraVerticalElevation=30;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Call main module for simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[PD,O] = Origami_Solver(I,KI);
```

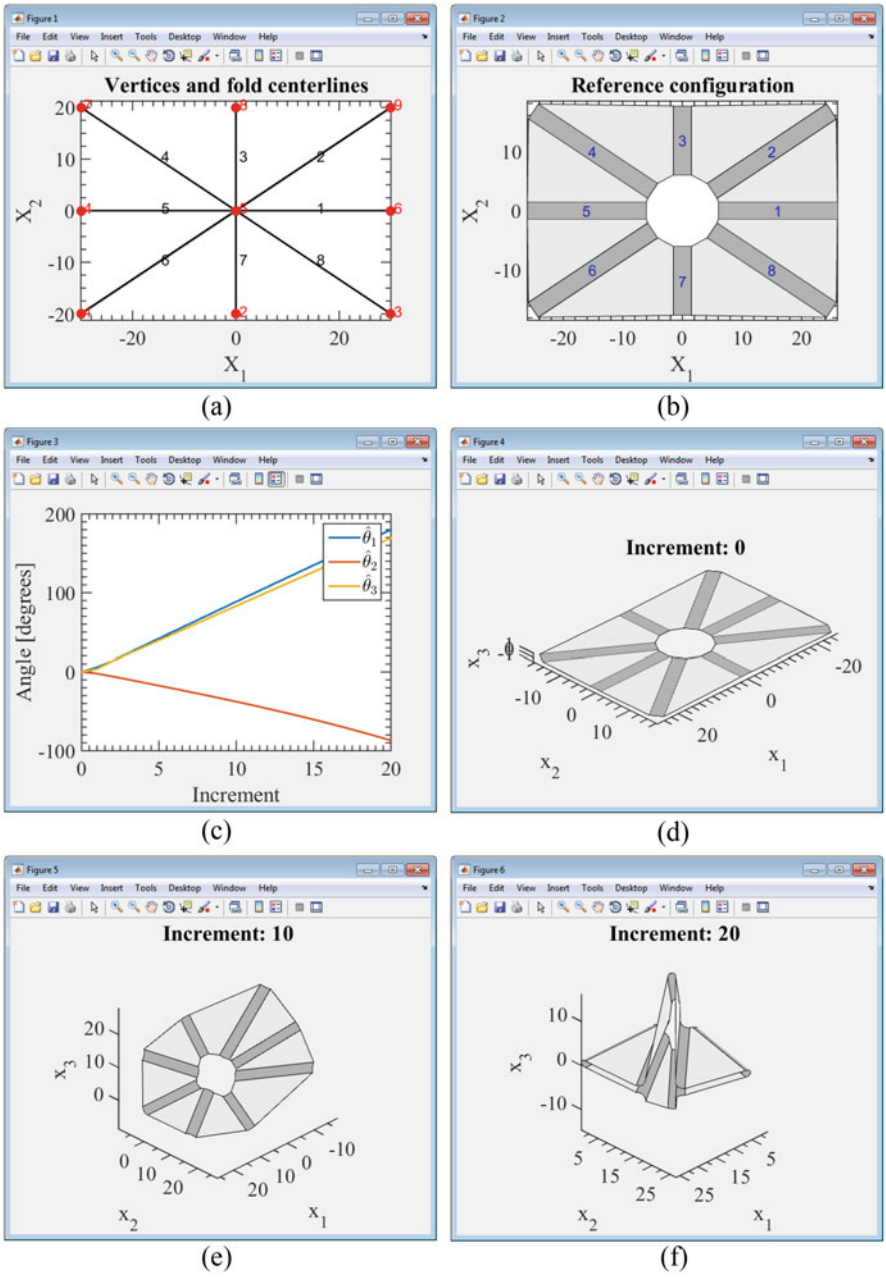


Fig. B.6 Screenshots of the output plots generated by the MATLAB scripts for the kinematic model of origami with smooth folds presented in Chap. 5. (**a**) Location of the vertices and the fold centerlines. (**b**) Reference configuration. (**c**) Fold angles vs. increment number. (**d**)–(**f**) Plots of configurations for the requested increments

B.5 Implementation of the Unfolding Polyhedra Method for Origami with Smooth Folds (Chap. 6)

The MATLAB scripts that represent the implementation of the unfolding polyhedra method for origami with smooth folds presented in Chap. 6 are described here. The full set of MATLAB scripts can be obtained from the Supplemental Material associated with that chapter.

To determine a net having smooth folds for a given goal mesh, the user opens and executes in MATLAB the corresponding `.m` file with name starting with `Input`. As an example, the code presented in this section is an input MATLAB script (named `Input_Unfolding_Octahedron.m` in the Supplemental Material) used to determine a net for an octahedron (Fig. 6.7c).

In the following code, we initialize the MATLAB environment in line 20 by clearing all the previously defined variables and functions from memory, closing all the open figures, and clearing the command window. We provide the fold widths in line 34. The order of geometric continuity of the smooth folds is provided in line 37 (1: first-order; 2: second-order). In line 40, we provide the index of the face that corresponds to the reference face. In lines 43–49, we provide the coordinates of the nodes of the goal mesh. In lines 52–60, we provide connectivity information and indicate which nodes are associated with each face of the goal mesh. In line 67, we call the function `Polyhedra_Solver.m` that manages all the other functions required to determine a net for the given goal mesh. Such functions include:

1. `Goal_Mesh_Data.m`: Determines the geometric parameters associated with the goal mesh and the spanning tree
2. `Fold_Width_Straight.m`: Determines the variable $\hat{w}_i$
3. `Initial_Plots.m`: Generates plots associated with the goal mesh and the spanning tree (Fig. B.7a–c)
4. `Configuration_Mapping.m`: Maps each face of the goal mesh towards its corresponding position in the net
5. `Reference_Configuration_Plot.m`: Generates a plot of the net (Fig. B.7d)
6. `Simulation_Input.m`: Generates the data required to simulate the folding motion of the determined net back toward the three-dimensional goal mesh using the kinematic model developed in Chap. 5

The main numerical output from such files is the MATLAB variable `IN` (saved in line 70 of the following code), which is a structure containing the data required to simulate the folding motion of the determined net using the kinematic model presented in Chap. 5.

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % Supplemental Material
3  %
4  % ACTIVE ORIGAMI
5  % MODELING, DESIGN, AND APPLICATIONS
```

```
%
% Edwin A. Peraza Hernandez, Darren J. Hartl, Dimitris C. Lagoudas
%
% Springer
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Code for Unfolding Polyhedra of Origami with Smooth Folds
% Code authors: Edwin A. Peraza Hernandez, Trent White
% 06-2017
%
% Refer to Appendix B of the book for further information about the codes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Initializing environment
clear all; close all; clc

% Main Structures
% I: Structure with general inputs
% MD: Structure with goal mesh data
% MO: Structure with mapping outputs
% IN: Structure with outputs corresponding to Kinematic Origami simulation
%       required inputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User inputs for goal mesh geometry
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.FoldWidthInput: Fold width
I.FoldWidthInput = 0.2;

% I.Continuity: Order of geometric continuity of the folds
I.Continuity = 2;

% I.ReferenceFace: Index of reference face for geometry
I.ReferenceFace = 1;

% I.NodeCoordinates: Coordinates of each node in goal mesh
I.NodeCoordinates = [...
    1   0   0 % Coordinates of node 1
    -1  0   0 % Coordinates of node 2
    0   1   0 % Coordinates of node 3
    0   -1  0
    0   0   1
    0   0   -1];

% I.FaceNodeIndices: Indices of nodes defining each face (ccw order)
I.FaceNodeIndices = [...
    1   3   5 % Node indices for face 1
    1   5   4 % Node indices for face 2
    2   4   5 % Node indices for face 3
    2   5   3
    4   2   6
    6   2   3
    6   3   1
    1   4   6];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Call main module for simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[I,MD,MO,IN] = Polyhedra_Solver(I);

% Save structure IN for use in kinematic origami simulation code
save('IN','IN')
```

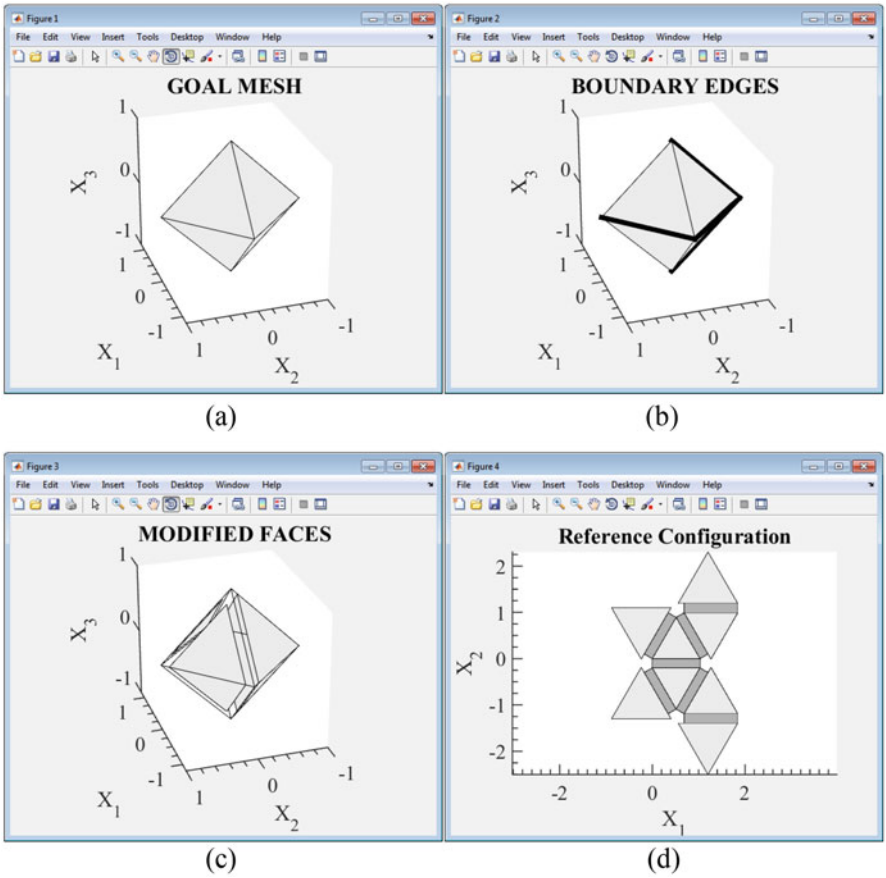


Fig. B.7 Screenshots of the output plots generated by the MATLAB scripts for the unfolding polyhedra method for origami with smooth folds presented in Chap. 6

After executing an input script, such as the one previously shown, and determining a net with smooth folds for a given goal mesh, the user can execute the MATLAB script `Kinematic_Simulation_Input.m`. The script loads the MATLAB variable `IN` saved by the input script in order to simulate the folding motion of the determined net. The user can specify the algorithmic parameters required to simulate the folding motion in the script `Kinematic_Simulation_Input.m` just as in the script described in Sect. B.4. After the execution of the script `Kinematic_Simulation_Input.m`, an arbitrary number of configurations of the determined net from the planar reference configuration to the configuration that approximates the goal mesh are obtained as shown in Fig. B.8.

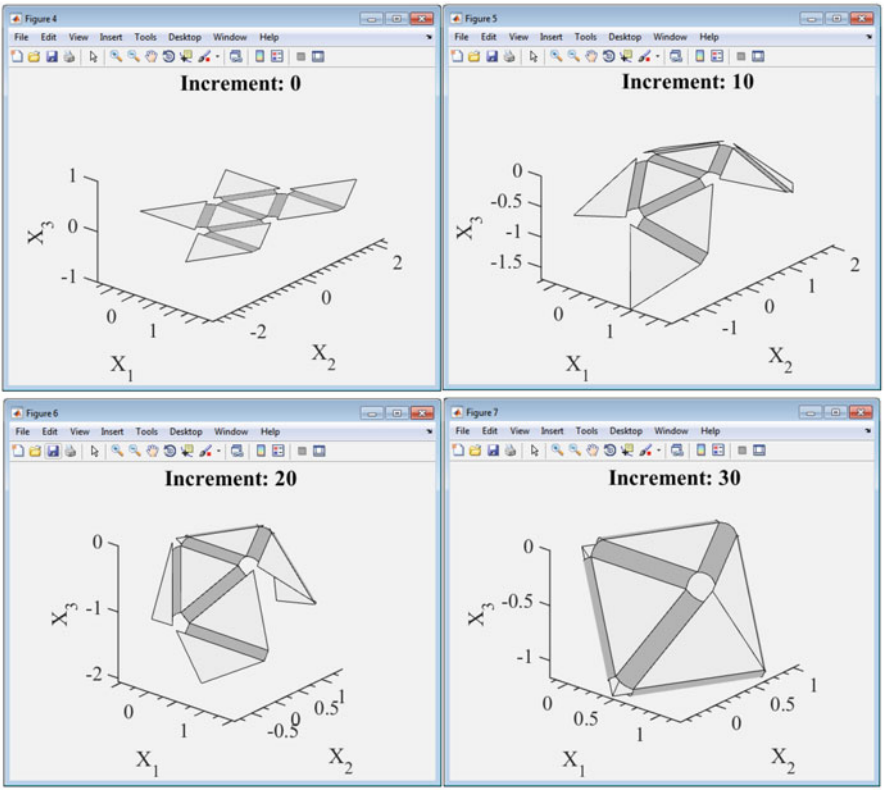


Fig. B.8 Screenshots of the output plots generated by the script `Kinematic_Simulation_Input.m` that simulates the folding motion of a net having smooth folds using the kinematic model presented in Chap. 5

B.6 Implementation of the Tuck-Folding Method for Origami with Smooth Folds (Chap. 7)

Here, we describe the MATLAB scripts representing the implementation of the tuck-folding method for origami with smooth folds presented in Chap. 7. The full set of MATLAB scripts can be obtained from the Supplemental Material associated with that chapter.

To determine an origami sheet design having smooth folds for a given goal mesh using the tuck-folding method, the user opens and executes in MATLAB the corresponding `.m` file with name starting with `Input`. For example, the code presented in this section is an input MATLAB script used to determine an origami sheet design for the pyramid shown in Fig. 7.3 (the script is named `Input_Origami_Single_Vertex_5_Edges.m` in the Supplemental Material).

In the following code, we initialize the MATLAB environment in line 20 by clearing all the previously defined variables and functions from memory, closing all the open figures, and clearing the command window. In lines 34–41, we provide the coordinates of the nodes of the goal mesh. We define connectivity and indicate which nodes are associated with each face of the goal mesh in lines 44–49. In line 52, we indicate the pairs of nodes defining the edges of the goal mesh that are assigned as boundary edges by the user (i.e., if these are not evident from the goal mesh, as in the case of closed shapes). In line 56, we provide the initial guess for the length variable of the edge modules ($\hat{W}_i$, refer to Fig. 7.5a). In line 60, we provide the initial guess for the angular variable of the edge modules ($\hat{\psi}_i$, refer to Fig. 7.5a). In lines 63 and 64, we provide the fold widths of the smooth folds in the edge modules. In lines 67–103, we provide the algorithmic parameters required in the numerical implementation of the tuck-folding method. In line 110, we call the function `Tuck_Fold_Solver.m` that manages all the other functions required to determine a sheet design for the given goal mesh. Such functions include:

1. `Goal_Mesh_Data.m`: Determines the geometric parameters associated with the goal mesh
2. `Fold_Width_Straight.m`: Determines the variable $\hat{w}_i$
3. `Initial_Plots.m`: Generates plots associated with the goal mesh (Fig. B.9a–c)
4. `Design_Parameter_Data.m`: Determines the values of the edge module variables using the procedure outlined in Sect. 4.2.5
5. `Angle_Equality.m`: Evaluates the constraint (7.9)
6. `Length_Equality.m`: Evaluates the constraint (7.10)
7. `Angle_Inequality.m`: Evaluates the constraints (7.11)
8. `Length_Inequality.m`: Evaluates the constraints (7.12) and (7.13)
9. `Configuration_Mapping.m`: Maps each face of the goal mesh towards its corresponding position in the sheet design
10. `Reference_Configuration_Plot.m`: Generates a plot of the origami sheet design (Fig. B.9d)
11. `Simulation_Input.m`: Generates the data required to simulate the folding motion of the determined sheet design back toward the goal mesh using the kinematic model presented in Chap. 5

The main output from such files is the MATLAB variable `IN` (saved in line 113 of the following code), which is a structure containing the data required to simulate the folding motion of the determined sheet design using the kinematic model developed in Chap. 5.

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % Supplemental Material
3  %
4  % ACTIVE ORIGAMI
5  % MODELING, DESIGN, AND APPLICATIONS
6  %
7  % Edwin A. Peraza Hernandez, Darren J. Hartl, Dimitris C. Lagoudas
8  %
```

```
% Springer
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Code for Tuck-Folding Method of Origami with Smooth Folds
% Code authors: Edwin A. Peraza Hernandez, Trent White
% 06-2017
%
% Refer to Appendix B of the book for further information about the codes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Initializing environment
clear all; close all; clc

% Main Structures
% I: Structure with general inputs
% O: Structure with outputs from solving for design parameters
% MO: Structure with mapping outputs
% IN: Structure with outputs corresponding to Kinematic Origami simulation
%       required inputs

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User inputs for goal mesh geometry
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.NodeCoordinates: Coordinates of each node in goal mesh
I.NodeCoordinates = [...
    10 10 9 % Coordinates of node 1
    17 10 2 % Coordinates of node 2
    0  -2 0 % Coordinates of node 3
    20 0  0
    0  20 0
    20 20 0
    ];

% I.FaceNodeIndices: Indices of nodes defining each face (ccw order)
 I.FaceNodeIndices = [ 1 3 4 % Node indices for face 1
    4 2 1 % Node indices for face 2
    1 2 6 % Node indices for face 3
    1 6 5
    3 1 5
   ];

% I.BoundaryEdges: Extra boundary edges defined by user
I.BoundaryEdges = [];

% I.WInitialGuess: Initial guess for distance between the two midpoints of
%       the outer folds of each edge module
I.WInitialGuess = 6;

% I.OuterFoldAngleGuess: Initial guess for angle between outer folds of
%       each edge module
I.OuterFoldAngleGuess = 0;

% Fold widths
I.InnerFoldWidthInput = 1.0;
I.ExternalFoldWidthInput = 1.0;

% I.ExtraAngleShrinkage: Extra trim angle shrinkage
I.ExtraAngleShrinkage = 0*pi/180;

% I.ExtraAngleBoundary: Extra trim angle boundary
I.ExtraAngleBoundary = 0*pi/180;

% I.Continuity: Order of geometric continuity for the folds
I.Continuity = 2;

```

```
% I.AllowTrimMaterial: 1: allow trim to add more material, otherwise: ...
    disallow
I.AllowTrimMaterial = 0;

% I.ReferenceFace: Index of reference face
I.ReferenceFace = 3;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User algorithmic inputs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% I.MaxIterations: Maximum number of iterations
I.MaxIterations = 400;

% I.ToleranceResidual: Tolerance for norm of residual vector
I.ToleranceResidual = 1e-5;

% I.ToleranceChangeVariables: Tolerance in change of variables
I.ToleranceChangeVariables = 0;

% I.AngleBoundaryShift: Boundary shift to convert strict angle ...
    inequalities to relaxed inequalities
I.AngleBoundaryShift = 10*pi/180;

% I.LengthBoundaryShift: Boundary shift to convert strict length ...
    inequalities to relaxed inequalities
I.LengthBoundaryShift = I.InnerFoldWidthInput*1;

% I.AngleFinDiffStep: Finite difference step for angles
I.AngleFinDiffStep = 1*pi/180;
% I.LengthFinDiffStep: Finite difference step for lengths
I.LengthFinDiffStep  = 1e-2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Call main module for simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[I, O, MO, IN] = Tuck_Fold_Solver(I);

% Save structure IN for use in kinematic origami simulation code
save('IN','IN')
```

After executing an input script, such as the one previously shown, and determining a sheet design with smooth folds for a given goal mesh, the user can execute the MATLAB script `Kinematic_Simulation_Input.m`. The script loads the MATLAB variable `IN` saved by the input script in order to simulate the folding motion of the sheet design. The user can specify the algorithmic parameters required to simulate the folding motion in the script `Kinematic_Simulation_Input.m` just as in the script described in Sect. B.4. After the execution of the script `Kinematic_Simulation_Input.m`, an arbitrary number of configurations of the sheet design from the planar reference configuration to the configuration that approximates the goal mesh are obtained as shown in Fig. B.10.

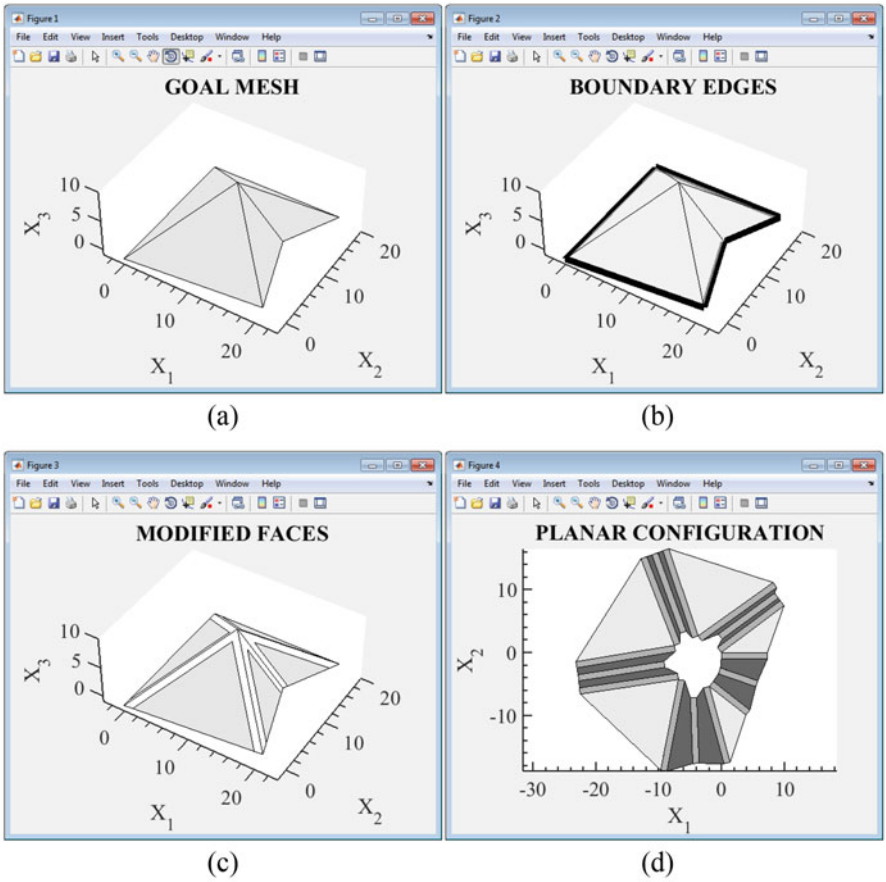


Fig. B.9 Screenshots of the output plots generated by the MATLAB scripts for the tuck-folding method for origami with smooth folds presented in Chap. 7

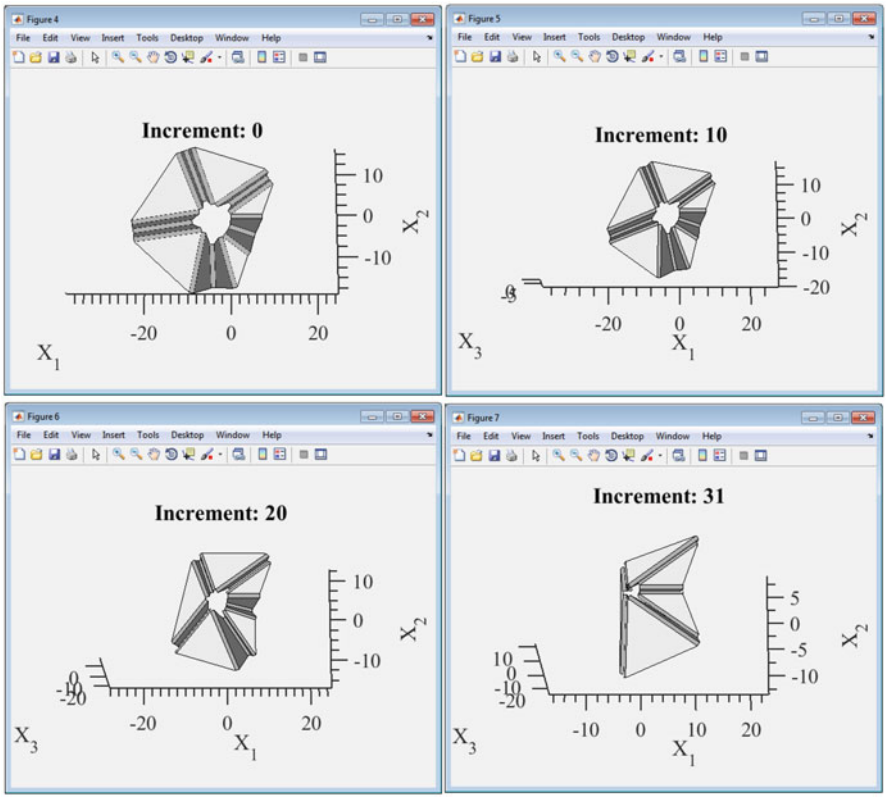


Fig. B.10 Screenshots of the output plots generated by the script `Kinematic_Simulation_Input.m` that simulates the folding motion of a determined origami sheet design using the kinematic model presented in Chap. 5

Appendix C
Constitutive Models

In this appendix, we summarize the constitutive models for the materials considered in the examples of Chap. 8. As noted in Sect. 8.3.2, the constitutive response of a material is manifested in the equation of conservation of linear momentum (8.14) in the formulation for the second-order *Cauchy stress tensor* $\boldsymbol{\sigma}$, which is a function of the material deformation and other physical fields. Here we provide such a formulation for the stress tensor $\boldsymbol{\sigma}$ considering: linear elastic materials, thermoelastic materials, piezoelectric materials, and phase transforming materials. It is assumed that the reader is familiar with introductory continuum mechanics and tensor algebra [5–8].

C.1 Linear Elastic Materials

For a *linear elastic material*, the stress tensor $\boldsymbol{\sigma}$ is related to the second-order *linearized strain tensor* $\boldsymbol{\varepsilon}$ via Hooke's law [9]:

$$\boldsymbol{\sigma} = \boldsymbol{C}\boldsymbol{\varepsilon}, \tag{C.1}$$

where $\boldsymbol{C}$ is the constant fourth-order *stiffness tensor*.

For the plane stress assumptions of classical plate theory, which is employed in Chap. 8, the stress-strain relation (C.1) is simplified as follows [10]:

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & 1-\nu \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \end{bmatrix}, \tag{C.2}$$

where E and ν are, respectively, the Young's modulus and Poisson's ratio of the linear elastic material, which is assumed *isotropic*.

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2

C.2 Thermoelastic Materials

For a *thermoelastic material*, the stress tensor $\boldsymbol{\sigma}$ is related to the strain tensor $\boldsymbol{\varepsilon}$ and the *absolute temperature* T as follows:

$$\boldsymbol{\sigma} = \boldsymbol{C}\left(\boldsymbol{\varepsilon} - \boldsymbol{\alpha}(T - T_0)\right), \tag{C.3}$$

where $\boldsymbol{\alpha}$ is the constant second-order *tensor of thermoelastic coefficients* and T_0 is the reference absolute temperature. For plane stress, and assuming that the material is isotropic, (C.3) is simplified as follows:

$$\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{12}\end{bmatrix} = \frac{E}{1-\nu^2}\begin{bmatrix}1 & \nu & 0\\ \nu & 1 & 0\\ 0 & 0 & 1-\nu\end{bmatrix}\left(\begin{bmatrix}\varepsilon_{11}\\ \varepsilon_{22}\\ \varepsilon_{12}\end{bmatrix} - \begin{bmatrix}\alpha\\ \alpha\\ 0\end{bmatrix}(T - T_0)\right), \tag{C.4}$$

where α is the *coefficient of thermal expansion*.

C.3 Piezoelectric Materials

Piezoelectric materials exhibit coupling between mechanical and electric fields [11, 12]. This coupling allows piezoelectric materials to function as sensors (by generating electric signals under the application of stress) and actuators (by generating *actuation strains* under the application of an electric field due to the piezoelectric effect). Here, we are particularly interested in the use of piezoelectric materials as actuators for active origami structures. For these materials, the stress tensor $\boldsymbol{\sigma}$ is related to the strain tensor $\boldsymbol{\varepsilon}$ and the *electric field vector* $\boldsymbol{E}$ as follows [13]:

$$\boldsymbol{\sigma} = \boldsymbol{C}\left(\boldsymbol{\varepsilon} - \boldsymbol{d}^P\boldsymbol{E}\right), \tag{C.5}$$

where $\boldsymbol{d}^P$ is the third-order tensor of piezoelectric coefficients.

Assuming an orthotropic piezoelectric material poled in the 3-direction, (C.5) is written in plane stress form as follows:

$$\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{12}\end{bmatrix} = \begin{bmatrix}E_1/(1-\nu_{12}\nu_{21}) & \nu_{21}E_1/(1-\nu_{12}\nu_{21}) & 0\\ \nu_{12}E_2/(1-\nu_{12}\nu_{21}) & E_2/(1-\nu_{12}\nu_{21}) & 0\\ 0 & 0 & 2G_{12}\end{bmatrix} \times \left(\begin{bmatrix}\varepsilon_{11}\\ \varepsilon_{22}\\ \varepsilon_{12}\end{bmatrix} - \begin{bmatrix}0 & 0 & d_{113}\\ 0 & 0 & d_{223}\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\hat{E}_1\\ \hat{E}_2\\ \hat{E}_3\end{bmatrix}\right), \tag{C.6}$$

where E_1, E_2, ν_{12}, ν_{21}, and G_{12} are material parameters that define the orthotropic stiffness matrix in plane stress, $\hat{E}_1$, $\hat{E}_2$, and $\hat{E}_3$ are the components of the electric field vector $\boldsymbol{E}$, and d_{113} and d_{223} are piezoelectric coefficients. The parameters E_1, E_2, ν_{12}, and ν_{21} satisfy the following relation:

$$\frac{\nu_{12}}{E_1} = \frac{\nu_{21}}{E_2}. \tag{C.7}$$

C.4 Phase Transforming Materials

In *phase transforming materials*, actuation strains are generated by changes in the material microstructure. These changes in the microstructure are dependent on different physical fields such as stress, temperature, electric field, magnetic field, etc. depending on the particular phase transforming material. Examples of phase transforming materials include *shape memory alloys* (SMAs) and *shape memory polymers* (SMPs). The phase transformation process in both SMAs and SMPs is dependent on stress and temperature. The model presented in this section is intended for SMAs; however, we note that analogous mathematical formulations can be used to model other phase transforming materials. *Shape memory alloys* are active materials that undergo solid-to-solid martensitic phase transformations induced by temperature and/or stress stimuli during which they can generate or recover seemingly permanent strains [14]. Such martensitic phase transformations occur between a *martensite* phase (stable at low temperatures) and an *austenite* phase (stable at high temperatures). *Actuation strains* in an SMA are obtained in the form of *transformation strains*, which are generated when the SMA transforms from austenite to martensite (*forward transformation*) and recovered when the SMA transforms from martensite to austenite (*reverse transformation*). The magnitude of such transformation strains typically ranges between 1% and 5% [14].

Shape memory alloys have various applications in different fields such as aerospace [15–17], robotics [18, 19], and others [20]. Shape memory alloys exhibit the highest actuation energy density of all conventionally used active materials [14]. Therefore, SMAs can provide a significant amount of strain under large stresses, a characteristic of high utility in active origami structures.

A phase diagram associated with SMAs is illustrated in Fig. C.1. The experimentally determined transformation temperatures at zero stress are denoted *martensite start temperature* M_s, *martensite finish temperature* M_f, *austenite start temperature* A_s, and *austenite finish temperature* A_f. The *stress influence coefficients* C^A and C^M are the slopes of the boundaries where transformation starts and ends as indicated in Fig. C.1.

An *actuation cycle* for an SMA is shown schematically in Fig. C.2. The thermomechanical loading path in the phase diagram (cooling and heating at constant stress in this example) is shown in Fig. C.2a while the corresponding SMA response in the strain-temperature space is shown in Fig. C.2b. The point 0 denotes the

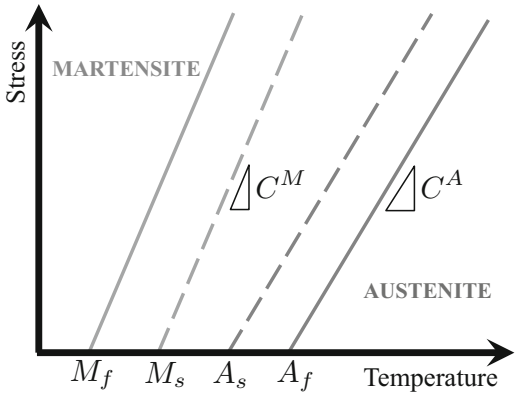


Fig. C.1 Schematic of the stress-temperature phase diagram of an SMA

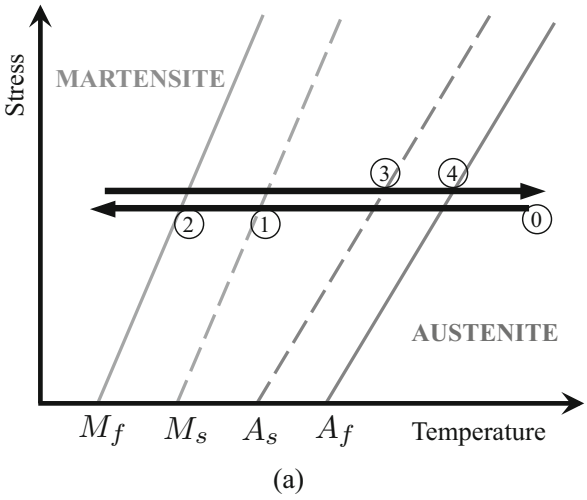


Fig. C.2 Schematic of an actuation cycle for an SMA at constant stress: (**a**) Thermomechanical loading path in the phase diagram (cooling and heating at constant stress); (**b**) SMA response in the strain-temperature space

initial state of the material (100% austenite). During cooling, the SMA exhibits a thermoelastic response until it reaches state 1 (at the martensite start boundary in the phase diagram). After the SMA reaches state 1, forward transformation occurs and transformation strains are generated until the SMA reaches state 2 (at the martensite finish boundary). During heating, after the SMA reaches state 3 (at the austenite start boundary), reverse transformation occurs and transformation strains are recovered until the SMA reaches state 4 (at the austenite finish boundary).

After briefly reviewing the main actuation characteristics of SMAs, we now describe the model that allows us to simulate such an actuation response. The detailed description of the constitutive model for monolithic SMAs presented here is provided in [21] and we only summarize the main equations used. The model has been previously applied to the analysis of a large spectrum of active structures having SMA components [22, 23]. The internal state variables associated with the inelastic transformation process of SMAs are the *martensite volume fraction* ξ and the second-order *transformation strain tensor* $\boldsymbol{\varepsilon}^t$. The martensite volume fraction ξ ranges from 0 (SMA in 100% austenite phase) to 1 (SMA in 100% martensite phase). The stress tensor $\boldsymbol{\sigma}$ is given as follows (cf. (8.18)):

$$\boldsymbol{\sigma} = \boldsymbol{C}(\xi)\left(\boldsymbol{\varepsilon} - \boldsymbol{\alpha}(T - T_0) - \boldsymbol{\varepsilon}^t\right), \tag{C.8}$$

The fourth-order stiffness tensor $\boldsymbol{C}(\xi)$ is determined as follows:

$$\boldsymbol{C}(\xi) \;=\; \left(\boldsymbol{S}^A + \xi(\boldsymbol{S}^M - \boldsymbol{S}^A)\right)^{-1} \;=\; \left(\boldsymbol{S}^A + \xi \Delta \boldsymbol{S}\right)^{-1}. \tag{C.9}$$

where $\boldsymbol{S}^A$ is the fourth-order compliance tensor of austenite and $\boldsymbol{S}^M$ is the fourth-order compliance tensor of martensite, both assumed isotropic and therefore being defined by Young's moduli E^A and E^M and Poisson's ratios ν^A and ν^M. The following relation between the rates of $\boldsymbol{\varepsilon}^t$ and ξ is assumed [21]:

$$\dot{\boldsymbol{\varepsilon}}^t = \boldsymbol{\Lambda}^t \dot{\xi}, \qquad \boldsymbol{\Lambda}^t = \begin{cases} \boldsymbol{\Lambda}^t_{\text{fwd}}; & \dot{\xi} > 0 \\ \boldsymbol{\Lambda}^t_{\text{rev}}; & \dot{\xi} < 0. \end{cases} \tag{C.10}$$

In the above relation, $\boldsymbol{\Lambda}^t$ is the second-order transformation direction tensor and its expressions for forward and reverse transformation are given as follows:

$$\boldsymbol{\Lambda}^t_{\text{fwd}} = \frac{3}{2} H^{\text{cur}} \frac{\boldsymbol{\sigma}'}{\bar{\sigma}}, \qquad \boldsymbol{\Lambda}^t_{\text{rev}} = \frac{\boldsymbol{\varepsilon}^{t-r}}{\xi^r}, \tag{C.11}$$

where H^{cur} is the transformation strain magnitude for full transformation, $\boldsymbol{\sigma}'$ is the second-order deviatoric stress tensor, $\bar{\sigma}$ is the von Mises stress, $\boldsymbol{\varepsilon}^{t-r}$ is the second-order transformation strain tensor at the cessation of forward transformation, and ξ^r is the martensite volume fraction at the cessation of forward transformation. The

deviatoric stress tensor $\boldsymbol{\sigma}'$ is given as $\boldsymbol{\sigma}' = \boldsymbol{\sigma} - \frac{1}{3}\,\mathrm{tr}(\boldsymbol{\sigma})\mathbf{1}$, where $\mathbf{1}$ is second-order identity tensor and $\mathrm{tr}(\cdot)$ is the *trace* operator. The von Mises stress $\bar{\sigma}$ is given as $\bar{\sigma} = \left(\frac{3}{2}\|\boldsymbol{\sigma}'\|^2\right)^{\frac{1}{2}}$, where $\|\cdot\|^2$ denotes the *inner product* of the enclosed tensor. It is noted in (C.11) that the direction of the transformation strains generated during forward transformation $\boldsymbol{\Lambda}^t_{\mathrm{fwd}}$ corresponds to that of the deviatoric stress $\boldsymbol{\sigma}'$. Therefore, the transformation strains generated by an SMA are *isochoric* (i.e., they preserve volume). The transformation strain magnitude for full transformation H^{cur} is given as follows [21]:

$$H^{\mathrm{cur}}(\bar{\sigma}) = \begin{cases} H_{\mathrm{min}}; & \bar{\sigma} \leq \bar{\sigma}_{\mathrm{crit}} \\ H_{\mathrm{min}} + (H_{\mathrm{sat}} - H_{\mathrm{min}})(1 - e^{-k(\bar{\sigma} - \bar{\sigma}_{\mathrm{crit}})}); & \bar{\sigma} > \bar{\sigma}_{\mathrm{crit}}, \end{cases} \tag{C.12}$$

where H_{min} is the magnitude of the observable two-way shape memory effect strain, H_{sat} is the magnitude of the maximum transformation strain attained by the SMA, and $\bar{\sigma}_{\mathrm{crit}}$ denotes the critical von Mises stress below which $H^{\mathrm{cur}} = H_{\mathrm{min}}$. The parameter k dictates the rate at which H^{cur} exponentially evolves from H_{min} to H_{sat} as the von Mises stress $\bar{\sigma}$ increases.

Certain constraints on the evolution of martensite volume fraction ξ must be defined to capture the response of the SMA shown in the phase diagram (Fig. C.1) [21]. To this end, we introduce the transformation function Φ^t. The constraints on the evolution of ξ are then given as follows:

$$\Phi^t \leq 0, \qquad \dot{\xi}\Phi^t = 0, \qquad 0 \leq \xi \leq 1. \tag{C.13}$$

The previous constraints are known as the Kuhn-Tucker conditions [21]. The following branched form of Φ^t is used:

$$\Phi^t = \begin{cases} \Phi^t_{\mathrm{fwd}}; & 0 \leq \xi < 1, \quad \dot{\xi} \geq 0 \\ \Phi^t_{\mathrm{rev}}; & 0 < \xi \leq 1, \quad \dot{\xi} \leq 0, \end{cases} \tag{C.14}$$

where the transformation functions for forward and reverse transformation (Φ^t_{fwd} and Φ^t_{rev}, respectively) are given as follows:

$$\begin{aligned} \Phi^t_{\mathrm{fwd}} = (1 - D)\boldsymbol{\sigma} : \boldsymbol{\Lambda}^t_{\mathrm{fwd}} + \tfrac{1}{2}\boldsymbol{\sigma} : \Delta\boldsymbol{S}\boldsymbol{\sigma} + \rho\Delta s_0 T - \rho\Delta u_0 \\ - \left(\tfrac{1}{2}a_1(1 + \xi^{n_1} - (1 - \xi)^{n_2}) + a_3\right) - Y^t_0, \end{aligned} \tag{C.15}$$

$$\begin{aligned} \Phi^t_{\mathrm{rev}} = -(1 + D)\boldsymbol{\sigma} : \boldsymbol{\Lambda}^t_{\mathrm{rev}} - \tfrac{1}{2}\boldsymbol{\sigma} : \Delta\boldsymbol{S}\boldsymbol{\sigma} - \rho\Delta s_0 T + \rho\Delta u_0 \\ + \left(\tfrac{1}{2}a_2(1 + \xi^{n_3} - (1 - \xi)^{n_4}) - a_3\right) - Y^t_0. \end{aligned} \tag{C.16}$$

The parameters $\rho\Delta s_0$, $\rho\Delta u_0$, a_1, a_2, a_3, D, and Y_0^t are calibrated from the experimentally determined transformation temperatures at zero stress (M_s, M_f, A_s, and A_f, see Fig. C.1), the stress influence coefficients (C^A and C^M, see Fig. C.1), and parameters defining H^{cur} (see (C.12)) as follows:

$$a_1 = \rho\Delta s_0(M_f - M_s), \tag{C.17}$$

$$a_2 = \rho\Delta s_0(A_s - A_f), \tag{C.18}$$

$$a_3 = -\frac{a_1}{4}\left(1 + \frac{1}{n_1+1} - \frac{1}{n_2+1}\right) + \frac{a_2}{4}\left(1 + \frac{1}{n_3+1} - \frac{1}{n_4+1}\right), \tag{C.19}$$

$$\rho\Delta u_0 = \frac{\rho\Delta s_0}{2}\left(M_s + A_f\right), \tag{C.20}$$

$$Y_0^t = \frac{\rho\Delta s_0}{2}(M_s - A_f) - a_3, \tag{C.21}$$

$$\rho\Delta s_0 = \frac{-2\left(C^M C^A\right)\left[H^{\text{cur}}(\sigma) + \sigma\,\partial_\sigma H^{\text{cur}}(\sigma) + \sigma\left(\frac{1}{E^M} - \frac{1}{E^A}\right)\right]}{C^M + C^A}\Bigg|_{\sigma=\sigma^*}, \tag{C.22}$$

$$D = \frac{\left(C^M - C^A\right)\left[H^{\text{cur}}(\sigma) + \sigma\,\partial_\sigma H^{\text{cur}}(\sigma) + \sigma\left(\frac{1}{E^M} - \frac{1}{E^A}\right)\right]}{\left(C^M + C^A\right)\left[H^{\text{cur}}(\sigma) + \sigma\,\partial_\sigma H^{\text{cur}}(\sigma)\right]}\Bigg|_{\sigma=\sigma^*}, \tag{C.23}$$

where the smooth hardening parameters $n_i \in (0, 1]$, $i = 1, \ldots, 4$, determine the level of smoothness in the transitions between transformation and thermoelastic loading domains. The calibration stress for the stress influence coefficients C^M and C^A is denoted σ^*.

For the plane stress, (C.8) is simplified and only the following stress components are considered:

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \boldsymbol{C}^P(\xi) \begin{bmatrix} \varepsilon_{11} - \alpha(T - T_0) - \varepsilon_{11}^t \\ \varepsilon_{22} - \alpha(T - T_0) - \varepsilon_{22}^t \\ \varepsilon_{12} - \varepsilon_{12}^t \end{bmatrix}, \tag{C.24}$$

where $\boldsymbol{C}^P(\xi) \in \mathbb{R}^{3\times 3}$ is the stiffness matrix determined via (C.9) and simplified for plane stress.

References

1. J.P. Keener, *Principles of Applied Mathematics* (Addison-Wesley, New York, 1988)
2. R.A. Horn, C.R. Johnson, *Matrix Analysis* (Cambridge University Press, Cambridge, 2012)
3. G. Strang, *Linear Algebra and Its Applications* (Harcourt Brace Jovanovich, San Diego, 1988)
4. A. Ben-Israel, T.N.E. Greville, *Generalized Inverses: Theory and Applications*, vol. 15 (Springer, New York, 2003)
5. W.M. Lai, E. Krempl, D. Ruben, *Introduction to Continuum Mechanics* (Butterworth-Heinemann/Elsevier, Burlington/Massachusetts, 2009)
6. M.E. Gurtin, *An Introduction to Continuum Mechanics* (Academic, New York, 1981)
7. J.G. Simmonds, *A Brief on Tensor Analysis* (Springer, New York, 2012)
8. R.M. Bowen, C.-C. Wang, *Introduction to Vectors and Tensors* (Dover Publications, Mineola, 2008)
9. W.S. Slaughter, *The Linearized Theory of Elasticity* (Birkhäuser, Boston, 2002)
10. J.N. Reddy, *Mechanics of Laminated Composite Plates: Theory and Analysis* (CRC Press, Boca Raton, 1997)
11. D.J. Leo, *Engineering Analysis of Smart Material Systems* (Wiley, Hoboken, 2007)
12. R.C. Smith, *Smart Material Systems: Model Development* (SIAM, Philadelphia, 2005)
13. D.C. Lagoudas, Z. Bo, The cylindrical bending of composite plates with piezoelectric and SMA layers. Smart Mater. Struct. **3**(3), 309 (1994)
14. D.C. Lagoudas (ed.), *Shape Memory Alloys: Modeling and Engineering Applications* (Springer, New York, 2008)
15. D.J. Hartl, D.C. Lagoudas, Aerospace applications of shape memory alloys. Proc. Inst. Mech. Eng. G: J. Aerosp. Eng. **221**(4), 535–552 (2007)
16. S. Barbarino, E.S. Flores, R.M. Ajaj, I. Dayyani, M.I. Friswell, A review on shape memory alloys with applications to morphing aircraft. Smart Mater. Struct. **23**(6), 063001 (2014)
17. D.J. Hartl, J.H. Mabe, O. Benafan, A. Coda, B. Conduit, R. Padan, B. Van Doren, Standardization of shape memory alloy test methods toward certification of aerospace applications. Smart Mater. Struct. **24**(8), 082001 (2015)
18. Y. Furuya, H. Shimada, Shape memory actuators for robotic applications. Mater. Des. **12**(1), 21–28 (1991)
19. M. Sreekumar, T. Nagarajan, M. Singaperumal, M. Zoppi, R. Molfino, Critical review of current trends in shape memory alloy actuators for intelligent robots. Ind. Robot Int. J. **34**(4), 285–294 (2007)
20. J.M. Jani, M. Leary, A. Subic, M.A. Gibson, A review of shape memory alloy research, applications and opportunities. Mater. Des. **56**, 1078–1113 (2014)
21. D. Lagoudas, D. Hartl, Y. Chemisky, L. Machado, P. Popov, Constitutive model for the numerical analysis of phase transformation in polycrystalline shape memory alloys. Int. J. Plast. **32–33**, 155–183 (2012)
22. D.J. Hartl, D.C. Lagoudas, F.T. Calkins, Advanced methods for the analysis, design, and optimization of SMA-based aerostructures. Smart Mater. Struct. **20**(9), 094006 (2011)
23. S.D. Oehler, D.J. Hartl, R. Lopez, R.J. Malak, D.C. Lagoudas, Design optimization and uncertainty analysis of SMA morphing structures. Smart Mater. Struct. **21**(9), 094016 (2012)

Index

E. A. Peraza Hernandez et al., *Active Origami*,
https://doi.org/10.1007/978-3-319-91866-2