Transportation Analysis

Managing Editors

Lucio Bianco, Rome

Agostino La Bella, Rome

Lucio Bianco · Amedeo R. Odoni (Eds.)

Large Scale Computation and Information Processing in Air Traffic Control

With 63 Figures

Springer-Verlag

Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Professor Dr. Lucio Bianco
University of Roma Tor Vergata
and
Consiglio Nazionale delle Ricerche
Progetto Finalizzato Trasporti
Viale dell' Università 11
I-00185 Rome, Italy

Professor Dr. Amedeo R. Odoni
Massachusetts Institute of Technology
Department of Aeronautics and Astronautics
Room 33-404
Cambridge, MA 02139, USA

ISBN-13: 978-3-642-84982-4 **e-ISBN-13: 978-3-642-84980-0**
DOI: 10.1007/978-3-642-84980-0

Softcover reprint of the hardcover 1st edition 1993

2142/7130-543210 – Printed on acid-free paper

FOREWORD

This volume is a compendium of papers presented during an Advanced Seminar on Air Traffic Control (ATC) that took place in Capri, Italy on October 28-31, 1991. The Seminar was organized by the Progetto Finalizzato Trasporti of the Italian National Research Council. The papers presented in the Seminar dealt with a wide range of topics which are currently important in ATC. For example, there were papers on such subjects as recent developments in primary and secondary radar technologies, communications networks and protocols, and the future uses of satellite-based communications, navigation and surveillance in ATC. However, all the papers contained in the volume were selected exclusively from that set of papers that addressed some aspect of the main area of emphasis in the Seminar, namely massive data-processing requirements and computer-intensive problems in ATC.

Data-processing requirements in ATC have grown enormously over the years. Obviously, the rapid increase in air traffic volumes in most of the world is one of the factors that has contributed to this growth. However, two other developments have contributed much more significantly: first, the ATC system now collects (mostly automatically) immensely more "information per flight" than in the past; and, second, as the system's complexity increases and as it becomes more tightly interconnected geographically, so grows the need to communicate, process and "filter" the data presented to the system's various components.

Moreover this growth in data-processing needs will accelerate, if anything, in the future. For example, recently-introduced, advanced Flight Management Systems (FMS) are now capable, in principle, of transmitting, via satellite, practically any imaginable type of information about the status of a flight and of the weather environment around it to ATC systems anywhere in the world. On the ground, concerns about the workload of air traffic controllers and about maximizing the effectiveness of these individuals in their role as traffic managers will undoubtedly lead to concerted efforts toward providing to each controller position "the right amount of data, in the right form and at the right time".

The same developments - more traffic, more information, higher complexity, higher workload - will intensify the already pressing need to take full advantage of the many opportunities for automation in ATC. This means using the computer to perform not only routine data-manipulation functions - as has been the rule until very recently - but also to provide assistance to the controller in carrying out "intelligent" tasks such as traffic flow management and aircraft spacing and separation assurance. To accomplish this objective, it is necessary to develop sophisticated Decision Support Systems (DSS), drawing on the methodologies of such fields as large data-base management and the decision sciences (operations research, optimal control, artificial intelligence, statistical decision theory).

In the first paper of the volume, Bertocchi and Zenios provide a general introduction to the subject of parallel and distributed computing, technologies vital to the future of air traffic control. The paper introduces the terminology of the field and points to some of the fundamental advantages that parallel and distributed computing enjoy. The authors also suggest that some large-scale optimization problems in air traffic flow management have a structure which makes them "natural" candidates for solution through parallel computation. This is an observation confirmed by several research teams in recent years. For example, another paper in this volume (Andreatta, Odoni and Richetta) shows how a specific and important problem in this family can be decomposed into many parallel problems, each of which can be solved by using standard and efficient network optimization algorithms.

In the successive paper, McCarthy offers an overview of one of the most comprehensive data collection, processing and dissemination systems currently under development for ATC. The new Aviation Weather System for the United States will assemble weather data from a very large number of sensors on the ground, on aircraft, on satellites or airborne by themselves (e. g. balloons) and will distribute and display this information in a timely manner, as needed. The system will enhance greatly both the efficiency and the safety of air traffic and offers a paradigm, on a large scale, for other aviation weather systems of the future.

Even ten years ago, relatively little attention was being paid to flow management, the activity of controlling the flow of air traffic for the purpose of maximizing efficiency and minimizing the cost of delays. Today, with congestion reaching the dimensions of a crisis at many of the world's busiest airports and airspace sectors, this attitude has changed dramatically. Flow management is now seen as one of the two fundamental activities of the ATC system, the other one being the function of monitoring and assuring the safe separation between aircraft in the air and on airfields. The next three papers in the volume address flow management in ATC, each from a different perspective.

Bianco and Bielli provide an overall context in their introduction. It is pointed out that, given a set of ATC facilities and their associated capacities, flow management can be viewed as a hierarchical problem, that begins with strategic decisions which are made months in advance and end with short-term tactical interventions with a time-horizon of minutes. The authors then review a series of models for various stages of the problem that have been developed by them and their colleagues over the years, both for en route and for terminal airspace. This paper points out that flow management problems can be extremely complex computationally. Selecting the right level of detail is essential, if a model is to have practical utility.

The paper by Winer describes an extensive array of simulation and optimization tools that have been or are now being developed by the Federal Aviation Administration (FAA) as decision support tools for air traffic flow managers in the United States. As noted earlier in this introduction, the methodologies that are being used can be vastly different. One of the tools described is an expert system, others use discrete mathematical optimization, while yet another set serve essentially as graphic traffic visualization aids for the controllers. Winer (and Zellweger later in the volume) also refers to "rapid prototyping", a model development approach used with considerable success in recent years by the FAA: an initial version of the model is developed quickly and the model is subsequently evaluated and refined with the participation of those air traffic controllers who will be the model's eventual users.

Perhaps the most important problem in ATC flow management, at least in terms of cost implications, is that of developing "ground-holding strategies", i.e., rescheduling the take-off times of aircraft that are expected to encounter delays. If an aircraft is certain to suffer some delay, it is better that as much of this delay as possible be absorbed on the ground before take-off, rather than in the air. Andreatta, Odoni and Richetta show that the ground-holding problem (GHP) is a difficult one with dynamic, stochastic and combinatorial characteristics and provide a survey of a series of models they and their colleagues have developed to address it. The selection of the model to be used in any given ATC environment depends on the quality and timeliness of the data available to ATC regarding the current and future status of airport capacities and of air traffic demand during the course of daily operations. The paper also describes an extensive set of computational experiments that suggest the feasibility of developing highly beneficial automated tools that would provide decision support to ATC regarding ground-holding strategies.

Applications of artificial intelligence technologies are increasingly appearing in ATC, so far primarily in experimental form. In his paper Simpson describes the development of two expert systems to assist the planning of air traffic flows in a busy terminal area. One of the expert systems generates in real-time a set of conflict-free descent paths for aircraft entering a terminal area to land. The paths adhere to patterns which are immediately recognizable and familiar to pilots and air traffic controllers. The second system assists local air traffic controllers at a major airport to develop a plan for the runway configurations to be used on any given day at the airport, taking into consideration weather conditions, traffic demand, noise abatement policies, etc. The paper also contains some observations on the types of ATC functions for which the use of expert systems would seem appropriate, as well as on the difference between typical "static" applications of expert systems and the applications encountered in ATC where the temporal element is critical.

The final two papers in the volume deal with the development of simulation environments for the development and testing of new ATC concepts and systems. As ATC costs and complexity increase, the availability of such environments has become critical. Zellweger describes a large-scale effort that has been undertaken by the FAA to develop a National Simulation Capability (NSC) for ATC in the United States. The NSC will be a geographically distributed group of simulation facilities that will rely, at least initially, on already existing simulations. Its objective is to assist the FAA in understanding the entire range of implications of proposed innovations, in order to ensure that such innovations are truly effective and that they can be properly integrated into the overall ATC system. Thus, the NSC will emphasize a "systems" viewpoint and will perform such tasks as: concept validation at a system-wide level; systems engineering, planning and development; and human factors evaluations and assessments. Among other benefits, the NSC is seen as an instrument for involving at an early stage all interested parties in the development of new ATC concepts, procedures or equipment. Eventually, the NSC can also become a mechanism for accelerating the acceptance and implementation of ATC enhancements and innovations. A few NSC experiments have already been completed at this point and several more are in progress.

The paper by Benoit, Swiestra and Garcia describes a tool that could be typical of the kinds of capabilities that might reside in the NSC or its future counterparts elsewhere in the world. That tool is STANS, a simulation testbed for experimenting with alternative airspace configurations and for assessing the impacts of new procedures and equipment in an ATC region. STANS is designed with the objective of serving as a low-cost, flexible and transportable simulation environment. This testbed would be used to "screen" new concepts by evaluating their performance and potential in a preliminary way. Presumably, the most promising concepts would then be subjected to more detailed testing and experimentation.

Overall, the papers in this volume and the other papers presented during the Advanced Seminar in Capri provide additional evidence of the strides being made in advanced ATC research and development. It is noteworthy that, to varying degrees, the work described herein has benefited not only from advances in data processing technologies and the decision sciences, as might be expected, but also from striking recent progress in such areas as software engineering, telecommunications, information-display technologies and the development of low-cost sensors and other data acquisition technologies.

It was the conclusion of the Seminar that there are good reasons to be optimistic about future enhancements to existing ATC systems on both a local and a global level, especially if the currently favorable political climate vis-a-vis ATC research, development and capital investments can be maintained. What can be considered as the first comprehensive terminal area automation system has already been deployed in Frankfurt under the name of COMPAS. Other automation systems are now in advanced stages of development or near implementation, including the terminal area MAESTRO system in France and the Terminal Area Air Traffic Control Automation (TATCA), Automated En Route Air Traffic Control (AERA) and Advanced Traffic Management System (ATMS) in the United States. The prevailing feeling in the technical community today is that ATC may finally be on the verge of its long-awaited transition toward advanced automation. Some of the ideas, concepts or systems described in this volume will undoubtedly contribute to current or future developments along these lines.

Lucio Bianco　　　　Amedeo R. Odoni

Acknowledgment

We wish to thank M. Bertocchi, G. Galati, A. Scognamiglio and A.M. Innocenzi for their contributions to organizing and conducting the Advanced Seminar in Capri on which this volume is based.

We also gratefully acknowledge C. Petrosillo and G. Totaro for their carefull assistance in the editorial work.

CONTENTS

LARGE SCALE ARCHITECTURES AND PARALLEL PROCESSING IN AIR-TRAFFIC CONTROL[1]

Marida Bertocchi

Department of Mathematics, Statistics, Informatics and Applications University of Bergamo, Bergamo, ITALY.

Stavros A. Zenios

Decision Sciences Department, The Wharton School University of Pennsylvania, Philadelphia, USA.

The performance of von Neumann, serial, computers improved by a factor of 10^6 over the period 1955-1990. Within the next decade we expect to see similar improvements, delivered by parallel supercomputers. As this new technology matures, it shifts from being a research topic in its own right, to being a fundamental tool for scientific, engineering and business applications. Multiprocessor systems that once cost several millions of dollars and were found only in Government laboratories are now very affordable. They have proliferated in academia, high-technology industries, and several business applications.
In this paper we review recent developments in the general area of large scale computer architectures, with particular emphasis on parallel processing. The impact of this technology in different areas of application — real time management of traffic, crew scheduling — is highlighted. The paper concludes with a discussion of several applications from Air Traffic Control (ATC) that are well suited for parallel processing, and stand to gain substantially from this technological innovation.

1. Introduction

Parallel computing has received increased attention from the scientific and business communities over the last five years or so. Reasons for the interest in this technological development are numerous. Just to name a few: limitations of the serial, von Neumann computer, economies of scale in manufacturing multiprocessors, the natural parallelism inherent in several applications and so on. It can be said that parallel computing has moved from being a research

[1] Research partially supported by NSF grants CCR-9104042 and SES-00216, and AFOSR grant 91-0168 and MURST 60% Italian grant 91

topic in its own right to the role of a fundamental tool for furthering our understanding of natural phenomena (like the weather), physical phenomena (like the flow of oil in reservoirs), engineering processes (like computer aided design and manufacturing) and management problems (like the logistics associated with the deployment of numerous military units during operation Desert Shield). The recent commitments of big computer manufacturers to massive parallelism — such as IBM's collaboration with Thinking Machines Corporation, and Digital's arrangements with Maspar — give new impetus to the parallel computing industry. The transition of parallelism to a broad marketplace is fast approaching.

In this article we give an overview of large scale architectures and parallel processing, and discuss its potential application to problems from Air-Traffic Control (ATC). We also highlight the impact of this computing technology on some specific applications that are of central interest to air transport system modeling: the real time management of traffic and the optimal scheduling of airline crews. The rest of this paper is organized as follows: Section 2 reviews developments in parallel processing, Section 3 reviews the current use of this technology in different applications and Section 4 discusses several ATC models where parallel processing could have a substantial impact. Some concluding remarks are given in Section 5. We begin with a general discussion on parallel computing technology.

2. What is Parallel- and Super-Computing

Computing power improved by a factor of 10^6 in the period 1955-1990 and it is expected to improve by that much more just within the next decade. How can this accelerated improvement be sustained? The answer is found in novel computer architectures: Multiple, semi-autonomous processors, each with its own local memory, coordinating for the solution of a single problem. There is a lot of flexibility in how this architecture can be realized. Indeed, this flexibility has hampered acceptance of parallel computers by a broad spectrum of industrial and business users, since no *standard* parallel architecture seemed possible. Before we proceed with a classification of computer architectures it is important to understand the distinction between a *supercomputer* and a *parallel* computer.

Super computer is defined as the fastest machine at any point in time. Some computers, however, may be very efficient for some tasks (e.g., array processing), while they lack substantial capability for others (e.g., list processing). Hence, this definition is rather vague.

Parallel computer refers to a class of computer architectures, with multiple processing units. Detailed classifications are given below. A parallel computer does not automatically qualify as a supercomputer. However, it is widely accepted that improved supercomputing performance can only be achieved using parallel architectures.

A broad classification of parallel architectures was offered in Flynn [1972]. While several extensions have been added (see, for example, Hockney and Jeeshope [1981]) Flynn's taxonomy is still fundamental in understanding parallel architectures. He identified four classes, depending on the interaction of the instructions of a program with the data of the problem:

SISD (*Single Instruction stream Single Data stream.*) Systems in this class execute a single instruction on a single piece of data before moving on to the next instruction and the next piece of data. Traditional uniprocessor, scalar, computers fall under this category.

SIMD (*Single Instruction stream Multiple Data stream.*) A single instruction can be executed simultaneously on multiple data. This of course implies that the operations of an algorithm are identical over a set of data, and that data can be arranged for concurrent execution. This is one type of parallel computing and some of the successful instances of SIMD machines are the Connection Machine CM-2, the Active Memory Technology DAP and MasPar.

MISD (*Multiple Instruction stream Single Data stream.*) Multiple instructions can be executed concurrently on a single piece of data. This form of parallelism has not received, to our knowledge, extensive attention from researchers. It appears in Flynn's taxonomy for the sake of completeness.

MIMD (*Multiple Instruction stream Multiple Data stream.*) Multiple instructions can be executed concurrently on multiple pieces of data. The majority of parallel computer systems fall into this category. Multiple instructions indicate the presence of independent code modules that may be executing independently from each other. Each module may be operating either on a subset of the data of the problem, have copies of all the problem data, or access all the data of the problem together with the other modules in a way that avoids read/write conflicts. The Intel hypercubes, the CRAY X-MP and Y-MP, the Alliant, and the Convex are MIMD systems.

A mode of computing deserving special classification is that of *vector* computers. While they fall in the category of SIMD machines, vector computers constitute a class of their own. This is due to the appearance of vector capabilities in parallel systems and most supercomputers. The development of algorithms and software for vector computers, such as the CRAY, presents different problems than the design of algorithms for a system with multiple processors that operate synchronously on multiple data - such as, for example, the Connection Machine CM-2.

Another major classification of parallel machines is based on their memory organization: *Shared memory* machines are those where multiple processors can access directly all the memory of the system. Communication among processors takes place by writing and reading data to and from this common memory. The algorithm designer has to ensure that no read or write conflicts arise (i.e., no two processors access the same memory location simultaneously) otherwise the integrity of the data cannot be guaranteed. ***Distributed memory*** machines are those where each processor has direct access only to some local memory. Communication among processors takes place by exchanging messages through some communication network. Accessing the local memory is a very efficient operation, while communication among processors is much more expensive. Distributed memory systems are equipped with sophisticated routing algorithms that direct messages from the sending to the receiving processors. Nevertheless, it is the algorithm designer who has to specify which processors communicate at different phases of the execution of an algorithm.

2.1 Some Unifying Concepts of Parallel Computing

Flynn's taxonomy created the impression that a uniform model of parallel computation were not possible. However, some recent concepts allow end-users to a get a uniform view of the computer system, independently of the underlying architecture.

Single Program Multiple Data (SPMD). The distinction between MIMD and SIMD gives way to the unified notion of SPMD. Parallelism can be viewed as the operation of a single program on multiple sets of problem data. Each data set is operated upon by its very own processing elements.

Using a linear programming (LP) solver to solve multiple related instances of a linear program — as, for example, in the analysis of multiple scenarios in a portfolio management model — is an SPMD application. Whether each solver executes exactly the same pivot steps or not is irrelevant to the programmer. Of course an SIMD computer would impose this restriction, and could be inefficient if each solver should execute different pivot steps. If most of the solvers execute identical steps, efficiency will be substantially improved. An MIMD computer might be more efficient. But the main point is that the parallel program looks, at least to the analyst, the same for both computers.

SPMD has been motivated by the computing paradigm of SIMD architectures, as introduced by Hillis [1985] for his Connection Machine. However, it is likely that MIMD architectures will be more effective environments for the execution of SPMD applications. The advantage of SPMD is that the user need not worry about the details of the computer architecture.

Distributed Virtual Memory (DVM). The distinction between distributed memory and shared memory architectures becomes fuzzy with the introduction of the notion of distributed virtual memory. This idea was originated by distributed computing environments — like LINDA from Scientific Computing Associates and Parasoft EXPRESS; see Carriero and Gelernter [1989].

Within a DVM environment a distributed architecture can be viewed as a system with (shared) virtual memory. The algorithm designer, of course, has to deal with the problem of data integrity to avoid incorrect results. However, the fact that memory is implemented in a distributed environment is of no direct consequence to him. It is the developer of the distributed computing environment who has to keep track of global objects. Where is a global object (like a common variable) stored? One approach would be to distribute all global objects to local memories. Another approach would be to distribute the address of all global objects to local memories, and let each processor fetch the data only if and when they are being used by the application. Of course, these designs tradeoff space with time efficiency. The point, however, is that the analyst need not worry about the shared/distributed memory distinction. Indeed, systems like LINDA and EXPRESS have made it possible to port applications between shared memory machines, distributed memory machines and distributed heterogeneous networks of workstations with little effect on performance and no reprogramming effort.

Heterogeneous Parallel Processing. In the early days of parallel processing — and in the quest for a "winner" parallel computer —

the debate on the right architecture was quite heated and often dogmatic. With the emergence of unifying computing paradigms — like SPMD and the distributed virtual memory — the notion of heterogeneous parallel processing is finding widespread acceptance. By this term we mean a parallel computing environment consisting of multiple machines — some of which could be parallel processors themselves - linked together through some communication network or a fast switching device.

For example, networks of workstations can be linked together in a most cost-effective form of parallel processing. Fiber optics provide data-transfer rates (100 Mbits/sec) that were until recently found only on tightly coupled systems. Software systems like LINDA or EXPRESS allow users to decouple their problems in a distributed virtual memory environment. Multiple workstations will then grab pieces of work from this environment, complete them at their own pace (depending on workload and the performance of each workstation) and return the results. There is no reason to assume that all the servers on the network are identical workstations, or even just workstations. Some of the servers could be more advanced parallel architectures. For example, an array processor attached to the network could be used to execute any linear algebra calculations, while workstations could be executing the less homogeneous operations.

2.2 Why Parallel and Super-Computing?

What are the factors that motivate the development of parallel architectures? Is this a sustainable development or just a scientific fad? The original motivation in designing vector supercomputers, like the CRAY 1S, was the limitations of serial von Neumann architectures. Systems with serial, scalar processing are limited by the speed with which signals can be communicated across different components. With the speed of light as an upper bound, we find current vector supercomputers operating near their limit as far as clock cycles are concerned. Parallel designs appear to be the only way to overcome this barrier.

Following the breakthroughs of Cray Research Inc. in the mid-seventies the computer industry realized that manufacturing parallel machines could also be cost effective. There are economies of scale in utilizing multiple processors around some common memory. After all, only a tiny fraction of a computer's silicon is found in the processing unit. Continued progress in very large scale integrated circuits provides additional justification for packing multiple processors in the same system.

Furthermore, as parallel machines became available, scientists from several disciplines realized that parallelism is a natural way to view their applications. For example, weather forecasting models can naturally be split by geographical regions, and forecasting models run separately for each region. Only boundary information needs to be communicated across adjacent regions. Tracking the flow of traffic on the high-altitude jet routes can be allocated to multiple processors: one for each control sector, or one for each route within the sector, or one for each aircraft. Information among processors would only be exchanged as aircraft change routes, or move from the jurisdiction of one control sector to the next.

In terms of cost effectiveness, there are a number of reasons to be optimistic about the future of parallel processing environments. The domain of "Performance versus R&D Investment" for computers — or any other technology for that matter — takes an S-shaped form; see Figure 1. At the early stages of the development effort progress is expensive and time consuming. As basic technological difficulties are being resolved progress accelerates, and this is captured by the steep middle part of the curve. Eventually, however, we see a slowdown in improvements as the original design concepts reach their limit. The rightmost, upper part of the curve is once more flat.

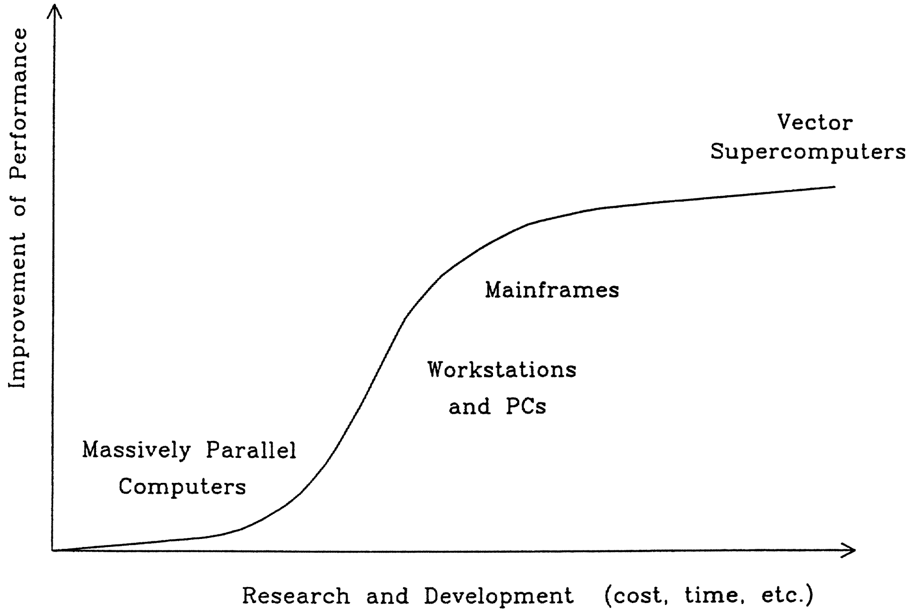


Figure 1. The S-shaped domain of the Rate of Improvement of Performance versus Research & Development effort for computer systems.

The S-shaped curve is easily justified from our observations about current parallel machines: massively parallel systems are at the early stages of their development cycle while vector supercomputers are at the later stage. Evidence to this can be seen from the diagram of computational power versus cost (Figure 2), obtained from the report of Deng el al. [1992]. From the CRAY X-MP to the CRAY Y-MP we have seen a shift to a lower cost-effectiveness ratio. Small improvements in performance came at a large cost. The cost-effectiveness ratio of the DEC mainframes remained almost constant: mainframes lie at the middle (linear) part of the S-shaped development domain. Finally, personal computers became much more cost-effective, as they are moving along the lower part of the S-shaped domain. That is, their performance improves rapidly with little increase in their cost.

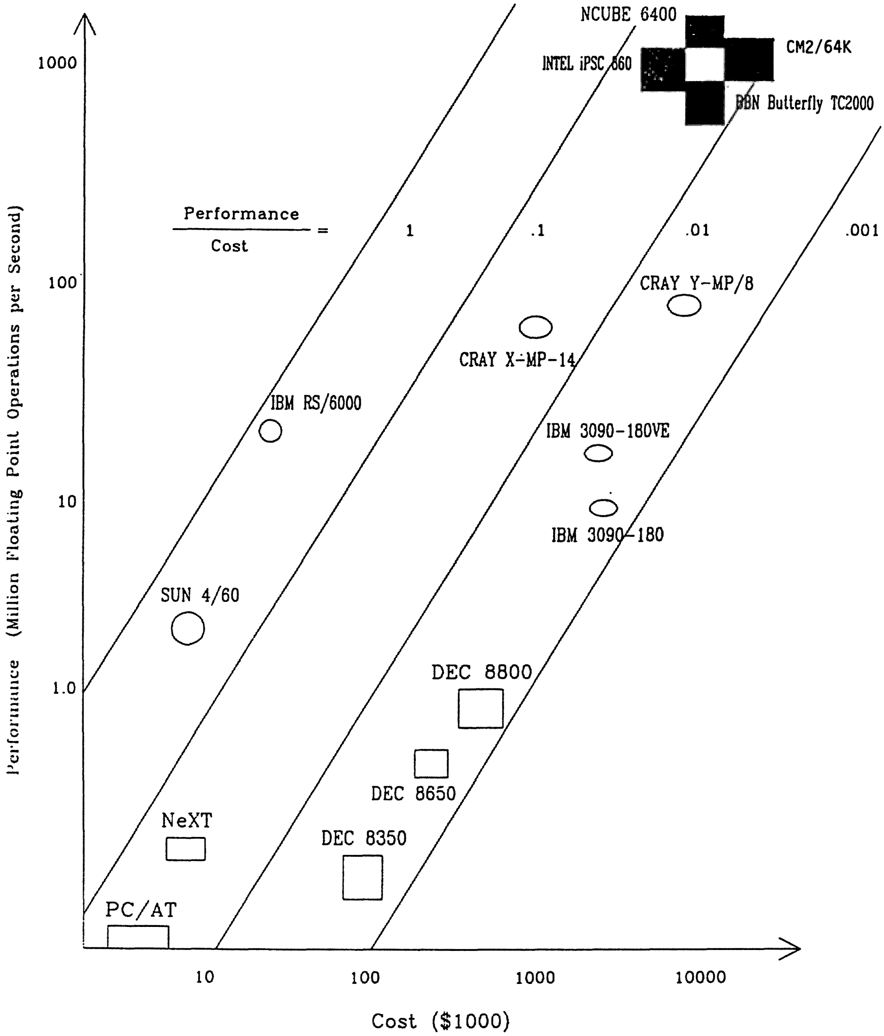


Figure 2. Computational power versus cost. Constant cost-effectiveness ratios are indicated by the lines A-D.

Similarly, we expect massively parallel architectures to improve very rapidly in performance, while experiencing marginal increase in cost. This is precisely where we see the potential of massive parallelism: not so much in what has been achieved to-date, but what is expected to happen in the next several years.

We conclude that parallelism is an unavoidable step in the development process of computing technology. Even major breakthroughs in serial processing technology will not change this development. Instead, they will accelerate it: for instance, the earlier hypercube systems by Intel were based on hardware found in personal computers. Their latest models coordinate a thousand processors that are typically found in workstations.

For the purpose of this study it suffices to acknowledge the growing importance and capabilities of parallel computing as well as traditional supercomputing. We are more interested in the potential effects of both of these technologies on ATC than we are in the debate about their relative merits.

2.3 Measuring Computing Performances: The Laws of Amdahl and Gustasfon

The following question needs to be resolved: "How will performance scale as we add more processors?" Amdahl's law specifies that improvements in execution time *(speedup)* when moving from a serial machine to one with P processors is given by

$$Speedup = \frac{1}{s + p / P}$$

where s is the fraction of serial execution of the application, and p is the fraction of execution time that can be performed in parallel ($s + p = 1$). Even with an infinite number of processors, speedup cannot exceed $1/s$. As scientists observed that 10% of a typical application could not be parallelized (input/output, initializations, serial bottlenecks of the algorithm) they concluded that a speedup of 10 was the best one could expect.

Research performed at Sandia National Laboratory over a period of years gradually brought the community out of this "mental" block. Parallel computers are not just used to solve 10 times faster an existing application. Instead, they were used to solve in about the same time

1000-fold bigger instances of the same problem. Speedups of 1000 were not uncommon on a hypercube with 1024 processors at Sandia. When an application is scaled in size, to fit in the larger number of processors, the serial part usually remains unchanged. What scales up is the parallel part. Hence, linear speedups can be expected. Consider a problem with a serial execution part s and a parallelizable part p. When a P processor system becomes available the application would be scaled. If the parallel part would scale linearly then the larger application would require $s+pP$ execution time. A modified Amdahl's law, due to E. Barsis from Sandia, is given in Gustafson [1988]:

$$\textit{Scaled speedup} = s + p \times P.$$

Viewed in this context the prospects of parallelism are much more promising. Indeed, it is misuse of the technology to try to solve faster existing applications. If the problem is currently being solved in an acceptable manner, why bother? The major thrust of parallelism is in solving applications that were not considered within reach with serial computers. In the next section we describe several such applications from diverse industries.

3. Parallel Processing Applications

We first describe some important management problems which share some characteristics and show some similarities with ATC applications. Parallel computing has been used effectively in solving these applications.

3.1 Logistics and the Management of Traffic

Recent advances in telecommunications, information technology, and automation in intelligent highway systems create an environment where real-time traffic management is made possible. For example, drivers of suitably equipped vehicles may have access to a control center via an Advanced Driver Information System (ADIS). They can obtain real-time information about the status (i.e., congestion level) of the transportation network. Going a step further, an Advanced Traffic Management System could provide route guidance instructions to drivers, or control the status of the network (for example, by changing the number of eastbound versus westbound lanes depending on traffic conditions).

The capability to collect information about the network, and communicate it to interested parties, is already available through ADIS. The ability to provide real time guidance and improved management of the network requires additional modeling and algorithmic research. For example, the central controller will need dynamic assignment capabilities. Given information about Origin-Destination (O-D) trip desires he should be able to route all traffic from their current position to the desired destinations. The assignment must take into account time-dependent link status conditions, like loadings, link travel times, capacity reducing incidents or weather conditions and so on. Furthermore, the analysts should be able to determine the time-dependent link flow patterns that result from the path choices made by motorists in response to the ADIS information.

The complexities of such a system are enormous. A multi-year investigation is currently under way at the University of Texas at Austin under H. Mahmassani, sponsored by the Federal Highway Administration (FHA) — his research team also includes G-L. Chang, L. Lasdon and one of the authors. An integral part of this investigation is the use of parallel computers for executing the various components of the model in real time. Indeed, some earlier methodological research has demonstrated that several components of a dynamic assignment model could be computed effectively on parallel architectures. These developments have prompted FHA to proceed with this very ambitious project, Mahmassani et al. [1991]:

1. The static traffic assignment problem, Wardrop [1952], has been studied recently by Chen and Meyer [1988]. Taking advantage of the

block diagonal structure of the constraint matrix they designed a special purpose decomposition algorithm that was suitable for implementation on a parallel architecture. Witness their results in modeling traffic in the city of Winnipeg by solving a nonlinear model with 140400 constraints and 382860 variables:

Processors	Solution time
1	30 hours
10	3 hours 50 minutes
16	2 hours 28 minutes

These results were obtained on a Crystal multicomputer, a cluster of 20 VAX 11/750, at the University of Wisconsin, Madison. Moving from the rather primitive technology of the VAX 11/750 to top range RISC workstations will provide at least an order of magnitude improvement. Hence, a model that would require a two-day run, can be completed within minutes on a network of top-range workstations.

2. The problem of distributing multiple distinct commodities — like, for example, different kinds of traffic, trucks, cars, aircraft and so on — over a transportation network is known as the *multicommodity network flow problem.* The essence of the problem is to distribute each commodity over the transportation network in the most cost-effective way, taking into account the limited capacity of the links that must be shared by all commodities. Such problems are usually of very large sige: the transportation network may cover large geographical regions, and a large number of commodities are typically present. Nevertheless, it has a very special structure: block-diagonal with a few coupling constraints. Each block reflects conservation of flow constraints for each commodity, while the coupling constraints ensure that the link capacities are not exceeded by the aggregate flow of all commodities.

   This special structure has prompted Schultz and Meyer [1991], and Pinar and Zenios [1992] to develop algorithms that are well suited for implementation on parallel architectures. The algorithms were used to solve logistics applications for the Military Airlift Command. Using a KORBX system, with its implementation of Karmarkar's algorithm, will take a day for the optimization of a 20-period planning model. The same size application can be solved within 10 minutes on a CRAY Y-MP with 8 processors when one takes advantage of parallelism, as well as the vector architecture (results from Pinar and Zenios [1992]):

Planning horizon	Rows x Columns	KORBX times	CRAY Y-MP times
10 day planning	15389 x 48763	3 hrs 20 min	1 min 36 sec
20 day planning	31427 x 105728	17 hrs	12 min 20 sec
Monthly planning	46453 x 154998	NA	42 min 46 sec

3. The problem of estimating Origin-Destination (O-D) matrices based on incomplete, and perhaps noisy, data can be modeled as a nonlinear (entropy) optimization problem with transportation constraints. It has been known in the literature as the *matrix balancing* problem, see Schneider and Zenios [1990]. Zenios and Iu [1990] and Zenios [1990] studied this problem in the context of parallel computing. Extremely large problems were solved within seconds of solution time on a CRAY X-MP/48 or the Connection Machine CM-2 with 32K processors. Typically such problems would take several hours, even on large mainframes with vector capabilities, like the IBM 3090. We provide below an illustrative comparison of the massively parallel algorithms, compared with the algorithm of Nagurney et al. [1990] executing on the vector supercomputer:

Dimension of O-D matrix	IBM 3090	CM-2
500 x 500	1 min 9 sec	18 sec
1000 x 1000	8 min 3 sec	22 sec
2000 x 2000	1 hr 3 min 43 sec	47 sec

The performance of the algorithm on the CM-2 led to the investigation of more general formulations of the matrix balancing model. In particular, Censor and Zenios [1991], proposed interval-constrained formulations of the matrix balancing model. These interval formulations allow us to solve problems when the observations on the O-D matrix are subjected to noise — a common phenomenon in practice — in which case the equality constrained formulation may not have a solution. This is an instance where the parallel computing technology motivated the development of new model formulations.

3.2 Airline Crew Scheduling

Scheduling an optimal sequence of complex and interdependent operations has always been a challenging problem for management scientists. Several real-world examples can readily be cited: Vehicle routing for the distribution of goods and services, Fisher [1985] or Glover et al. [1978], scheduling of machines for manufacturing, Guignard and Kim [1987], crew scheduling for airlines, Gershkoff [1989], operator scheduling for telephone companies and so on. Models for these applications usually fall under the realm of combinatorial optimization. This is an area of extensive research for the design of new algorithms that exploit parallel computer architectures. While parallelism does not make these models theoretically easier, it opens up new ways for the design of heuristics or the development of approximation procedures.

American Airlines has recently challenged the operations research community to attack extremely large crew scheduling problems. The availability of high-performance supercomputers has provided the catalyst for some recent work that led to the solution of multi-million variable instances of this application. Bixby et al. [1991] use a combination of interior point codes with a simplex algorithm to solve a test problem from tne American Airlines Challenge with over 12 million columns. Using a CRAY Y-MP they were able to streamline the software implementation to execute at over 200 MFLOPS computing rates. The problem was solved in a remarkable 7 minutes, using either interior point or simplex algorithms, and in approximately 4 minutes using a hybrid approach. Interestingly enough, the time needed to process the input data, remove some duplicate columns and load the data into the memory exceeds by far the solution time. However, supercomputers like the CRAY offer a solid-state storage device with extremely fast input/output channels that substantially enhance the computer's capability in handling very large data files. The integrated use of new developments in linear programming algorithms, the effective use of the vector hardware and the availability of high-speed storage devices made the solution of extremely large scheduling applications feasible.

4. Parallel Processing Applications in Air Traffic Management System

The field of Air Traffic Management is becoming extremely important today as air traffic congestion is growing all around the world. This problem has indeed existed since the early days of flight but only now, with the great increase in aircraft transportations and travel, is the problem becoming critical.

Using FAA terminology, two major components can be identified in the Air Traffic Management system:

1. Flow management
2. Air traffic control.

The former ensures optimum flow of air traffic when demand saturates the available capacity of the system. The latter provides correct and safe separations to prevent collisions between aircraft while mantaining an ordered and prompt flow of air traffic. Because safety is ensured first, irrespective of other considerations, flights are often penalized in terms of time, fuel and cost. Recently many researchers have been developing optimisation models for air traffic management in order to reduce these penalties. We point out in the next paragraphs the main aspects of these models, and discuss why and how parallel processing can help in their solution.

4.1 Flow Management

As Bianco and Bielli [1991] and Odoni and Richetta [1991] observe, flow management can be seen as a hierarchical problem in time where possible interventions can be scheduled in long, medium and short time horizons. Of course the most difficult problem is related to short term flow management. Two main types of flow control can be identified:

1. Strategic control which determines when to release an aircraft into the ATC system or, in other words, how long to delay an aircraft before take-off (*ground holding problem*).
2. Tactical control for controlling an aircraft once it is airborne. Typical control actions are holding (that is stacking aircraft at different altitudes), re-routing aircraft on a path different from the original one, and flow rate restrictions.

The latter has been widely studied, models have been proposed and related algorithms implemented. However none of the proposed models have been able to capture the multiple complexities of this problem, and tactical control is still performed "on the spot" by a traffic controller without any planning on the implications of local decisions on the global performance of the system. Zenios [1991] proposed a multi-period model for a single sector over all possible altitudes with the dual objectives of optimizing congestion and fuel cost. Results on a prototype modeling system for the Indianapolis control sector show that significant reduction of congestion is possible. Moreover it is shown that using a supercomputer, like the CRAY X-MP/24, the model is solved within a few seconds versus many hours on a VAX 11/750. This result makes possible the use of the developed models as operational planning tools under real time conditions.

Models for multisector control become much more complex. Zenios [1991] and Bianco and Bielli [1991] study different global, dynamic models that consider the flow of traffic through major airports at a nationwide level. In Zenios' model the objective is to minimize delays of departing flights while avoiding congestion at the destination airports. In Bianco's model the objective is to maximize a weighted sum of the source-generated flows and a weighted sum of the arc flows where the weights can be chosen to assign special priorities. The scale of the problem varies substantially depending on the time and space representation chosen.

Efficient algorithms for multi-sector models need greater computational power than those for a single control sector. Moreover, to have a complete practical tool, the multisector and the single sector model should interact with each other. It follows naturally that the computational technology for solving the single sector is quite different from that needed for multisector control. The former needs local powerful supercomputers able to carry out simultaneous simulations, while the latter needs a supercomputer working as a master and able to collect data from local processors spread over the different sectors and linked together in a loosely coupled way.

Returning now to the problem of strategic control, this is motivated by the need to find a trade-off between delays on the ground before take-off and delays suffered while airborne (which are much more expensive than the former). As Odoni et al. [1991] point out, the problem is intrinsically stochastic and dynamic. The probabilistic structure comes essentially from the fact that the landing capacity at each airport cannot be predicted exactly even few hours in advance, due to uncertainty both in weather conditions and in demand levels. Dynamic aspects are related to the system data involved which are characterized by continuous and fast changes.

It is well known that stochastic programming was proposed as a remedy to deterministic programming which is unable to take into account dynamic behaviour in real-time applications. Recent works show that large stochastic problems can be solved in a very efficient way compatible with real-time requirements and without losing the structure of the original problem during the solution process (Rockafeller and Wets [1987]). Dantzig [1988] discusses how parallel processing can affect the solution process of large stochastic programs. Moreover, Nielsen and Zenios [1992] showed that using a decomposition technique on the decision variables and the row-action algorithm of Censor and Lent [1981] in solving quadratic stochastic network problems is suitable for massively parallel SIMD computers like the Connection Machine CM-2.

What these researchers have collectively shown is that it is possible to solve extremely large dynamic and stochastic problems very effectively. Various forms of parallel computing technologies have provided the catalyst for these developments.

We also observe that, even if we focus on the capacity of a single airport, the number of alternatives that should be considered is very large. The problem is approached through a stochastic linear model with control on individual flights in Terrab and Odoni [1990] and with groups of aircraft in Odoni [1991]. The first method leads to a very large problem in practice while the second leads to a problem which can be solved on a single workstation. This latter has been tested on a single airport and it has been shown that a substantial reduction of delay can be achieved with respect to the current strategy.

It follows that if we want to solve the ground-holding problem on an individual flight basis we can solve it using a massively parallel approach. The problem becomes much more complex if we extend our analysis to multiple airports. However, a loosely coupled environment with massively parallel processing at each airport could potentially solve it.

4.2 Simulation

We have not yet discussed the importance of simulation studies in ATC and the impact of parallel processing on them. Simulation has been used for many different aspects of aviation system development ranging from airport studies (relocation of gates, building of new runways, etc.) to separation strategies or to training of users. Certainly the most stimulating and promising research and development efforts are in real-time simulations where on-line data and possible scenarios have to be used to obtain an advanced knowledge of what would happen if a certain event takes place. In a very advanced environment an optimization strategy could also be used to obtain the optimal use of the resource available (determination of least-fuel path, track optimization, etc.).

For example, the Operations Research Service of FAA has developed FLOWSIM, an on-line planning support tool to be used by air traffic managers to control national traffic flows. A dynamic ocean track system is also under development, able to advise each aircraft of track alternatives on the basis of actual weather conditions and with the goal of reducing the workload and improving fuel efficiency.

The FAA National Simulation Laboratory is carrying out a more demanding task, that is to simulate the interactions between different components of the aviation system. The goal is to understand how a modification in performance in a part of the system can influence the behaviour whole system. This is a challenging subject for parallel and distributed processing.

5. Conclusions

Parallel processing promises to be the dominant paradigm for large-scale computations over the next decade, and perhaps beyond. Air traffic management stands to gain substantially from the development of this technology: parallel processing promises to solve problems that were considered so far intractable. The real time air traffic management includes complex operations that play a key role for the effective use of it. With the development of parallelism this role appears to be redefined, to address more complete and ambitious instances of the application. In particular, Dantzig's old admonition to "cope with uncertainty" is well within reach. Tools like stochastic programming and robust optimization provide the framework within which we may control the impact of uncertainty, while parallel processing provides the required computing power.

We should expect to see substantially more progress in the use of parallelism in applications of air traffic management. After all, it has been less than five years since the development of the first parallel optimization algorithms, and we have already seen some very interesting applications.

References

[1] L. Bianco and M. Bielli. Air traffic management: optimization models and algorithms. Advanced Seminar on *Air Traffic Control: Data Processing and Large Scale Computation, Capri, ITALY, 1991.*

[2] L. Bianco, A. Odoni, and G. Szego. *Flow control of congested networks.* Springer Verlag, 1987.

[3] R.E. Bixby, J.W. Gregory, I.J. Lustig, R.E. Marsten, and D.F. Shanno. *Very large-scale linear programming: A case study in combining interior point and simplex methods.* Report SOR 91-08, Princeton University, May 1991.

[4] L.D. Cagan, N.S. Carriero, and S.A. Zenios. *Pricing Mortgage-Backed Securities with Network Linda.* Report 92-01-10, Decision Sciences Department, The Wharton School, University of Pennsylvania, Philadelphia, PA 19104, 1992.

[5] N. Carriero and D. Gelernter. *How to write parallel programs: A guide to the perplexed.* Technical Report, Scientific Computing Associates, New Heaven, CT, 1989.

[6] Y. Censor and S.A. Zenios. Interval-constrained matrix balancing. *Linear Algebra and its Applicalions,* 150:393-421,1991.

[7] R.J. Chen and R.R. Meyer. Parallel optimization for traffic assignment. *Mathematical Programming,* 42:327-345, 1988.

[8] Y. Deng, J. Glimm, and D.H. Sharp. Perspectives on parallel computing. *Daedalus,* 121:31-52, Winter 1992.

[9] M.L. Fisher. An applications oriented guide to lagrangian relaxation. *Interfaces,* 15:10-21, 1985.

[10] M.J. Flynn. Some computer organizations and their effectiveness. *IEEE Transactions on Computers,* C-21(9):948-960, 1972.

[11] I. Gershkoff. Optimizing flight crew scheduling. *Interfaces,* 19:29-43, 1989.

[12] F. Glover, G. Jones, D. Karney, D. Klingman, and J. Mote. An integrated production, distribution, and inventory planning system. *Interfaces,* 9:21-35, 1979.

[13] M. Guignard and S. Kim. Lagrangean decomposition for integer programming: theory and applications. *R.A.I.R.O. Recherche Operationnelle,* 21(4):307-323, 1987.

[14] J.L. Gustafson. Reevaluating Amdhl's law. *Communications of the ACM*, 31(5): 532-533, May 1988.

[15] W. D. Hillis. *The Connection Machine.* The MIT Press, Cambridge, Massachusetts, 1985.

[16] R. Hockney and C. Jeeshope. *Parallel Computers.* Adam Hilge Ltd., Bristol, England, 1981.

[17] G.A.P. Kindervater and J.K. Lenstra. An introduction to parallelism and combinatorial optimization. In J. van Leeuwen and J.K. Lenstra, editors, *Parallel computers and computations,* CWI Syllabus 9, Center for Mathematics and Computer Science, Amsterdam, The Netherlands, 1985.

[18] A.J. Kulikowski. Traffic flow management in today's ATC system. *Journal of ATC,* Jan.-March 1991.

[19] H. Mahmassani, S. Peeta, G-L. Chang, and T. Junchaya. *A Review of Dynamic Assignment and Traffic Simulation Models for ADIS/ATMS Applications.* Technical Report DTFH61-90-R00074-1, Center for Transportation Research, University of Texas at Austin, 1991.

[20] A. Nagurney, D-S. Kim, and A.G. Robinson. Serial and parallel equilibration of large-scale constrained matrix problems with application to the social and economic sciences. *International Journal of Super-computer Applications,* 4:49-71, 1990.

[21] A. Odoni and 0. Richetta. Stochastic linear programming solution to the static ground-holding policy problem in air-traffic control. Advanced Seminar on *Air-Traffic Control: Data Processing and Large Scale Computation,* Capri, ITALY, 1991.

[22] P.M. Pardalos and X. Li. Parallel branch and bound algorithms for combinatorial optimization. *Supercomputer,* 39:23 - 30, 1990.

[23] P.M. Pardalos and G.P. Rodgers. Parallel branch and bound algorithms for quadratic zero-one programs on the hypercube architecture. *Annals of Operations Research,* 22:271- 292, 1990.

[24] M.C. Pinar and S.A. Zenios. Parallel decomposition of multicommodity network flows using linear-quadratic penalty functions. *ORSA Journal on Computing,* 4, 1992.

[25] M. H. Schneider and S. A. Zenios. A comparative study of algorithms for matrix balancing. *Operations Research,* 38:439-455, 1990.

[26] G.L. Schultz and R.R. Meyer. A structured interior point method. *SIAM Journal on Optimization,* 1(4):583-602, Nov., 1991.

[27] M. Terrab and A. Odoni. *Ground holding strategies for ATC flow control.* Working Paper, Operations Research Center, MIT, 1990.

[28] J.G. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineering,* pages 325-378, 1952.

[29] S.A. Zenios. Matrix balancing on a massively parallel Connection Machine. *ORSA Journal on Computing,* 2:112-125, 1990.

[30] S.A. Zenios. Network based models for air-traffic control. *European Journal of Operational Research,* 50:166-178, 1991.

[31] S.A. Zenios and S-L. Iu. Vector and parallel computing for matrix balancing. *Annals of Operations Research,* 22:161-180, 1990.

A VISION OF AVIATION WEATHER SYSTEM TO SUPPORT AIR TRAFFIC MANAGEMENT IN THE TWENTY-FIRST CENTURY[1]

John McCarthy

National Center for Atmospheric Research (NCAR)[2]
Boulder, CO 80307

Today's aviation weather system provides imprecise information, covers huge geographic areas, and often overpredicts or entirely misses adverse weather conditions. When weather conditions are marginal or rapidly changing, the safety, efficiency, and capacity of aviation operations are compromised. The vision of the future aviation weather system presented in this paper includes a data-rich, gridded observation and forecast system that will make it possible to deal with weather situations strategically rather than reactively, as is the case today. The accurate, high-resolution weather products provided by this system will be presented to pilots, controllers, and other users in clear, simplified formats on user-friendly displays. The result will be a major enhancement of the safety, efficiency, and capacity of the National Airspace System.

1. Introduction

Aircraft and weather have been inseparably linked since the first day of powered flight at Kitty Hawk. Despite decades of technological advances, weather remains one of the most important factors affecting aviation safety and efficiency. From 1970 to 1985, for example, 40 percent of all airline accidents were weather related. Sixty-five percent of annual air traffic control system delays are attributable to weather and account for $1.7 billion of direct costs to the airline industry each year—not including the inconvenience and costs suffered by the traveling public. With aviation demand and business costs projected to double within 20 years, weather-related delays will increase unless avoidable weather-related delays are identified and weather system improvements are planned and implemented.

---

[1] An earlier version of this paper was prepared for the Federal Aviation Administration (FAA) Office of System Development.

[2] NCAR is sponsored by the National Science Foundation (NSF); principal funding for this work is provided by the FAA.

Unfortunately, the quality of aviation weather services in the United States has major deficiencies, many existing since the 1950s or up to about five years ago when selected improvements were implemented. Today's aviation weather system is characterized by observations that are too sparse, prohibiting the creation of high-quality, definitive, and reliable weather information. This insufficient or inaccurate weather information compromises safety (when pilots inadvertently get into hazardous situations) and efficiency (as they make unnecessary, costly flight path diversions). The lack of adequate predictions about wind and weather conditions that lie ahead results in reactive operational decision making by pilots and controllers.

A much-improved aviation weather system is needed to resolve hazardous and operationally significant weather and make useful products available to ground crews, flight crews, and the automation system. Groundwork has already begun, as U.S. institutions responsible for improving weather forecasting have concluded that higher resolution observations and numerical prediction computer models are necessary. For example, knowledge to detect and predict one of the severest, smallest, and most deadly weather phenomena—the microburst—was virtually unavailable a decade ago. The experimental use of integrated weather sensors, including the Terminal Doppler Weather Radar, at Denver Stapleton International Airport to detect and track microbursts has been credited with saving hundreds of airline passenger and crew lives.

By anticipating a time when improved weather sensors, processors, and much more usable products will become available, the aviation weather system infrastructure has undergone an evolution during the past decade. While significant progress has been made, the rapidly accelerating state of atmospheric science and associated technology compel a much greater emphasis in the coming years.

2. Future Vision: Objectives and Benefits

From the perspective of various aviation users, the current aviation weather system has a number of operational shortfalls. The envisioned system has several objectives that could mitigate these shortfalls. They are:

1. To improve the timeliness and accuracy of weather information so that pilots are not forced to "guess" about weather conditions and potentially compromise the safety of passengers and crew.
2. To provide controllers and traffic management with accurate weather information about current and forecasted conditions to support the controllers' responsibility to ensure pilots' ability to make timely and informed operational decisions in response to hazardous weather conditions. In addition, controllers and traffic managers will be able to do a much better job of planning and optimizing flow, as well as provide separation from known hazardous weather.
3. To "fine-tune" aviation terminal forecasts (FTs), which currently cover all possible hazards and often overforecast local weather conditions, so that only germane hazards are forecasted; in addition, an elucidation of terminal winds and hazardous areas is needed.
4. To improve winds aloft forecasts and identification of hazardous airspace, allowing for better pre-flight planning and in-flight rerouting strategies.
5. To provide the wake vortex advisory service with information about current and anticipated atmospheric conditions. Without precise knowledge of wake vortex location and movement, separation standards are arbitrarily large, thus decreasing capacity.
6. To provide pilots and controllers with easily understood weather products. Many current aviation weather products are highly coded, taxing the abilities of seasoned pilots, controllers, and meteorologists to interpret them. Significant meteorological

warnings (SIGMETs) cover areas that are typically much larger than the extent of the actual hazard; nevertheless, they must be broadcast to pilots.

7. To provide en route users with critical information about local weather conditions. Numerical models have improved forecasts in the 8, 12, 24, 36, and 48 hour forecast periods, but they do not predict rapidly changing weather conditions in the short term of several hours. (Most continental U.S. flights are completed within 2 hours.) Warning and forecast offices and centers have therefore been established by the operational arms of the weather service to provide information about local weather events such as the location, movement, and severity of thunderstorms. However, this urgently needed information may not be transmitted to the aviation user, especially to general aviation pilots, in a timely manner, an important problem of connectivity between the National Weather Service's Weather Forecast Offices and the FAA's air traffic control facilities.

It should be noted that although the aviation system operates satisfactorily about 80 percent of the time, when weather is either "ceiling and visibility unlimited" (CAVU) or "Low Instrument Flight Rules" (LIFR), it is impacted negatively 100 percent of the time by poor information about marginal and rapidly changing weather conditions. Reactive flying, increased controller workload, major air traffic control delays, increased voice congestion, and inefficient use of the airspace are symptomatic of this situation. The expected doubling of flights will compound this deficiency.

Achievement of these objectives will lead to annual safety, capacity, and efficiency benefits of approximately $4 billion, based on an assessment of weather products that could be developed between 1990 and 2000 and a description of a proposed FAA and National Weather Service aviation weather system. And the FAA has estimated enhanced weather detection and forecasting for FAA systems alone to have a benefit of $110 billion over the next 20 years.

A need or requirement for a four-dimensional weather system is based on the concept that a broad consortium of weather capabilities can produce a highly resolved weather system which includes a very high temporal structure. The air traffic control system should begin now to accommodate four-dimensionality in its advanced stages of modernization.

3. Realization of the Vision

3.1 Current and Planned Improvements

The ability to resolve weather on space/time scales of a few miles and a few minutes, necessary for aviation, has been unsuccessful due to an insufficiently dense observation network. To address this inadequacy and achieve the objectives of the future vision, several enhancements have been planned (and some have already been procured and deployed) by the FAA, the National Oceanic and Atmospheric Administration, and the Department of Defense. Doppler radars for en route and terminal areas, weather satellites, automatic surface weather observing systems, wind profilers, in situ atmospheric measurements taken by aircraft, and lightning detection systems form a groundwork for greatly improved weather resolution. In the U.S. heartland, where observations are planned to be most dense, the number of daily weather observations may increase by as much as 30 times by the year 1995 and perhaps by 50 times early in the next decade. The state of the atmosphere will be known in detail, including temperature; humidity; wind; all types of precipitation such as rain, snow, and freezing precipitation; cloud type and extent; and pressure. Aviation-state weather variables will be known similarly, including the extent of thunderstorm conditions, windshear, icing and ground icing conditions, ceiling and visibility changes, and turbulence. Figure 1 shows the improved resolutions of the observing system between 1985 and 1995.

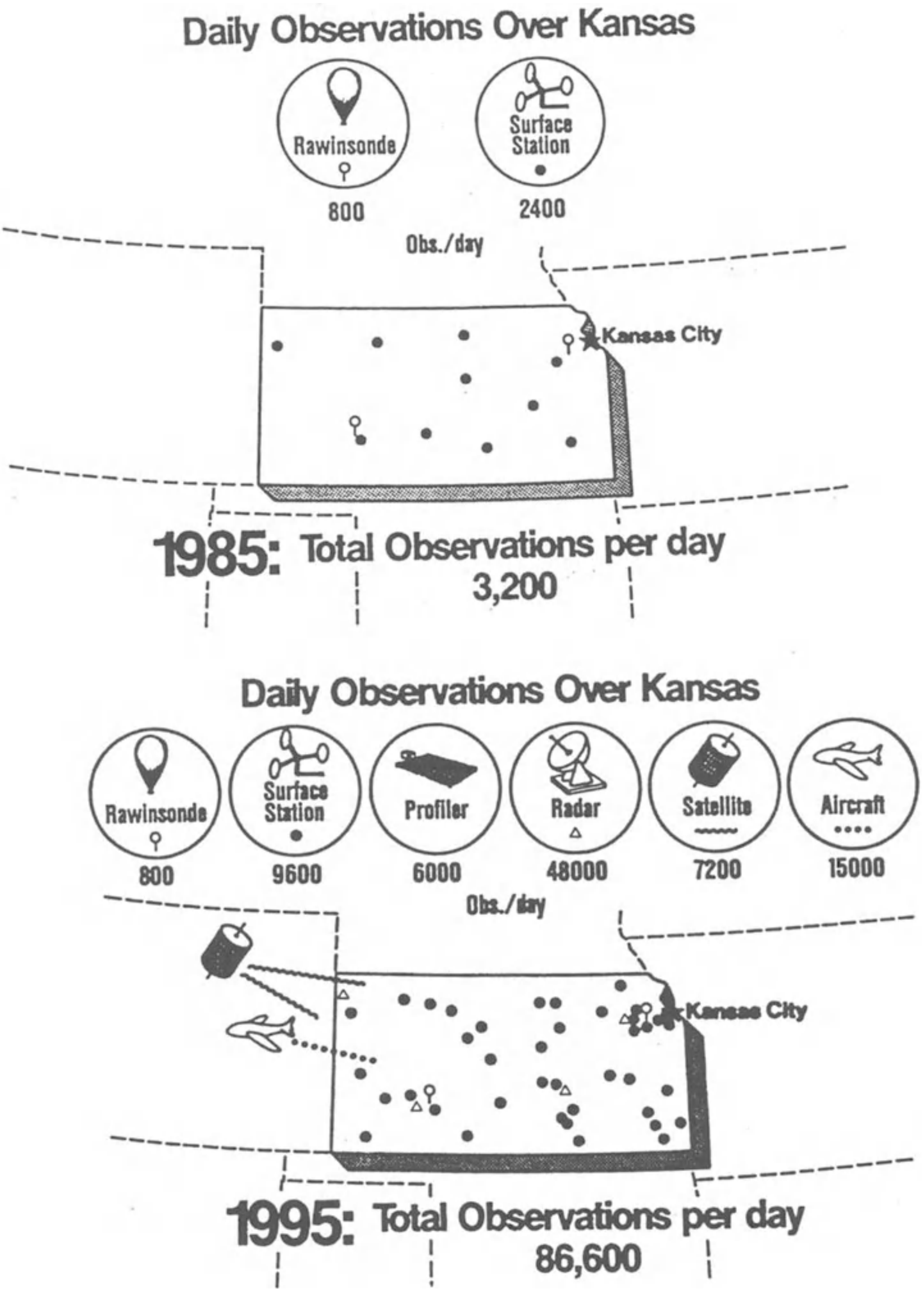


Figure 1. Instead of today's 250 mile and 12 hour resolutions, observations in the early 21st century will be characterized by 1-5 mile and 5 minute resolutions. (Source: Alexander MacDonald, NOAA/ERL/FSL)

In addition to enhanced observational systems, on going improvements in parallel-processing computers and communications technologies will allow for similarly enhanced capabilities in weather processing and dissemination. The federal agencies that focus on weather have created, or are in the process of creating, real-time weather processors that will assimilate thousands of times more weather data, and will provide those data to weather information displays that allow for a much more clear description of weather conditions. Major advances in weather forecasting models will make routine short-term prognostication of aviation weather conditions on 1, 2, and 3 hour scales. Other weather information extrapolation techniques will allow for very short term forecasting capabilities in the 1, 5, 10, 30, and 60 minute time frames, important for turbulence and windshear warnings.

An example of where this very short term capability applies is in wake vortex forecasting, detecting, and warning, included in this vision of the future aviation weather system because of the close association wake vortex detection and decay has with aviation weather sensors and certain atmospheric conditions. Needed are microwave Doppler and light detection and ranging (lidar) radar systems that can positively identify vortices, and very high resolution wind field mapping over the airport, necessary for tracking detected vortices, to create major advances in a workable wake vortex advisory service. By having an expanded capability to measure and reliably forecast the atmospheric conditions that cause wake vortices to advect and/or decay, one can provide weather-adaptive wake vortex separations. The Terminal Automation System should permit the practical and safe utilization of such separations without increasing controller workload.

Other improvements include:

- Airborne hazardous weather detection systems will provide warnings to flight crews. These warnings could then be sent automatically to other flight crews en route. In an advanced weather system, this airborne sensing capability (now in its early stage) will grow immensely to include infrared and lidar clear air turbulence sensing, Doppler radar sensing of windshear, lightning detection, and a host of aviation-state variables, including humidity, the presence of icing, and the presence of the aircraft in cloud conditions.

- An integrated terminal surveillance and weather system early in the twenty-first century could include electronically scanning Doppler radars; these advanced, multiparameter radars could provide a three-dimensional image of terminal weather hazards such as thunderstorms with update rates as frequent as 30 to 60 seconds.
- Satellite-derived soundings of the atmosphere in remote regions (e.g., use of occulting Global Positioning System satellites to derive temperature and humidity soundings) will be combined with airborne measurements, improving weather warnings and forecasts. Satellite communications, integrated with the navigation system for precise location, will provide this new information to flight decks worldwide.

To some degree, the deficiencies of the current aviation weather system have been addressed for the past decade. An extraordinary array of new weather sensors and processors, along with an improving understanding of the aviation weather science, has been evolving. Some of these new systems (e.g., Terminal Doppler Weather Radar and Low-Level Windshear Alert System) have paved the way to a future vision for an even better aviation weather system.

3.2 A New Aviation Weather System for the United States

The FAA will establish aviation weather product generators to provide useful weather information to all classes of pilots, controllers, traffic managers, and other users. This is in stark contrast to the weather data overload that is rapidly accumulating in today's weather system.

Terminal Area Airspace

The Integrated Terminal Weather System will receive gridded observation and forecast data from the National Weather Service every 5 minutes and combine them with FAA terminal sensor data (e.g., from the Terminal Doppler Weather Radar and Enhanced Low-Level Windshear Alert System). It will generate four-dimensional estimates of the current and predicted hazardous weather and distribute the information as products to pilots via data link and to flight controllers via graphical displays. Traffic managers will be able to rely on Integrated Terminal Weather System information to ensure efficient airspace operations. Air traffic management computers will use Integrated Terminal Weather System information to maximize traffic acceptance and departure rates at high-traffic airports. Programs associated with the Terminal Air Traffic Control Automation such as the Center/Terminal Automation System will be able to use high-resolution information about winds to provide four-dimensional metering, spacing, and rerouting of inbound and outbound traffic. These products will be user-friendly and not require interpretation by a meteorologist.

Pilots will have advanced onboard graphics screens that integrate weather hazard information with a four-dimensional navigation system and traffic control instructions. Highly succinct textual products also will be available. For example, the Integrated Terminal Weather System will provide simple, textual microburst alerts:

MICROBURST ALERT, EXPECT
60 KNOT LOSS ON 1 MILE FINAL

coupled with situational color displays of severe windshear hazard areas. The Terminal Doppler Weather Radar and enhanced Low-Level Windshear Alert System have this windshear warning capability, and the Integrated Terminal Weather System will expand it to a considerably broader complement of terminal area weather hazards identification. Additionally, in our view of the future aviation weather system, it will be incumbent upon the air traffic control system to provide separation from these sorts of extreme and life-threatening weather hazards.

Creative use of text and graphics will define areas to be avoided by aircraft, due either to hazards (e.g., thunderstorms) or to operationally significant weather (e.g., strong localized winds), with update rates of a few seconds to minutes. By stratifying the airspace in the terminal area into "go" and "no-go" space, airspace capacity could be increased by as much as 25 percent in many types of weather conditions and maintained when it would otherwise be lost in today's system. However, certain weather events such as blizzards will still close airports.

En Route/Regional Airspace

For the en route airspace, a Regional Aviation Weather Products Generator will be situated in the Area Control Façility. Typical resolutions of this domain will be in the 1 minute and 1 mile range, somewhat more coarse than the Integrated Terminal Weather System domain, but many times more detailed than today's system. High-resolution hazard and operationally significant weather products will be delivered to pilots via data link and to controllers and traffic managers via the Advanced Automation System. The Advanced Automation System (e.g., using Automated En Route Air Traffic Control) will ingest highly resolved weather data for planning to maximize airspace usage in the Area Control Facility airspace by employing the products of the Regional Aviation Weather Products Generator. Figure 2 illustrates high resolution icing and turbulence hazard areas, in a three-dimensional perspective, that should become available in the next several years. Figure 3 represents a view of the same hazard as seen from a modern glass cockpit.

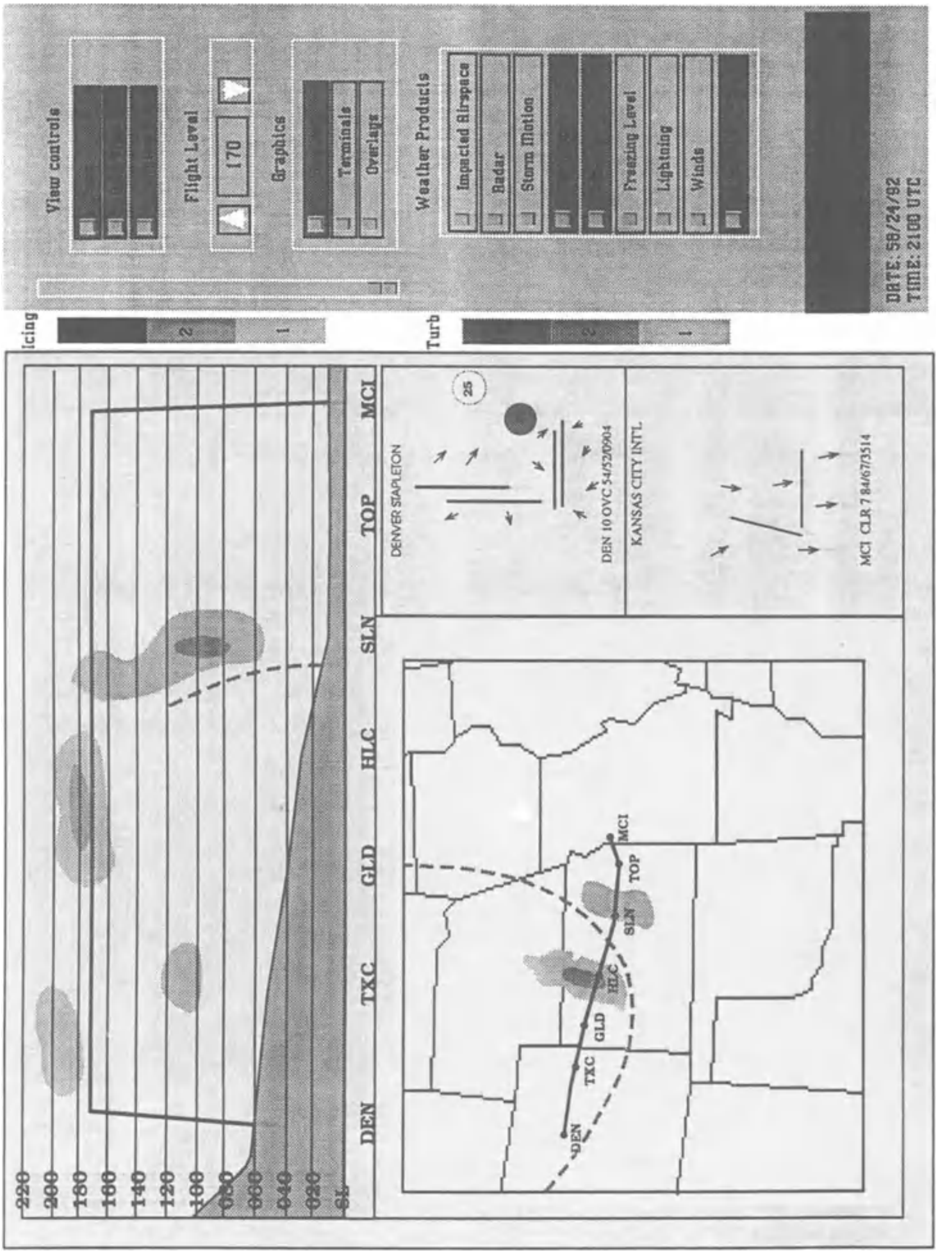
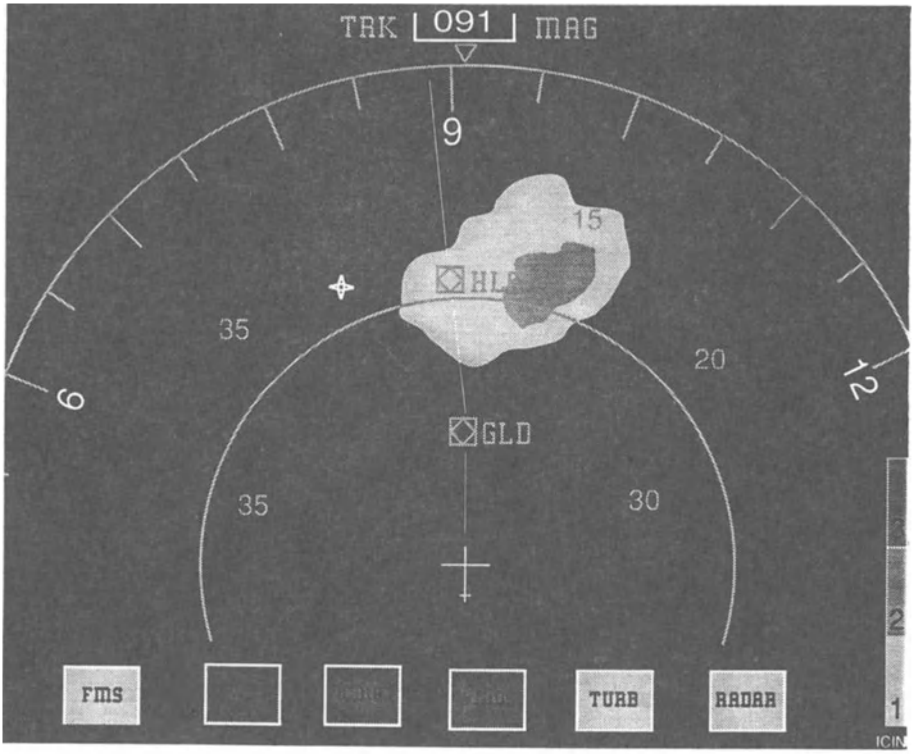


Figure 2. High resolution depiction of icing and turbulence zones along a flight track at 17,000 feet, between Denver and Kansas City.



COCKPIT DISPLAY

Figure 3. Depiction of icing zone (same zone as in Figure 2), as shown on an in-flight display typical of a modern glass cockpit. Note the ATC "automatic reroute" around the hazard.

National Airspace

The national airspace environment will also be protected by an advanced, high-resolution system. At the Central Flow Control Facility, a National Aviation Weather Products Generator will provide traffic managers and traffic management computers with a mosaic of weather hazards for the continental United States, similar to that provided by the Integrated Terminal Weather System and the Regional Aviation Weather Products Generator. The resolutions will likely be a coarser 10 minutes and 25 miles, sufficient for national traffic management purposes. Ceiling, visibility, and other acceptance restrictions will be closely monitored at hub airports on time scales of a few minutes to 6 hours, the duration of a transcontinental flight. Consequently, a one to three hour terminal forecast will be valuable to traffic management and of great interest to central flow control. With an improved wind mapping capability in the one to three hour forecast, national air traffic management of traffic flow will become significantly more efficient, a capability that will be enhanced when thunderstorm cells, snowstorms, and reduced ceiling and visibility conditions are much more succinctly known.

Figure 4 illustrates a national depiction of convective hazard airspace along a specific route of flight, while Figure 5, illustrates highly refined airspace hazards that should be avoided.

The Automated Flight Service Station will have advanced graphic and textual capabilities emanating from the Integrated Terminal Weather System, Regional Aviation Weather Products Generator, and National Aviation Weather Products Generator. A new-generation Flight Service Automation System will use products from the Regional Aviation Weather Products Generator to provide route-specific weather conditions for general aviation pilots querying the Automated Flight Service Station.

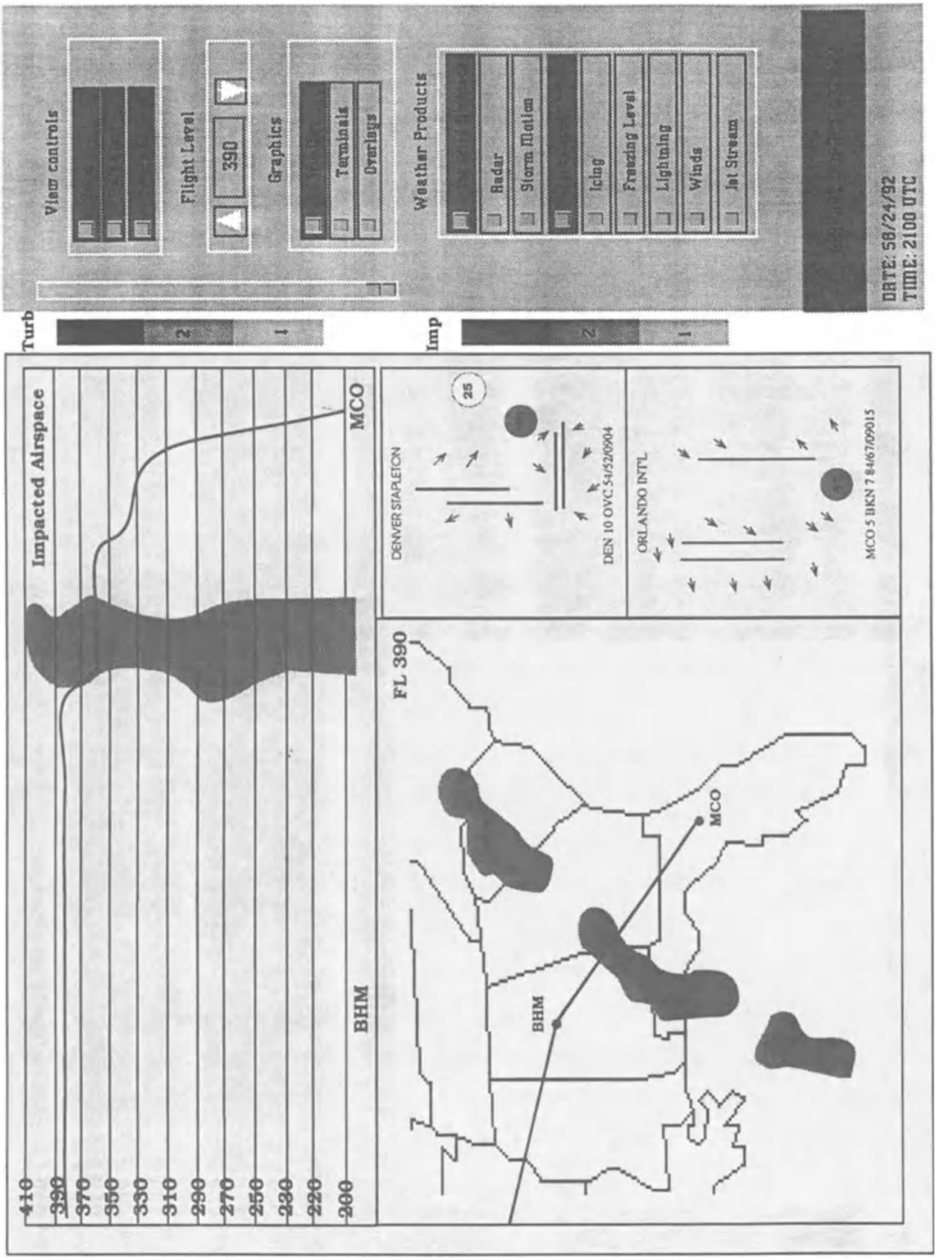


Figure 4. A national-scale display of convective storm hazard, similar to a penetration of a thunderstorm squall-line located in the Southeast U.S., for a flight from Denver to Orlando.

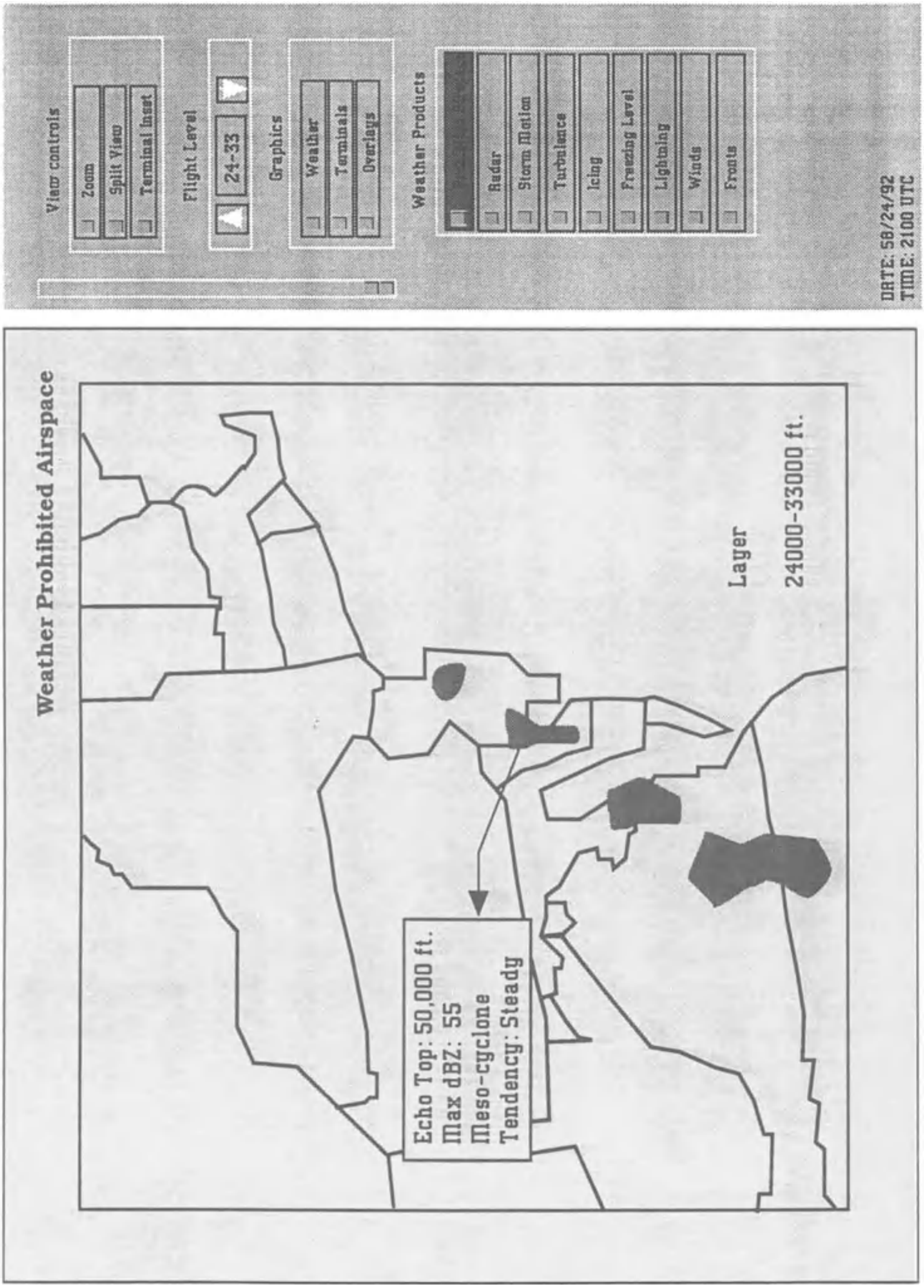


Figure 5. Illustrates the concept of weather prohibited airspace, which are zones so hazardous that no flight should penetrate except in an emergency. These composite zones are derived from a variety of sensors, rather than from any one sensor, all part of weather system modernization.

All components of the future system will have route-specific information available by computer. Only the aircraft location and intended route are needed to provide route-specific weather information via voice, data link, or satellite communications. Each flight in the continental United States will have onboard access to information about weather conditions to within a few miles of either side of the intended route. If intervening weather suggests a strategic change, alternative routes will be commonly understood by both pilot and controller (Figure 6). This is a crucial element of the envisioned aviation system. Getting route-specific weather information to pilots, particularly general aviation pilots, will provide a tremendous improvement in flight safety.

The role of weather vendor services will be greatly enhanced through satellite broadcast of Integrated Terminal Weather System, Regional Aviation Weather Products Generator, and National Aviation Weather Products Generator data bases. Sharing the improved strategic assets of the FAA's traffic management system, airline dispatchers will be able to adapt their operations to better deal with restricted weather situations.

Oceanic Airspace

The U.S. aviation weather system is important to international transoceanic flights. With continuing pressure on the capacity of the transatlantic and transpacific routes, and with the advent of satellite navigation and communications, the need to provide short-term information regarding rapid changes in adverse weather conditions en route will become critical. A transoceanic weather warning and forecast system will provide these advisories, using weather satellite monitoring and improved hemispheric forecast models. Onboard weather measurements from aircraft flying these routes will be linked via satellite to weather processors in oceanic weather warning and forecast centers that serve these routes. Oceanic weather improvements will be important to both strategic planning (e.g., better optimal routing with more closely spaced routes) and tactical

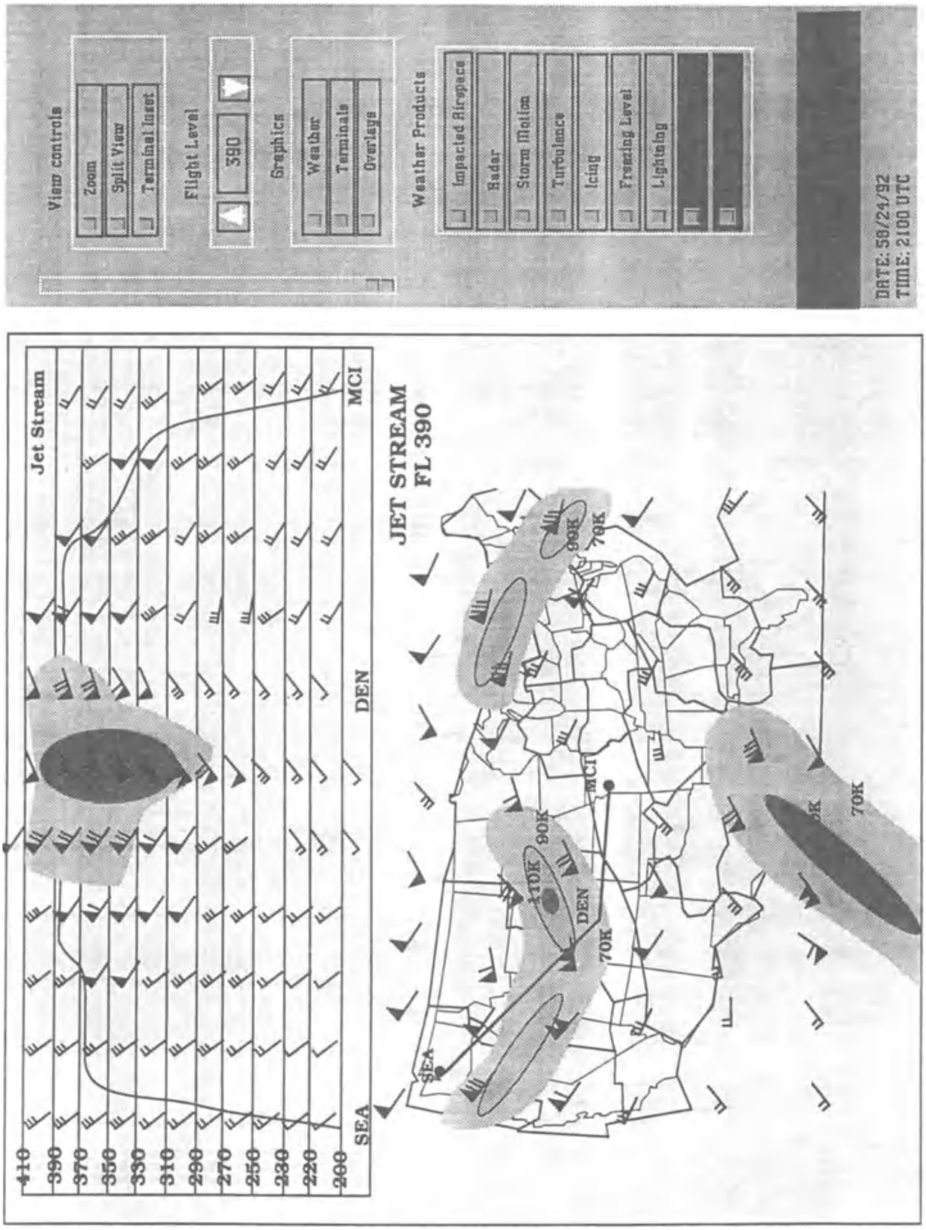


Figure 6. Depiction of winds aloft with a jet stream core analysis superimposed, along with an anticipated optimal air routing.

planning (e.g., the ability to make rapid route shifts due to turbulence, convective cells, and small-scale jet-stream features). This is in contrast to today's rigid routing and separation standards that allow little flexibility in diverse weather situations.

In the more distant future, most or all of the capabilities indicated here should become available world-wide as satellite-borne weather sensors become more prevalent, and as improved ground-based sensors become more available in countries throughout the world that are intent on improving aviation commerce. The resulting global weather system will eventually join with a global ATL system, as we achieve a global economy.

3.3 Evolving to a Viable Operations Concept for Aviation Weather

Improving weather data, developing a data-delivery system, and providing aviation users with products is the government's framework for evolving to an advanced aviation weather system. Hardware and software infrastructure and interfaces can be sketched out early in the design process so that as components of the envisioned system are defined, they can be demonstrated and implemented quickly. The envisioned future aviation weather system takes full advantage of existing FAA and National Weather Service assets. The integrated Terminal Doppler Weather Radar/Low-Level Windshear Alert System is a prime example of expected high-precision improvements, involving explicit textual and graphical user-friendly products.

Evolutionary strategy will focus on quickly developing prototypes with key related systems such as the FAA's Center/Terminal Automation System and the National Oceanic and Atmospheric Administration's Gridded Forecast System and Advanced Weather Interactive Processing System to ensure that interfaces and operations concepts are thoroughly validated. Prototypes will be demonstrated in an operational environment (test beds), and feedback from user evaluation teams will help refine them.

A flexible design will accommodate many levels of sophistication in the receiving equipment, from simple, separate weather displays intended for use by air traffic supervisors to the display of complex presentations of combined Doppler radar and weather satellite suitable for the most sophisticated graphical display system. For example, preflight weather information could be available to an airline's computer system, a distribution service (such as today's DUAT providers), a flight service specialist, the Weather Channel, or the meteorologists on AM Weather.

All aviation weather-related decisions for safe aircraft operations will continue to be retained by the pilot. As the capability of the weather system continues to evolve and short-term predictions of hazards become more accurate, routing changes will occur less tactically and be more strategic. Optimal route selection, which factors in weather and traffic information, will be determined through a cooperative decision-making process between the pilot and controller.

Human factors considerations of information system design and sensory overload must be taken seriously. In the cockpit, obtaining or assimilating weather information cannot detract from controlling the aircraft. Complex keystrokes or the necessity of orienting a graphic product to the direction of flight will be avoided. At the controller's console, weather information will require little or no interpretation and not detract from ensuring aircraft separation; it will be integrated with the controller's primary display, as airways and fixes are depicted today.

The appropriate components of the aviation weather information system will, through direct connections and via data link, provide more timely, more accurate, and higher resolution data to the future air traffic control automation, flow management, and cockpit systems. A four-dimensional navigation system onboard the aircraft could request and use information regarding winds aloft, requiring no more human effort than monitoring. A terminal automation system could estimate aircraft arrival over the arrival gate by taking into account a severe thunderstorm ("no-go" space) that will force rerouting. A similar en route automation system could provide automatic rerouting around hazardous weather areas, with much greater strategic planning than is possible today.

In light of these significant operational and technological advancements, a major effort to re-educate all users of the airspace system will be necessary. Pilots and controllers nearing retirement today have spent their entire careers learning to cope with many of the same weather products that were available when they began working in the 1950s. Our most competent flight professionals take the current, imprecise forecasts with a grain of salt, while worrying about liability when forecasts show a chance of hazardous weather in nearly every prediction.

There are immediate, critical tasks before us. A description of the current state of the atmosphere is an unmet need today. We need to describe the state of the atmosphere with greater clarity, even without the new sensors, processing, automation, and communications envisaged early in the next century. The evolution of the aviation weather system to a truly four-dimensional weather system can be accelerated by developing a capability to digitize what today is essentially an analog system. By so doing, we can maximize current capabilities, making the baselined National Airspace System more effective, meeting more immediate weather needs of the users, and positioning the current aviation weather system to phase into the future weather vision with greater success.

4. Conclusion

Today's 30-year-old marginal aviation weather system operates well only when weather conditions do not strain the safety, efficiency, and capacity of the current airspace system. With the anticipated improvements, it will evolve into a weather system that, in concert with air traffic control modernization at the turn of the century, will provide accurate, high-resolution aviation weather products. Weather-related capacity limitations that restrict operations and cost billions of dollars each year will be mitigated. Pilots, controllers, and traffic managers will be able to use these new weather capabilities much more effectively than today's limited weather information products. With an open-ended design and development strategy, improvements will be integrated into the system in a smooth, continuous process.

Users will be able to fly preferred trajectories. The situational weather awareness of pilots and controllers will be enhanced. Users will know where hazards will be located, minutes to hours in advance, allowing for strategic planning, with air traffic control's assistance, to avoid these areas. Traffic routes will be adjusted perhaps hundreds of miles before encountering hazardous weather, providing for a smooth circumnavigation. Voice congestion on air traffic control frequencies will decrease. Some reduction in large fuel reserves will be realized with more accurate weather forecasts. Airspace over the oceans will have a much-increased capacity as improved tactical and strategic planning become possible with improved oceanic weather information. Most importantly, weather factors that currently limit aircraft operations will be predicted with much greater accuracy, resulting in greater safety, efficiency, and capacity of the National Airspace System.

Acknowledgements

This paper was edited by Ms. Debbie Davis and Ms. Rose Lundeen of NCAR. In the course of its development, major input was received from Dr. James Evans of Massachusetts Institute of Technology, and Mr. Ken Byram, Mr. Anthony Broderick and Mr Marty Pozesky of the Federal Aviation Administration (FAA). The work was funded by Interagency Agreement No. DTFA01-91-Z-02039 between the FAA and NSF.

SYSTEM ASPECTS AND OPTIMIZATION MODELS IN ATC PLANNING

Lucio Bianco

Professor of Operations Research, Tor Vergata University of Rome, and Director of Istituto di Analisi dei Sistemi e Informatica & Progetto Finalizzato Trasporti, National Research Council, Rome, Italy

Maurizio Bielli

Senior Researcher of Istituto di Analisi dei Sistemi e Informatica & Progetto Finalizzato Trasporti, National Research Council, Rome, Italy

The current air traffic control systems are mainly conceived to ensure, with tactical interventions, the safety of flights. This approach often suffers due to the difficulty of accurately foreseeing the future traffic evolution, so that planes are often penalized more than necessary with the real traffic conditions. In fact, in such a situation safety standards are redundant and the single airplane can be forced to fly non optimal routes and unnecessary holding patterns.
Today, this management philosophy is no longer valid because of congestion phenomena which often occur in the most important terminal areas. Therefore, it seems necessary to introduce in the future control systems not only more automated procedures to keep adequate safety levels, but also planning functions to increase the system capacity and reduce the total cost; in this manner, it is possible to improve system efficiency.
In recent years several studies have been carried out in these two directions so that new control concepts and some optimization models and algorithms have been developed.
In this paper a survey of our earlier work in this field is provided. In particular a multilevel model of air traffic control is proposed and discussed. Then, the functions corresponding to on-line control, that is flow control, strategic control of flights and aircraft sequencing in a terminal area, are examined and the corresponding optimization models and solution algorithms are illustrated. Finally, some relevant problems discussed in recent research are mentioned and new trends are indicated.

1. Introduction

The current Air Traffic Control (ATC) systems are mainly conceived to ensure safety of flights and orderly operations, according to procedures and international rules. So, they are essentially tactical ground-based systems operated by human controllers, performing manually many relevant functions with short term interventions.

The effectiveness of this approach is diminished by the difficulty of accurately foreseeing the dynamic evolution of traffic on the airspace network, in face of aleatory events, such as weather, airport congestion, accidents or failures, etc.

As a consequence, safety standards are often redundant and planes are often penalized more than necessary with the real traffic conditions, because the single aircraft can be forced to fly non optimal routes and unnecessary holding patterns.

Today, this approach to air traffic management is no longer valid because of increasing travel demand, deregulation policy and congestion phenomena which often occur in the most important terminal areas. Therefore, it seems necessary and practicable to introduce in the future control systems not only more automated procedures to maintain adequate safety levels, but also planning functions to increase system capacity and to improve system efficiency, with a reduction of the total cost.

In recent years, several scientists have carried out studies in these two directions and developed new control concepts and some optimization models and algorithms.

In this paper, we firstly deal with the system aspects of ATC and propose and discuss a multilevel model architecture. Then, the single functions corresponding to on-line control, that is flow control, strategic control of flights and aircraft sequencing in the terminal area are examined and the optimization models and solution algorithms are illustrated.

Finally, trends in ATC development, based on the introduction of new technologies, and recent research directions are outlined.

2. A multilevel model of air traffic control

The air traffic control problem is a typically large-scale problem characterized by:

- high number of variables and constraints;
- numerous sub-systems and strong interactions;
- numerous control objectives also conflicting;
- limitations and complexity of models used to forecast the movement of airplanes and traffic evolution;
- fast dynamics and real-time interventions;
- presence of several human operators in the system (pilots, controllers, etc.), each having a certain independence on taking decisions.

To approach this problem it seems right to resort to the possibility of decomposing the overall problem into sub-problems solvable in a simpler manner (Mesarovic and others, 1970; Wismor, 1971). Typically, each control function can be decomposed into different levels; each level deals with a specific problem that can be solved by known methods and techniques. Moreover, each level can be viewed as part of a multilevel system of the global process.

In the ATC field an appropriate decomposition criterion is based on a hierarchy of control functions, each related to a different time horizon.

Figure 1 represents a possible scheme where both the available resources and long term air traffic demand are supposed to be known.

The highest level is associated with the planning of the air space structure, routes and ATC procedures and generally with the evolution of the overall control system. The second level represents the planning activity of flights carried out within a time horizon of some months in connection with the estimated traffic demand. The level called "strategic control" represents a planning activity of medium-long term interventions to organize traffic flows and/or define amendments to single flight plans. Finally, tactical control is a real-

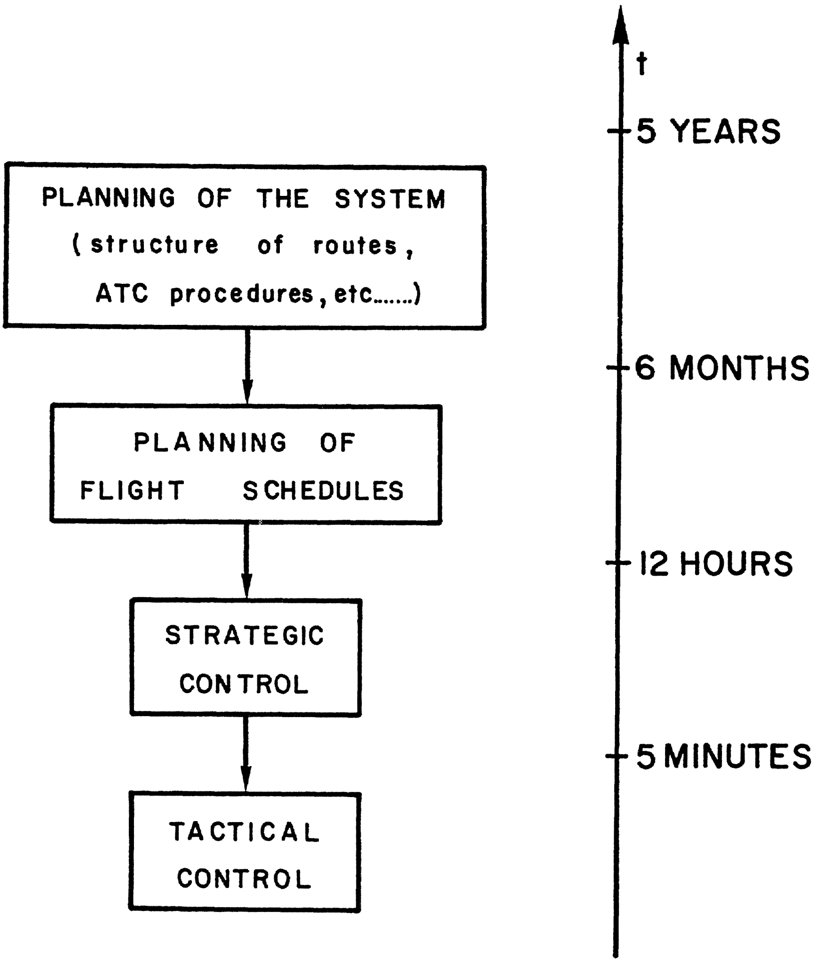


Figure 1. A multilevel model of air traffic control

time control action to satisfy short-term requirements and/or to solve emergency situations.

At present, the actions relative to the first two levels are not being carried out on the basis of optimization criteria. For example, in general, there is no coordination in the decision of the airlines; then, it is not a priori possible to prevent the congestion situations.

Available studies refer especially to the two lowest levels of the mentioned hierarchy or, more precisely, to sub-functions of these levels. In particular, strategic control, up to now, has only been utilized in a partial and limited manner; in fact, the current ATC systems are still mainly based on tactical control actions. On the contrary, we feel that a systematic introduction of a strategic function could constitute the most relevant and revolutionary innovation in air traffic control. In fact, this approach seems to be, in theory, the only one able to minimize the operational cost and, at the same time, improve safety standards in ATC (Erwin 1974, 1975). In practice, the implementation of a control function that optimizes planning of flights implies considerable difficulties due to the necessity of correct long-term forecasts and the inadequacy of available methodologies to solve problems of such complexity. Therefore, it seems convenient to decompose the strategic control function, in its turn, on the basis of a multilevel model of distinct sub-functions.

A first decomposition criterion could be a distinction between "off-line" and "on-line" control. By off-line control, we mean a planning activity carried out before interested aircraft enter the system (generally, before the aircraft departures) and based exclusively on traffic forecasts. By on-line control, we mean a control activity based mainly on observations of the current state of the system and performed in a rather short period (in comparison with dynamics of flights) to allow subsequent interventions on the controlled traffic.

A second criterion could be based on a distinction between control on aggregate variables (traffic flows) and control on variables relative to single planes (flight plans).

Figure 2 represents a multilevel decomposition of strategic control corresponding to the above criteria. To completely understand the meaning of this model, one must point out that the hierarchical order of the various levels refers mainly to the different time horizon and generally does not express a decisional hierarchy in the control actions. More precisely, we can say that higher levels basically ought to simplify decisional problems relative to the lower levels. In the following, functions associated with each level and set of operations are illustrated.

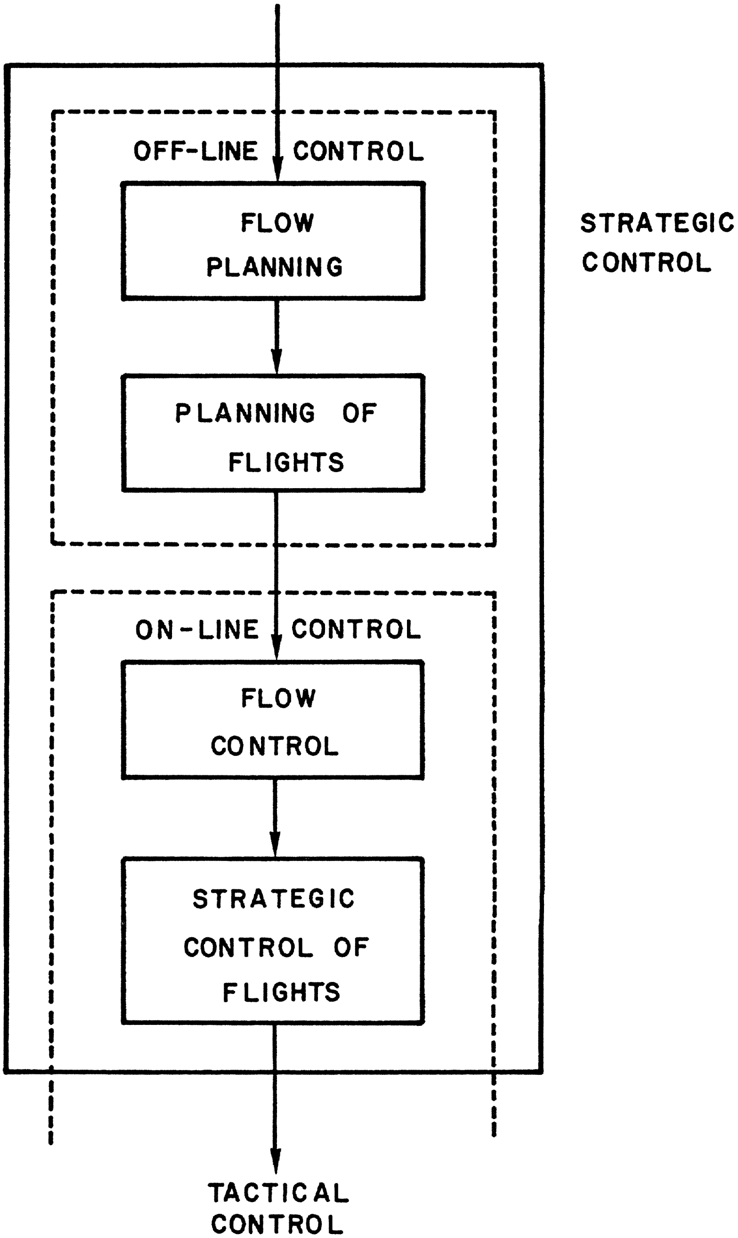


Figure 2. A multilevel model of strategic control

Flow Planning

The flow planning function should be carried out with a horizon of a few hours, comparing the expected traffic demand with an estimate of the system capacity (airways, terminal areas). Then, a distribution of traffic flows in the airspace should be determined to relate the demand to the capacity.

Planning of Flights

On the basis of flow planning and of the same time horizon, this off-line activity should determine amendments for the requested flight plans. These amendments should have the aim both of reducing the a priori conflict probability and satisfying a somewhat optimal criterion. We must also specify that many random factors affect the above process; so we represent the flights by utilizing a simplified model made up, for example, of passing times and altitudes at the way-points of the planned route.

Flow Control

The on-line flow control refers to a time horizon of about 15-30 minutes and, in particular, it should forecast local and short-period traffic peaks and reduce possible effects on the system. This can occur both by imposing delays on the flows upstream of traffic congestion and planning a different distribution of traffic in adjacent sectors.

Strategic Control of Flights (SCF)

This function has the same time horizon as the on-line flow control. It should plan conflict-free trajectories on the basis of observation of the current state of the system in order to satisfy the constraints on airplane performance, space structures and possible flow limitations.

Once a hierarchical structure of an ATC system has been defined, it seems worthwile to develop specific mathematical models to support the optimization of semi/fully automatic air traffic control and management systems at different levels of intervention.

In the following sections we consider in detail only those functions related to the on-line ATC problem.

3. Flow control

Whenever traffic congestion is present or is likely to originate in the controlled area, a flow control action must be taken. This means, essentially, matching traffic demand with capacity of the ATC system facilities.

The flow control activity can be, conceptually, decomposed into two different phases:

1) Congestion forecast: This refers to the evaluation of where and when congestion is likely to arise and the corresponding order of magnitude of the overload.
2) Congestion prevention: Whenever an overload is forecast, a control action, possibly optimal, must follow so as to prevent congestion development.

The first phase is the basis for an efficient ATC system; in particular, it requires accurate evaluation of air traffic capacity and demand. These two characteristics show random variations with space and time. Capacity varies for several reasons: (a) airport and airspace structure, communication, control and aid to navigation facilities used; (b) workload of controllers; (c) weather; (d) special events: incorrect functioning or out-of-order of control facilities, etc.

The traffic demand depends on time-varying characteristics of air transport, on delays etc.

Once traffic demand and capacity associated to the controlled region have been fully evaluated, congestion prevention can be taken into consideration. It can be performed by means of certain typical control actions: take-off delays, holdings, re-routing and upstream flow restrictions. Take-off delays may only be imposed on the airports located within the controlled region. Holding is used whenever an aircraft cannot initiate its landing procedure, because of airport congestion. In this case, the aircraft is instructed to join a waiting line which is called a "holding stack" because aircraft are "stacked" at different altitude levels above a check point, called "feeder" fix point. The re-routing action involves a flow reassignment on a path different from the normal one, constrained to the same destination point. Upstream flow restrictions consist of limitations imposed on the "flow rate" at the boundary points of the region under control.

In this paper, attention is primarily devoted to the prevention of congestion. As a consequence, a constrained optimization model based on an airspace and traffic model can be established so as to perform the control actions, aforementioned, in an efficient way. The objective is that of minimizing an overall cost function, respectively fuel consumption and/or total delay in the system, taking into account capacity and safety constraints.

4. Airspace and Traffic Flow Optimization Model

The airspace under control is defined by a structure of a given set of standard routes and waypoints, corresponding to navigational aids and check point locations, and by the associated flights procedures. In modelling it, one must take into account the subdivision of the airspace into several control sectors and, in general, the associated rules of operation (Ratcliffe, 1974).

The airspace structure can be represented in a straightforward way as a directed network of arcs and nodes. The nodes are of the following types:

- Source nodes, representing terminal areas, generating departing traffic, or boundary points of the airspace where traffic arriving from outer regions is "generated";

- Intermediate nodes, representing route intersections or given check points;
- Sink nodes, representing points of arrival into terminal areas, or boundary points where traffic "disappears" into outer regions.

The arcs represent route segments connecting a pair of nodes. Furthermore, different arcs can be associated with different groups of altitude flight levels along the same route. The control sectors, which are parts of the airspace controlled by one controller team, are represented by non-overlapping subnetworks connected by nodes representing the points of transfer of control from one sector to another.

Finally, as a controlled region is, in general, affected by different kinds of traffic (departure, arrival, overflight) and by different aircraft speed classes, they can be modeled by different commodities, each one characterized by its origin-destination pair. In general, when dealing with traffic congestion problems, it may be necessary to consider time-varying flows on the network. So, a dynamic traffic model can give a more precise description of the real environment (Bianco and others, 1980; Bielli and others, 1980). In this case, one needs to specify arc-transit times so as to represent dynamic evolution of traffic patterns entering the network. The planning horizon T is subdivided into elemental time intervals of suitable length δ, so as to establish a discrete-time model.

Then, the following assumptions have been made:

1) All the variables have been discretized (that is, they are assumed not to change appreciably within the elemental time period δ fixed),
2) To each commodity-flow on each arc, an average transit time is assigned (the greater the mix of aircraft speed and trajectories considered, the less this assumption is valid).
3) Each average transit time τ_{ij}^{k} is chosen as a multiple integer of δ.

Network flow equations assume an orderly and regular flow. To take into consideration any congestion situation one has to consider some waiting lines, according to the procedures actually employed by ATC controllers. These waiting lines are modelled as arcs closed on the nodes that represent the associated check points.

Once the network $G(N,A)$ and the set of commodities are defined, the optimal control strategy to solve traffic congestion can be obtained by setting up a constrained optimization model. It has been defined in the following way:

$$\max \sum_{t=1}^{T/\delta} \sum_{k=1}^{K} \left[c_s^k D^k(t) - \sum_{(i,j)\in A} c_{ij}^k x_{ij}^k(t) \right] \tag{1}$$

subject to:

$$\sum_{j\in A(i)} x_{ij}^k(t) - \sum_{j\in B(i)} x_{ji}^k(t - \tau_{ji}^k) = \begin{cases} D^k(t), & i = s^k \\ -E^k(t), & i = p^k \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$t = 1, \ldots, T/\delta;\ i \in N;\ k = 1, \ldots, K$$

$$\sum_{k=1}^{K} x_{ij}^k(t) \le h_{ij}(t), \quad t = 1, \ldots, T/\delta;\ (i,j)\in A \tag{3}$$

$$\sum_{t=1}^{T/\delta} \sum_{(i,j)\in S_r} \sum_{k=1}^{k} x_{ij}^k(t) \le h_r, \quad r = 1, \ldots, R \tag{4}$$

$$0 \le \sum_{q=1}^{t} D^k(q) \le \sum_{q=1}^{t} D_o^k(q), \quad t = 1, \ldots, T/\delta; \quad k = 1, \ldots, K \tag{5}$$

$$E^k(t) \le E_o^k, \quad t = 1, \ldots, T/\delta; \quad k = 1, \ldots, K \tag{6}$$

$$\sum_{(i,j)\in A} x_{ij}^k(0) + \sum_{t=1}^{T/\delta} D^k(t) = \sum_{(i,j)\in A} x_{ij}^k(\mathrm{T}) + \sum_{t=1}^{T/\delta} E^k(t) = \quad k = 1, \ldots, K \tag{7}$$

$$x_{ij}^k(t) \ge 0 \text{ and integer}, \quad t = 1, \ldots, T/\delta;\ (i,j)\in A;\ k = 1, \ldots, K \tag{8}$$

where

A = set of arcs; $A = \{1, \ldots, m\}$

$A(i)$ = set of nodes j "after" node i; $A(i) = \{j \in N | (i,j) \in A\}$

$B(i)$	= set of nodes j "before" node i; $B(i) = \{j \in N \mid (j,i) \in A\}$
c_s^k	= cost associated to the flow generated by the source s^k
c_{ij}^k	= cost associated to the k-commodity flow on arc (i,j)
$D_o^k(t)$	= k-commodity forecasted flow in the time interval t
$D^k(t)$	= feasible k-commodity flow in the time interval t
$E_o^k(t)$	= k-sink capacity in the time interval t
$E^k(t)$	= k-sink flow in the time interval t
$h_{ij}(t)$	= arc (i,j) capacity in the time interval t
h_r	= r-sector capacity
K	= set of commodities; $K = (1,..., K)$
N	= set of nodes; $N = (1,..., n)$
p^k	= k-commodity sink
R	= number of sectors
S_r	= r-sector
s^k	= k-commodity source
T	= control time interval
$x_{ij}^k(t)$	= k-commodity flow on arc (i,j) in the time interval t
δ	= elemental time interval
τ_{ij}^k	= k-commnodity arc-flow average transit time on arc (i, j), multiple of δ.

The objective function is made up of two terms: a weighted sum of the source-generated flows and a weighted sum of the arc flows. The first term involves maximization of the source-generated flows, so as to better match traffic demand with system capacity. Different weight coefficients can be assigned to different commodities, according to the priorities established. The second term concerns a better management in the traffic assignment on the network. In this way, weight coefficients can be assigned to different arcs, so that they can represent priorities in selecting the routes and flight levels or they can be directly associated to delays and/or fuel consumption, relative to different paths. Hence, the system can analyse different objectives, i.e. maximal flow, minimum cost rerouting and holding delays, etc.

As regards to constraints, equations (2) represent flow continuity at the various nodes. Since, for safety reasons, the aircraft have to maintain given in-trail standard separations, a maximum rate of flow on each arc can never be exceeded. This is expreased by arc capacity constraints (3). In each control sector, a maximum controller workload is defined. Here, it is approximated by the maximum number of aircraft in the sector during the period T. This is represented by sector capacity constraints (4). Upstream flow restrictions or source restrictions are given by constraints (5). The limitation both on airport acceptance rates and on overflight traffic to outer regions is taken into account by sink capacity constraints (6). Total flow conservation on the network, over the period T, is espressed by equations (7). Finally, the natural nonnegativity and integrality constraints (8) are to be included.

The proposed mathematical model has been assumed to be linear. As a matter of fact, the relationship between cost coefficients and flows and the controllers' workload (over a period T) are not exactly known and there is a need for further research in this field. However, since the purpose of this study is mainly to establish a suitable basis for working out more refined flow control models, the aforementioned hypotheses are acceptable. The dynamic problem (1)-(8) can be changed into a static one, by considering the network replicated in a multiperiod context (Fig. 3), i.e. each arc is repeated according to the number of time periods selected (Bielli and others, 1982). It becomes an equivalent linear static problem, belonging to the class of multicommodity network flow programming.

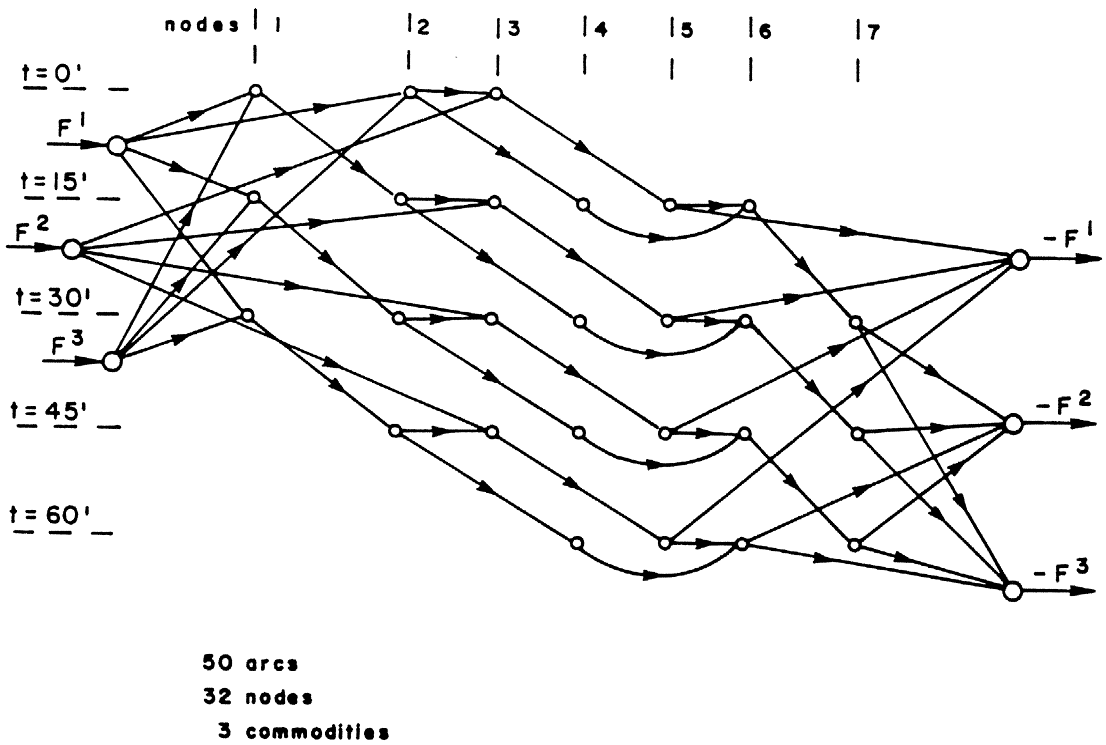


Figure 3. Example of time-space replicated network

5. Decomposition techniques and computation algorithms

The resolution of the problem can be more or less complex, depending on the accuracy required to represent the airspace structure and dynamic traffic evolution. If the controlled region is large and includes several airports and a complex route and flight levels structure, one has to deal with a network with a large number of arcs, nodes and commodities. Solving such a large-scale problem can become a heavy task if detailed airspace and traffic representation over space and time is required. In this case, simulation models appear to be a more suitable tool, especially for traffic analysis and management problems (EUROCONTROL, 1977). On the other hand, if a certain level of aggregation in the flows and airspace structure is acceptable, then the problem can be solved by suitable mathematical programming decomposition techniques. For instance, from a general point of view, resource-directive decomposition methods (such as the one developed by Assad, 1976) look promising, in spite of some computation difficulties.

This method is based on the distribution of arc capacities among the single commodities. Once the resulting one commodity ("local") subproblems are solved, the initial ("master") problem can also be considered solved, if the aggregate solution satisfies a given optimality criterion. Otherwise, the arc capacities are redistributed and the process repeated. To redistribute the arcs capacities one can apply in a suitable way an optimization method, based on the subgradient algorithm. The advantage of this approach lies in the fact that many efficient algorithms are available for solving one-commodity problems like the out-of-kilter algorithm. However, the "bundle" constraints (4) are peculiar to the problem considered in this paper.

Thus, a practical approach, suitable for implementing an operative flow control technique, is presented (Bielli and others, 1981). From an operative point of view, real-time flow control is aimed to cope with traffic peaks exceeding an acceptable demand level by optimally re-distributing the traffic flows over the network.

Moreover, the capacity constraints to be absolutely respected are, generally, the sector workloads and only a few arc capacity constraints (stacks of aircraft waiting for landing over the airports, departure and landing acceptance rates).

The former constraints measure the total flow over all arcs of the sector that can be handled by the associated team of controllers. Let us remark that these constraints are more restrictive, in the sense that they become binding before all individual arcs of the sector are full to capacity. Thus, the problem can be seen as one of choosing how to redistribute this workload among the arcs of the sector to optimize the objective function. Then, the approach can be summarized as follows:

(1) Smoothen the traffic peaks and then set fixed capacity values on some arcs and an initial capacity distribution h_0 on all other arcs. This initial distribution satisfies the sector workload constraints and insures a starting flexible flow on the network.
(2) Find capacity increments on the latter arcs satisfying workload constraints and optimizing the objective function.

If, instead, one is dealing with strategic flow control, only aggregated flows between sectors, rather than flows on specific route legs, need to be considered. In this way, the route network can be suitably aggregated: the constraints on the controllers' workload become upper bounds on the capacity of the arcs connecting two sectors. The integrality constraints can also be relaxed, considering only continuous flows, if the traffic volume is sufficiently large. Under the preceding hypotheses, the computational difficulties of the problem are greatly reduced (Bielli and others, 1981).

Figure 3 represents the space-time network resulting from the replication of the basic structure in a one-hour control interval. The corresponding discretization time is 15 minutes. Three super-sources and three super-sinks have been added, with dummy arcs having a capacity corresponding to the effective air traffic demand entering the sectors, the maximal value of the traffic exchangeable between sectors or the maximal airport landing and take-off rates.

Three possible cases have been considered: max flow (case Cl), max flow and min cost with costs $c_s^k >> c_{ij}^k$ (case C2) and max flow and min cost with costs $c_s^k = c_{ij}^k$ (case C3). The results for the commodities flows are shown in Table 1.

The actual forecasted traffic demands F_o^k on the network are larger than the sectors capacity (traffic congestion). As a consequence, the max flow is lower than demand by 12 aircraft units. These aircraft must be delayed outside the region under control.

When one considers a cost for the rerouting of the third commodity, that commodity is delayed while the other two are given priority. In the case in which the rerouting and delay costs of the demand F_o^3 are equal, the corresponding optimal flow is further decreased similarly to the global flow transiting through the network.

As far as the computational aspects are concerned, Table 2 reports the basic parameters for evaluating the performance of the suggested algorithm.

The stepsize is controlled in an adaptive way (as shown by the halving of λ). The solution is within a small percentage of the upper bound $\hat{v}$ (computed with the optimal flows obtained in the case that each commodity may use all the available resources of arc capacity). As a consequence, the distance of the solution from the true optimum is probably even smaller.

The number of iterations sufficient to obtain an acceptable sub-optimal solution is small and the CPU times on a Univac 1100 computer are also small. As a consequence, the algorithm is suitable to also deal with large-scale networks in a real-time air traffic flow control environment. These results confirm the conclusions of Ali and others (1980) that, for this class of multicommodity network flow algorithms, the resource-directive decomposition code is the fastest available.

Table 1. Results relative to optimal flows

	Commodities 1	2	3	Total flow
Actual air traffic demand	38.00	32.00	62.00	132.00
Case C1	33.90	27.48	59.01	120.39
Case C2	35.16	28.38	55.91	119.45
Case C3	35.50	28.50	51.00	115.00

Table 2. Computation algorithm results

Iterations	Case C1 $\frac{\hat{v}-v}{\hat{v}}$ %	λ	Case C2 $\frac{\hat{v}-v}{\hat{v}}$ %	λ	Case C3 $\frac{\hat{v}-v}{\hat{v}}$ %	λ
0	7.53	2	7.50	2	2.63	2
1	7.21	1	7.23	1	2.58	1
2	7.08	1	7.06	0.5	2.52	0.5
3	7.02	0.5	7.02	0.25	2.50	0.25
4	6.95	0.25	6.98	0.25	2.49	0.125
5	6.79	0.25	6.92	0.25	- - - - - -	- - - - - -
6	6.75	0.25	6.84	0.125		
7	6.71	0.125	- - - - - -	- - - - - -		
CPU =	1.592 sec.;		1.673 sec. ;		0.328 sec.	

6. Strategic control of flights

As previously mentioned, the task of on-line SCF is to optimize flight plans of single airplanes in a prefixed region with a time horizon of about 15-30 minutes. The sub-system associated with the SCF functions can be outlined as in Figure 4 where both inputs and outputs are indicated. The inputs are the flight plans of the airlines, the information relative to the state of the system and the limitations imposed by flow control.

The outputs are the amendments to the flight plans that represent control interventions on the planes of the system and can be useful information for tactical control.

In more detail, the requested flight plans should be optimal from the point of view of fuel and/or time consumption if each plane were alone in the system. Information available on the state of the system concerns radar data on the evolution of flights, short and medium term meteorological forecasts and the status of the ATC infrastructure. Restrictions imposed by flow control limit the control decision for any single plane according to the flow planning in the control region. In particular, such limitations can refer to the maintaining of appropriate separations when the planes fly over prefixed points of the controlled airspace. Finally, for amendments to flight plans we consider:

- variations of altitude levels in comparison with the nominal flight plans;
- imposing route delays through speed control;
- imposing reroutings and/or holding patterns;
- imposing departure delays.

Now, to build a mathematical model of the SCF, we must first establish a control region, a time horizon and a representation of the airways network and the trajectories of planes; then we must define control variables, constraints and the objective function to optimize.

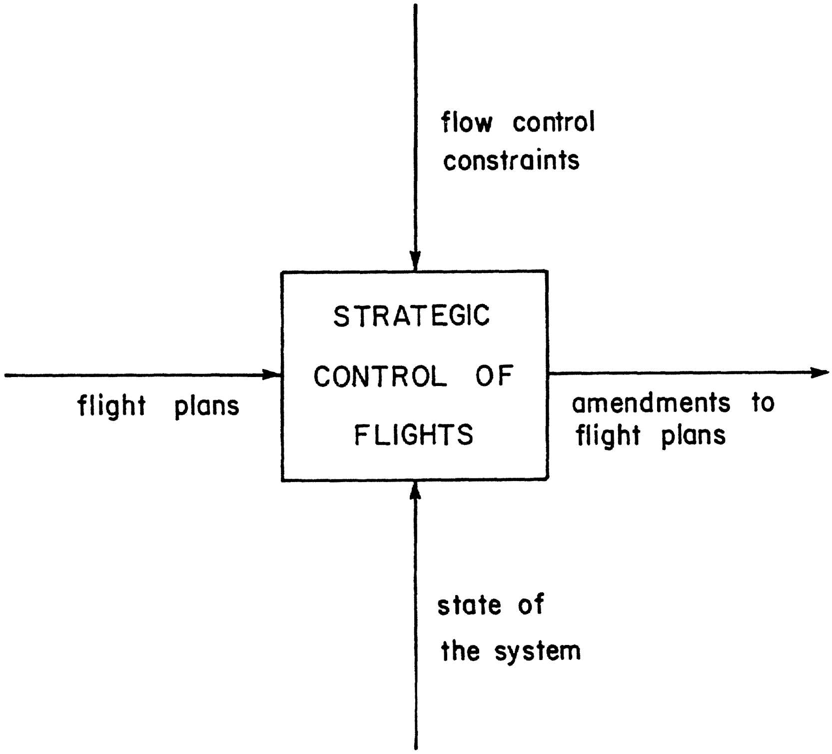


Figure 4. Representation of the SCF function

Control Region

The feasibility of carrying on a strategic action at the single plane level depends greatly on the size of the region where a coordinating intervention is possible. In fact, SCF generally aims to specify an efficient profile during all phases of the flight (take-off, cruise, landing). Therefore, the more the flight plans which refer to the prefixed control region, the more significant SCF is. Nevertheless, even without considering political, normative and organizational difficulties, an upper bound to the size of the control region exists because of limits in the performance both of mathematical methodologies and real time computation systems. In fact, the problem complexity increases as the dimensions of the control region increase.

Time Horizon

The time horizon is mainly determined by the maximum time interval in which it is possible to predict accurately future traffic conditions in the control region. At present, the available models seem to be able to provide adequate forecasts for look-ahead times of about 15-30 minutes. This limit could also assist in a spatial decomposition of the control action in the fixed region.

Representation of Airspace and Plane Trajectories

Given a control region R, the representation of the space and planned trajectories must be accurate enough to permit real-time data processing. To this end, in the following, airspace is described through a network representing both the airways and the terminal areas. In detail, the network nodes represents:

- the intersection of the airways with the boundary of R;
- the intersection of the airways with the boundaries of terminal areas;
- the intersection of two or more airways;
- the waypoints fixed on each airway.

Obviously, the arcs represent all possible connections of the nodes.

A set of discrete altitude levels is also associated with each node so that constraints on vertical separation, imposed by safety requirements, are respected.

As for the representation of flights, we utilize a discrete model in which trajectories are defined by the set od nodes to be traversed, by the corresponding altitude levels and by the passing times on each node. In this model we do not consider the detailed dynamics of flight between consecutive nodes and within the terminal area; nevertheless, the problem solutions are forced to be consistent with the performance of airplanes.

In the following section we suggest a mathematical formulation of SCF, as it has been previously described. In particular, the problem is formulated as an optimization problem with the hypothesis that the nodes to be traversed are fixed on the basis of the nominal flight plan. In this framework, one must determine altitude levels and passing times on the node trajectories, so that an appropriate set of constraints is satisfied and a cost function is minimized.

7. Mathematical model

Notations

R: strategic planning region;

$[t_0, t_0 + T]$: strategic planning interval;

$I = \{1,\ldots,i,\ldots,N\}$: set of indexes of flights in R in the interval $[t_0, t_0 + T]$;

$K = \{1,\ldots,k,\ldots,N\}$: set of indexes of nodes in R;

$K^i = \{k^i_0,\ldots,k^i_f\} \subseteq K$: set of nominal trajectory nodes traversed in R by plane i in the interval $[t0, t0 + T]$;

$\tilde{k}$: node following k in K^i;

$\bar{t}^i_k$: nominal passing time of plane i on node k;

t^i_k : actual passing time of plane i on node k;

H : set of indexes of altitude levels allowable in nodes of R;

$H^i_k \subseteq H$: set of indexes of altitude level allowable for plane i on node k;

z_h, $h \in H$: altitude level h;

q^i_{kh}, $h \in H^i_k$: binary variable indicating whether the plane i occupies or not the altitude level h of node k;

$\bar{q}^i_{kh}$, $h \in H^i_k$: binary variable indicating whether the plane i occupies or not the altitude level h of node k in the nominal trajectory;

$P^i_{\tilde{k}h} \in H^i_{\tilde{k}}$: set of indexes of altitude levels allowed in node $\tilde{k}$, when plane i starts from level h of node k.

Altitude Constraints

First of all, we must specify the set H^i_k of altitude levels allowable for plane i in each node of the trajectory. With reference to the definition of $P^i_{\tilde{k}}$ we have

$$H^i_{\tilde{k}} = \bigcup_{h \in H^i_k} P^i_{\tilde{k}h}, \quad \tilde{k}, k \in K^i.$$

To determine $P^i_{\tilde{k}h}$ correctly, we must consider that a speed range, consistent with the performance of plane i, corresponds to each allowable level. Then, knowing the distance between $\tilde{k}$ and k, we can determine, for each speed, the altitude levels reachable in $\tilde{k}$. Among these levels, we will only consider levels satisfying eventual control needs in $\tilde{k}$ and/or limitation on the maximum shifting of the actual altitude level from the nominal one. Therefore, if we introduce the binary variables q^i_{hk}, the altitude constraints can be expressed as

$$\sum_{h \in H^i_k} q^i_{kh} = 1, \quad \forall k \in K^i, \quad \forall i \in I \tag{9}$$

$$\sum_{h \in H^i_k} q^i_{hk} \sum_{j \in P^i_{kj}} q^i_{\tilde{k}j} = 1, \quad \forall k, \tilde{k} \in K^i, \quad \forall i \in I. \tag{10}$$

The first set of constraints states that plane i can occupy only one of the allowable levels of node k. The second concerns altitude variations when plane i goes from k to $\tilde{k}$. It states that, if plane i is in level h of k, it can occupy only one level of $P^i_{\tilde{k}j}$ in node $\tilde{k}$.

Transit Time Constraints

As regards the transit time on nodes, denoted with t^i_k, we must consider constraints depending on aircraft performance, safety standards and possible restrictions of traffic flows on the airways network. Constraints on aircraft performance can be expressed as

$$t^i_{\tilde{k}} \leq t^i_k + \sum_{h \in H^i_k} \sum_{j \in P^i_{\tilde{k}h}} q^i_{kh}\, q^i_{\tilde{k}h}\, \tau^{Mi}_{khj} \quad , \quad \forall k,\tilde{k} \in K^i,\ \forall i \in I \tag{11}$$

$$t^i_{\tilde{k}} \geq t^i_k + \sum_{h \in H^i_k} \sum_{j \in P^i_{\tilde{k}h}} q^i_{kh}\, q^i_{\tilde{k}h}\, \tau^{mi}_{khj} \quad , \quad \forall k,\tilde{k} \in K^i,\ \forall i \in I \tag{12}$$

$$T^i_{om} \leq t^i_{k^i_o} \leq T^i_{oM} \quad , \quad \forall\, i \in I \tag{13}$$

$$T^i_{fm} \leq t^i_{k^i_f} \leq T^i_{fM} \quad , \quad \forall\, i \in I \tag{14}$$

where τ^{Mi}_{khj} and τ^{mi}_{khj} denote, respectively, maximum and minimum time intervals from level h of node k to level j of node $\tilde{k}$; T^i_{om} and T^i_{oM} indicate limits on the initial node; T^i_{fm} and T^i_{fM} indicate limits on the final node.

Moreover, as regards the safety constraints and possible further restrictions caused by flow control, we can state, for every two planes, the following condition

$$|t^i_k - t^r_k| \geq \left[\sum_{h \in H^i_k \cap H^r_k} q^i_{kh}\, q^r_{kh} \right] \Delta T^{i,r}_k \tag{15}$$

$$\forall k \in K^i \cap K^r,\ \forall (i,r) \in I,\ i \neq r$$

where $\Delta T^{i,r}_k$ is the minimum time separation. Expression (15) ensures that if planes i and r cross node k at the same level, they have a minimum separation time $\Delta T^{i,r}_k$.

Transit Time Intervals

Exact specification of transit times in each node can be meaningless, especially for those nodes far from the initial one. To improve this model it is convenient to associate an interval of transit times, consistent with plane performances and assigned standard separations, with each plane and each node instead of a single transit time. To this end, constraints (11)-(14) must be modified.

Now, let $[t^i_k - \Delta t^i_k , t^i_k + \Delta t^i_k]$ be the interval of transit times of plane i on node k. The expressions (11)-(14) become, respectively

$$t^i_{\tilde{k}} \leq t^i_k + \Delta t^i_k + \sum_{h \in H^i_k} \sum_{j \in P^i_{\tilde{k}h}} q^i_{kh} \, q^i_{\tilde{k}h} \, \tau^{Mi}_{khj} - \Delta t^i_{\tilde{k}}, \quad \forall k, \tilde{k} \in K^i, \forall i \in I \qquad (11^*)$$

$$t^i_{\tilde{k}} \geq t^i_k + \Delta t^i_k + \sum_{h \in H^i_k} \sum_{j \in P^i_{\tilde{k}h}} q^i_{kh} \, q^i_{\tilde{k}h} \, \tau^{mi}_{khj} + \Delta t^i_{\tilde{k}}, \quad \forall k, \tilde{k} \in K^i, \forall i \in I \qquad (12^*)$$

$$T^i_{om} + \Delta t^i_{k^i_o} \leq t^i_{k^i_o} \leq T^i_{oM} - \Delta t^i_{k^i_o} \quad , \quad \forall i \in I \qquad (13^*)$$

$$T^i_{fm} + \Delta t^i_{k^i_o} \leq t^i_{k^i_o} \leq T^i_{fM} - \Delta t^i_{k^i_o} \quad , \quad \forall i \in I. \qquad (14^*)$$

Expressions (11*) and (12*) ensure that any transit time on node $\tilde{k}$, relative to plane i and included in the interval $[t^i_{\tilde{k}} - \Delta t^i_{\tilde{k}} , t^i_{\tilde{k}} + \Delta t^i_{\tilde{k}}]$ is always consistent with plane performance, while transit time on node k assumes any value in the interval $[t^i_k - \Delta t^i_k , t^i_k + \Delta t^i_k]$.

Expressions (13*) and (14*) are similar to (13) and (14) and extend to intervals the limitations on the initial and final nodes.

Obviously, system (11*)-(14*) is meaningful if conditions

$$\tau^{Mi}_{khj} - \tau^{mi}_{khj} \geq 2 \, [\Delta t^i_k + \Delta t^i_{\tilde{k}}], \quad \forall h \in H^i_k , \quad \forall j \in P^i_{\tilde{k}h}$$

$$T^i_{oM} - T^i_{om} \geq 2 \, \Delta t^i_{k^i_o} , \; T^i_{fM} - T^i_{fm} \geq 2 \, \Delta t^i_{k^i_f}$$

are satisfied.

As far as separation constraints are concerned, (15) becomes

$$|t^i_k - t^r_k| \geq \left[\Delta t^r_k + \Delta t^i_k + \Delta T^{i,r}_k \right] \left[\sum_{h \in H^i_k \cap H^r_k} q^i_{kh} \, q^r_{kh} \right] \qquad (15^*)$$

$$\forall \, k \in K^i \cap K^r, \; \forall \, (i, r) \in I, \, i \neq r.$$

Waiting Times

The suggested model may have no feasible solution because of transit time constraints. To avoid this circumstance, we can increase the degrees of freedom of the system by providing the possibility of waiting at prefixed nodes. To this end, it is sufficient to introduce in (11) or in (11*) a term representing waiting times. In detail, (11*) could be replaced by

$$t^i_{\tilde{k}} \le t^i_k - \Delta t^i_k + \sum_{h \in H^i_k} \sum_{j \in P^i_{\tilde{k}h}} q^i_{kh}\, q^i_{\tilde{k}h}\, \tau^{Mi}_{khj} - \Delta t^i_{\tilde{k}} + \tau^i_k$$

$$a_i\, \mu^i_k \le \tau^i_k \le b_i\, \mu^i_k\,, \quad \forall k, \tilde{k} \in K^i\,, \quad \forall i \in I$$

where μ^i_k is a binary variable equal to 1 if plane i waits at node k a time $\tau^i_k \neq 0$ and equal to zero otherwise.

Objective Function

We may consider several objective functions. The first possibility is to minimize total delay of planes in the control region. In this case, the objective function can be expressed as

$$J_1 = \sum \alpha^i_{k^i_f}\; t^i_{k^i_f}$$

where $\alpha^i_{k^i_f}$ are weight coefficients depending on the plane class and the terminal node.

In particular, J_1 seems meaningful in the terminal area, where, because of operating reasons, the main objective is to minimize the final delay.

When we consider larger airspace regions, we must also take into account the cost connected with deviations from the nominal flight profile. In this case, the objective function could be expressed as

$$J_2 = \sum_{i \in I} \sum_{k \in K} \left\{ \sum_{h \in H_k^i} \sum_{j \in H_k^i} \alpha_{khj}^i \; q_{kh}^i \; q_{\tilde{k}j}^i \left| t_{\tilde{k}}^i - t_k^i - \bar{\tau}_{khj}^i \right| + \right.$$
$$\left. + \beta_k^i \left| \sum_{h \in H_r^i} (q_{hk}^i - \bar{q}_{hk}^i)\, z_h \right| \right\}$$

where $\bar{\tau}_{khj}^i$ represents, for plane i, the transit time from level h of node k to level j of the subsequent node following a minimum fuel consumption profile; α_{khj}^i and β_k^i are appropriate weight coefficients. The first term of J_2 represents the total shifting of transit times between pairs of adjacent nodes from the corresponding times in the minimum consumption profile; the second term represents the total deviation of altitude levels from the nominal ones. Therefore, J_2 could be utilized as a cost index related to the total final consumption.

In this model one must also notice that, if delay costs and costs of deviations from the nominal flight profile could be evaluated in homogeneous quantitative terms, we could take as an objective function $J_3 = J_1 + J_2$, where arrival times and fuel consumption are simultaneously considered.

On the basis of this model, the minimization of any one objective function with the previous constraints, is a non-linear mixed variables optimization problem.

8. Problem decomposition

The general model illustrated above involves several computational and operational difficulties. In fact, mathematical complexity increases with the number of planes to be controlled; moreover, the provided control methodology may conflict with current ATC procedures. In fact, the suggested control philosophy may require, for every new plane entering the control region, an intervention on all planes previously planned. Consequently, this could involve heavy operational difficulties. To partially overcome these difficulties, in the following, we shall consider a simplified model of SCF based on the hypothesis that flights planning is carried out according to the FIFO (first in, first out) discipline. Specifically, the planning for each new plane entering the control region is carried out assuming, as constraints, the flight plans of airplanes already in the system.

This hypothesis obviously reduces solution optimality, but it reduces also mathematical complexity (variables relative to only a single plane must be taken into account). Moreover, the suggested control method becomes consistent with current ATC procedures. On this basis, the mathematical model can be simplified considering planning for a single plane at a time.

Since minimization of $J_3 = J_1 + J_2$ remains a problem whose difficulty increases with the number of nodes of the trajectory, a decomposition algorithmic procedure of the control problem is illustrated in the following. In this way, in general, a sub-optimal solution of the global problem can be obtained. The solution method consists of two sequential steps:

1) determine a solution satisfying constraints (9)-(15) and minimizing only deviations of altitude levels from the nominal ones;
2) for the levels fixed in the first step, determine a solution satisfying constraints (11)-(15) and minimizing transit time deviation from those relative to the nominal flight path (or delay at the terminal node).

Thus, we obtain two different sub-problems that can be denoted, respectively, as "altitude control" and "speed control".

In the following, we review only the fundamental mathematical aspects of each problem and the techniques utilized to solve them. More details on the mathematical models and solution algorithms can be found in an earlier paper (Andreussi and others, 1981).

Altitude Control

Let $K = \{1, 2, \ldots, k, \tilde{k}, \ldots, N\}$ be the ordered set of nodes to be traversed by a plane and denote with $\hat{t}^r_k$ the transit times of planes already planned for node k; constraints (9)-(15) become the following

$$\sum_{h \in H_k} q_{kh} = 1, \quad \forall k \in K \tag{9'}$$

$$\sum_{h \in H_k} q_{kh} \sum_{j \in P_{kh}} q_{kj} = 1, \quad \forall k < N \tag{10'}$$

$$t_{\tilde{k}} \le t_k + \sum_{h \in H_k} \sum_{j \in P_{\tilde{k}h}} q_{kh}\, q_{\tilde{k}j}\, \tau^M_{khj}, \quad \forall k < N \tag{11'}$$

$$t_{\tilde{k}} \ge t_k + \sum_{h \in H_k} \sum_{j \in P_{\tilde{k}h}} q_{kh}\, q_{\tilde{k}j}\, \tau^m_{khj}, \quad \forall k < N \tag{12'}$$

$$t_{om} \le t_1 \le t_{oM} \tag{13'}$$

$$t_{fm} \le t_N \le t_{fM} \tag{14'}$$

$$| t_k - \hat{t}^r_k | \ge \left(\sum_{h \in K_k \cap H^r_k} q_{kh}\, q^r_{kh} \right) \Delta T^r_k, \quad \forall k \in K, \ \forall r \in \hat{I}_k \tag{15'}$$

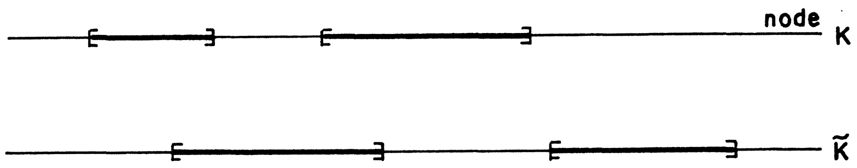
where m and M, as denoted before, indicate, respectively, the minimum and the maximum values of the time parameters and I_k is the set of indexes of planes already planned to pass through node k.

The problem corresponding to step 1) and referred to as "altitude control" is, therefore, that of determining altitude levels so that the function

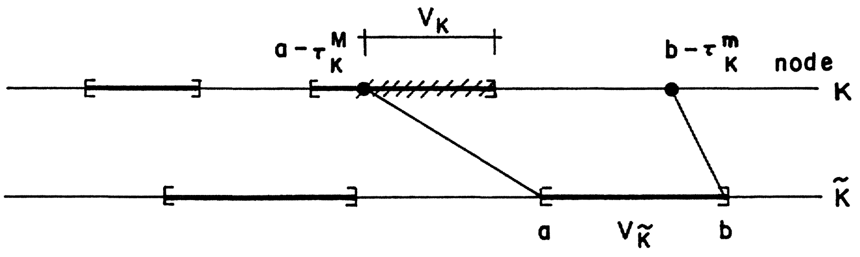
$$J_q = \sum_{k \in K} \beta_k \left| \sum_{h \in H_k} (q_{kh} - \bar{q}_{kh})\, z_h \right|$$

is minimized and constraints (9')-(15') are satisfied.

The problem, thus defined, is one with mixed variables, h_k and t_k, where the variables t_k are not present in the objective function and constraints on t_k require that at least a sequence of time intervals, in pairs consistent according to what is represented in Figure 5, exist. It follows that, given the altitude levels h_k, we can verify their feasibility through a finite number of comparisons. Therefore, the optimal solution h^*_k can be determined through a classical branch and bound algorithm.



a) Feasible time windows: the case of one altitude level for each node



b) Reflection of feasible time windows of node $\tilde{k}$
and representation of consistent time intervals V_k, $V_{\tilde{k}}$

Fig. 5. Examples of feasible time windows of two subsequent nodes and corresponding consistent time intervals

REMARK 1. If nominal altitudes are feasible (with respect to plane performance, separation constraints and possible limitations on available altitude levels) the optimal solution is determined in the first cycle of the algorithm.

REMARK 2. Corresponding to each feasible solution, a sequence $\{V_k\}$ of consistent time sets, with $k = 1, 2, \ldots, N$ is also determined. Thus, a sequence of transit times on each node satisfying all constraints, can be easily defined.

Speed Control

The problem referred to as "speed control" is to determine transit times at each node so that the function

$$J_V = \alpha_N t_N + \sum \alpha_k |t_{\tilde{k}} - t_k - \bar{\tau}_k|$$

is minimized and constraints on the performance of the controlled airplane and on safety are satisfied. Since in this problem we assume that the altitude levels are the nominal ones or those computed by the algorithm for altitude control, then the control action carried out is equivalent to the control of the plane speed.

With these hypotheses, constraints (11)-(15) can be further simplified by eliminating altitude variables. Therefore, they become

$$t_{\tilde{k}} \leq t_k + \tau_k^M, \quad \forall k < N \tag{11"}$$

$$t_{\tilde{k}} \geq t_k + \tau_k^m, \quad \forall k < N \tag{12"}$$

$$t_1^m \leq t_1 \leq t_1^M \tag{13"}$$

$$t_1^m \leq t_N \leq t_N^M \tag{14"}$$

$$|\hat{t}_k^r - t_k| \geq \Delta t_k^r, \quad \forall k \in K, \quad \forall r \in \hat{I}_k. \tag{15"}$$

The problem thus formulated is a linear programming one with mixed variables where the combinatorial aspect is due to constraints (15"). To solve this problem, we have carried out a hybrid algorithm that utilizes dynamic programming and branch and bound techniques. Dynamic programming is used to transform a problem with mixed variables into a problem with only discrete variables. The latter is then solved by an enumerative procedure.

An important aspect of this method is that, since no discretization of the continuous variables t_k is necessary, the optimal solution can be reached.

9. Computational results

The algorithms relative to altitude and speed control, previously described, have been implemented in a UNIVAC 1100/82 computer and different series of tests have been performed. Specifically, in a first step, the altitude levels have been determined; subsequently, transit times on the prefixed nodes have been computed. With this approach, the altitude levels and the consistent time intervals determined by the first algorithm are utilized as input data for the second algorithm.

The numerical tests were carried out by considering an increasing number of nodes N = 5, 10, 15, 17, 20 for a total of 400 tests. For each test, the input variables have been uniformly distributed over fixed intervals and the relative values have been randomly generated. Moreover, appropriate limits to ensure reciprocal consistency and operational meaningfulness have been imposed. For example, the number of levels at each node is five and upper and lower bounds for $\bar{\tau}$, τ^m, τ^M are derived from the speed/altitude aeroperformance diagrams.

The simulation results are reported in Table 3 where, for each value of N, the most significant computation times are indicated. In particular, t_a is the time required to determine altitudes and it coincides with the time necessary to find the first global feasible solution; t_0 is the time needed to find the optimal solution and t_c is the global computation time. The results show that the most time-consuming algorithm is the altitude control algorithm. Little computation effort is required, on the contrary, by the speed control algorithm.

The computation times also suggest that the proposed approach can be used as a tool to compute on-line conflict-free four-dimensional paths, as long as the number of nodes in the trajectory is less than 20. For N = 20, the algorithm begins to look unsatisfactory. These results demonstrate that the algorithm could be used, for example, in a strategic control system in large Terminal Areas.

However, one must also note that computation times depend strongly both on the number of nodes N and on the number of feasible windows corresponding to the various flight levels. This indicates that the tests that increase most the mean computation times are those that consider traffic congestion situations, which occur only under special operational conditions.

Table 3. Computational results of the strategic control model

Number of nodes	5	10	15	17	20
Number of tests	150	110	100	40	10
t_a (sec.)	.014	.27	6.2	17	102
t_o (sec.)	.014	.27	7.2	17	102
t_c (sec.)	.014	.37	7.4	19	112

10. Tactical control

Today the air traffic controller uses radar to assess the traffic situation and speed, altitude and heading instructions to separate and space aircraft. This is the "tactical" system using relative - position separation in which the planning horizon is very short and the situation is allowed to develop before solutions are offered. The pilot does his own navigation only when there are no nearby aircraft. As the flight encounters traffic of increasing density, ATC intervenes more and more until, in a busy terminal area, the controller is vectoring the airplane continuously in all three dimensions and in airspeed. This control system is manpower intensive, uses relatively inefficient flight paths to solve traffic situations and is at or near its capacity in handling traffic and supporting increased runway operations. This management philosophy is not adeguate in congestion situations and, in particular, near the main airports where, owing to user requirements, traffic distribution is highly concentrated.

The effects of this situation are felt mostly in the arrival and departure phases. In fact, in the last years it has been assessed that only 10% of the delays occur along the aircraft routes, while the corresponding percentages for arrivals and departures are, respectively, about 60% and 30%. For this reason, in the last years, specific attention has been devoted to the management of the Terminal Area (TMA) where the overall system efficiency becomes nearly as important as safety. Moreover, a reduction of human intervention in the management of operations is needed and, consequently, more automated systems have been proposed. From this point of view the TMA problem is "the automation of the aircraft flow control and sequencing in the proximity of the airport so as to satisfy an optimality criterion".

As a consequence, the need arises to define a model which can be dealt with mathematically and to analyze all those elements which constitute a semi-or fully-automated control system.

To contribute in solving the TMA control problems, in this paper, we consider in particular the Aircraft Sequencing Problem (ASP).

11. The aircraft sequencing problem

During peak traffic periods, the control of aircraft arrivals and departures in a TMA becomes a very complex task. Air traffic controllers, among other things, must guarantee that every aircraft, either waiting to land or preparing to take off in such a congested area, maintains the required degree of safety. They also have to decide what aircraft should use a particular runway, at what time this should be done and what manoeuvres should be executed to achieve this. The viable accomplishment of such a task becomes more difficult in view of the fact that aircraft are continuously entering and leaving the system and that, at peak periods, the demand for runway occupancy may reach or even exceed the capabilities of the system. It is at such periods that excessive delays are often observed, resulting in passenger discomfort, fuel waste and disruption of the airlines' schedules.

Under such "bottleneck" conditions, an increase in collision risk can logically be expected as well. As a consequence, because of safety considerations, the structure of TMA is rigidly defined and all aircraft must fly in a manner satisfying prefixed procedural constraints.

To simplify the understanding of the problem we refer to a TMA as shown in Figures 6 and 7 and we consider only landings, even if the approach proposed in the following allows to simultaneously take into account takeoffs.

Then, the following aspects must be underlined:

a) every aircraft must approach the runway for landing, flying along one of the prestructured paths of TMA;
b) the runway can be occupied by only one aircraft at a time;
c) every aircraft must fly along the common approach path following a standard descent profile;
d) during all the approach phases a separation standard between every pair of consecutive aircraft must be maintained;
e) the sequencing strategy used by almost all major airports of the world is still today the "First-Come First-Served" (FCFS) discipline.

As it is well known, FCFS strategy is very simple to implement, but it is likely to produce excessive delays. Therefore, an effort must be made to minimize the delay or optimize some other measure of performance related to passenger discomfort, without violating safety constraints.

Consequently, the TMA problem can be stated as follows: "Given a set of aircraft entering the TMA and given, for each aircraft, the Preferred Landing Time (PLT), the runway occupancy time, the cost per unit time of flight, the geometry of the approach path and glide path and the corresponding aircraft speeds, assign to each aircraft the starting time from the fix and the approach path in such a way that the procedural constraints are satisfied and a system performance index is optimized".

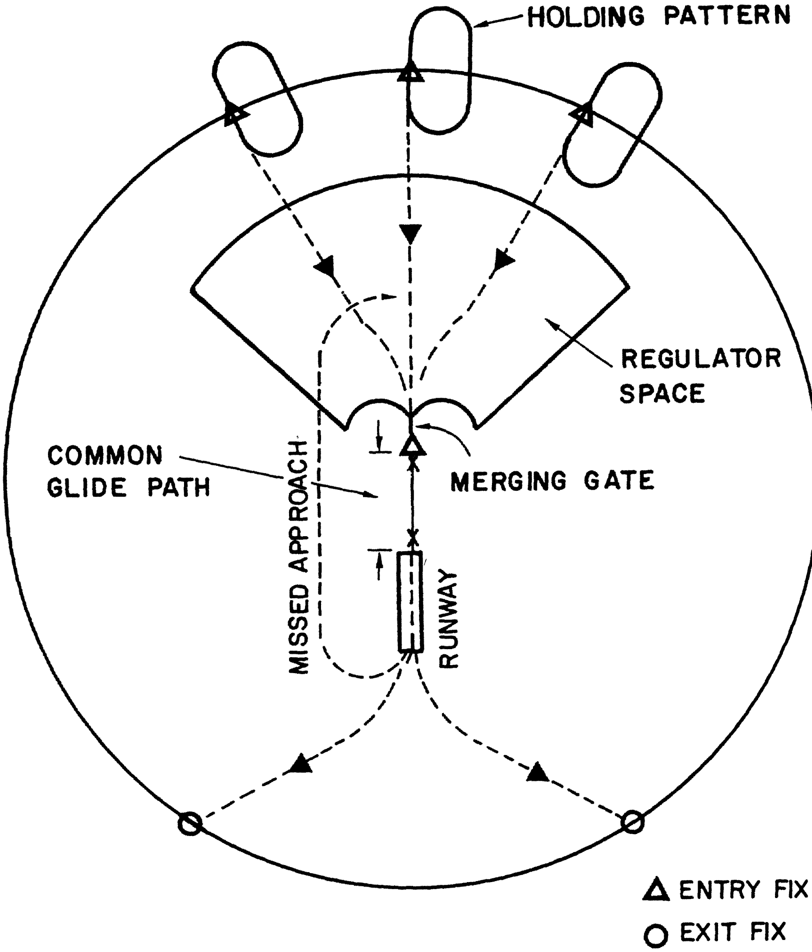


Fig. 6. Structure of TMA

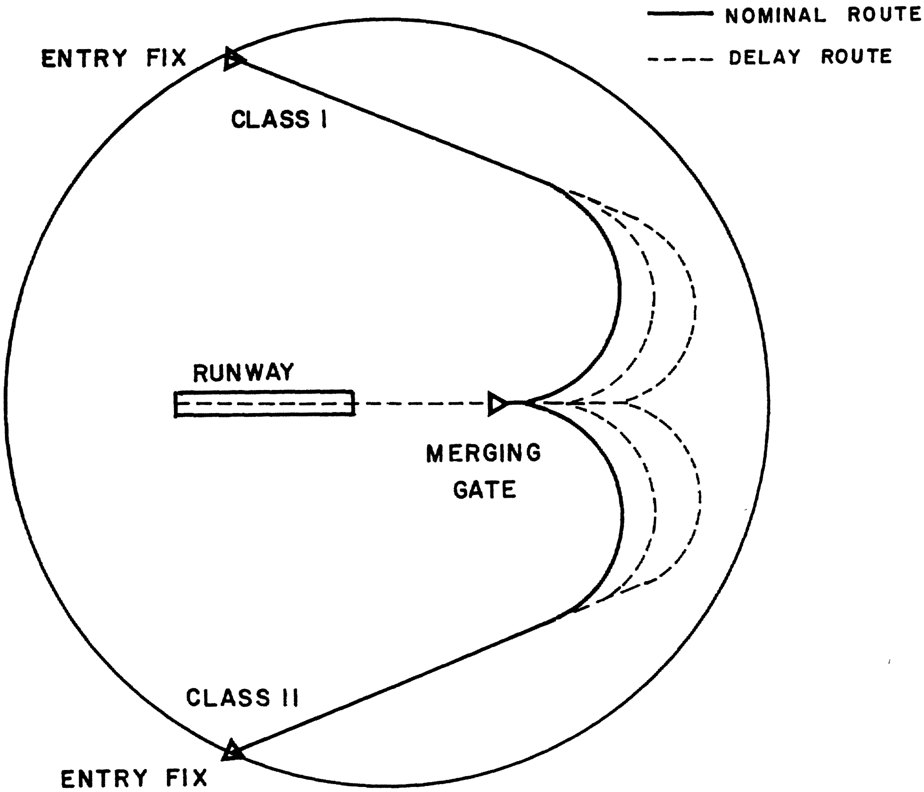


Fig. 7. Delay routes

With the TMA operating in the aforementioned way, the TMA problem can be decomposed into the two following sub-problems:

1) given the contraints on aircraft performance, the initial and final states (position and speed) and the pre-established flight time, determine the optimum trajectories which connect these states with the specified flight time;
2) given a set of PLT and the maximum admissible delay, determine the Actual Landing Times (ALT) sequence which satisfies the procedural constraints on the runway and the glide path and optimizes a system performance index.

To a large extent, these two problems are independent. In fact, as the required controls to follow the approach paths can be calculated in advance, it is possible to predetermine the optimal flight path. Therefore, the need of "real-time" calculations is limited only to sub-problem 2, called the Aircraft Sequencing Problem (ASP), which is the topic discussed here. The goal of solving the ASP is, at least theoretically, achievable for two reasons.

First, safety regulations state that any two coaltitudinal aircraft must maintain a "minimum horizontal separation", which is a function of the types and relative positions of these two aircraft.

Second, the "landing speed" of any type of aircraft is generally different from the landing speed of another aircraft.

A consequence of the variability of the above parameters (minimum horizontal separation and landing speed) is that the "minimum permissible time interval" between two successive landings is a variable quantity.

Thus, it may be possible, by rearranging the initial position of the aircraft, to take advantage of the above variability and obtain a landing sequence that results in a more efficient use of the runway as compared with that obtainable by using the FCFS discipline. In fact, an optimal sequence does exist; it is thoretically possible to find it by examining all sequences and selecting the most favorable one.

The method suggested above to determine the optimal sequence is safe, but extremely inefficient, because the computational effort associated with it is a factorial function of the number of aircraft and it is not possible to evaluate all combinations in a short time interval (as the nature of ASP requires) even on the fastest computer. To give an idea of the difficulty, it is sufficient to consider that, with only 10 aircraft, we would have to make 3,628,000 comparisons and with 15 aircraft 1,307,674,368,000 comparisons.

It should be also pointed out that while the main factor that suggests the existence of an optimal landing sequence is the variability of the minimum permissible time interval between two successive landings, it is the same factor that makes the determination of this optimal sequence a nontrivial task.

Moreover, the real world problem involves many other considerations, especially as far as the implementation of sequencing strategies is concerned.

For these reasons, the relevant literature on the subject has been, till now, considerable and growing [see for example Odoni, 1969; Tobias, 1972; Park and others, 1972,1973; Bianco and others, 1978,1979; Trivizas, 1985].

Three papers, in particular, seem to offer an adequate solution to the operational needs.

In the first (Dear, 1976), an excellent investigation of the ASP was carried out. In particular, the author points out that, in order to determine the landing order, we need to consider all aircraft currently in the system. As this number can be very large (20 or even more simultaneously), he raises serious doubts on the possibility of reaching an optimal solution in real-time even with pseudo-enumerative techniques. Therefore, he resorts to a simulation model where identical arrival streams, under various sequencing strategies, are compared. In particular, he proposes a "Constrained Position Shifting" (CPS) strategy instead of FCFS strategy. That is, no aircraft may be sequenced forward or rearward more than a prespecified number of positions (Maximum Position Shifting) from its FCFS position.

The second paper (Psaraftis, 1979, 1980) takes into account Dear's proposal about the CPS management concept, but develops an exact optimization algorithm based on the dynamic programming approach and referring to the "static" case when all aircraft are supposed to wait to land at a given time. In reality, every aircraft entering TMA has an earliest landing time (which is the PLT) depending on the characteristics of TMA, the aircraft speed, pilot preferences and so on. Therefore, the aircraft to be sequenced wait to land at different times. Fig. 8 shows an example of a real world situation. In this case, Psarafits approach cannot be easily utilized.

In a third paper (Bianco and others, 1987), a new combinatorial optimization approach, more consistent with the real environment, is proposed.

In the following, the basic ideas of this last work and an outline of the optimization model and of the solution algorithm are reported and discussed.

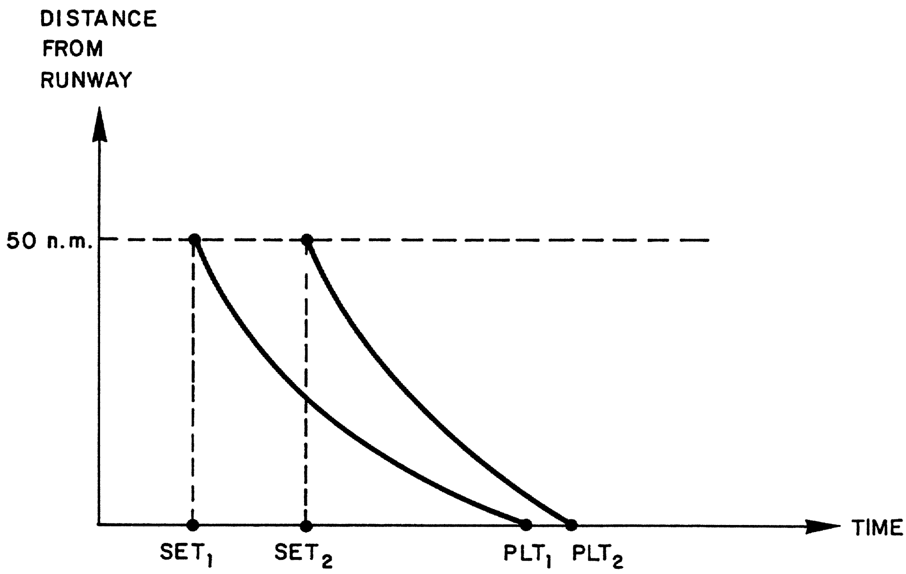


Fig. 8. Example of aircraft conflicting on the runway

11. A job-scheduling formulation of the ASP

Suppose that the air traffic controller is confronted with the following problem: A number n of aircraft are waiting to land at different PLT_i at a single-runway airport. His task, then, is to find a landing sequence for these aircraft, so that a certain measure of performance is optimized, while all problem constraints are satisfied.

We now make the problem statement more specific:

1) It is assumed that the pilots of all aircraft are capable and willing to execute the instructions of the controller given enough prior notice.
2) The measure of performance selected is the "Last Landing Time" (LLT). The corresponding objective is then to find a landing

sequence such that the aircraft that lands last does this as quickly as possible.

3) Concerning the problem constraints, only the satisfaction of the "minimum inter-arrive time" constraints is required. This means that the time interval between the landing of an aircraft i, followed by the landing of an aircraft j, must not be less than a known time interval t_{ij}.
4) The composition of the set of aircraft waiting to land is, of course, assumed to be known. For each ordered pair (i,j) of aircraft, the minimum time interval t_{ij} is also known.
5) At any stage of the sequencing procedure, the air traffic controller is free to assign the next landing slot to any of the remaining aircraft. This means that we ignore the initial positions which the aircraft had when they arrived at TMA.

At this point it is not difficult to see that the problem described above can be represented by means of a particular "job-machines scheduling" model. In fact, with the aforementioned assumptions, the following analogy may be established:

a) to each landing operation is associated a job;
b) the runway corresponds to a machine with capacity one;
c) the PLT_i of aircraft i corresponds to the ready time r_i of job i;
d) the ALT_i of aircraft i corresponds to the start time t_i of job i;
e) the LLT corresponds to the maximum completion time $C_{\max}$ of the schedule;
f) the minimum time interval t_{ij} between the landing of aircraft i, followed by the landing of aircraft j, corresponds to the processing time p_{ij} of job i when it depends on job j following in the sequence.

Therefore, the ASP, as defined here, can be mathematically reformulated as the $n|1|r_i|seq\text{-}dep|C_{\max}$ scheduling problem. For this problem the following approach (Bianco and others 1985) can be utilized.

12. The $n|1|r_i|seq\text{-}dep|C_{\max}$ model

Let $J = (1, 2,..., n)$ be a set of n jobs to be processed on a single machine, and denote by r_i the ready time of the job i. A matrix (p_{jj}), with $i,j \in J0 = J \cup \{0\}$, is given where p_{ij} $(i \neq 0)$ is the "processing time" of the job i if job j is the successor of i in the sequence ($j = 0$, if i is the last job in the sequence) and p_{oi} is the "setup time" of the machine when the sequence starts with job i.

The $n|1|r_i|seq\text{-}dep|C_{\max}$ n problem can be formulated using the following integer programming model:

$$\min \left(s + \sum_{i \in J0} \sum_{j \in J0} p_{ij} \, x_{ij} \right) \tag{16}$$

subject to

$$t_i + \sum_{i \in J0} p_{ij} \, x_{ij} - \sum_{k \in J0} \sum_{j \in J0} p_{kj} \, x_{kj} - s \leq 0, \quad i \in J0 \tag{17}$$

$$r_i - t_i \leq 0, \quad i \in J0 \tag{18}$$

$$(p_{ij} + T_{ij}) \, x_{ij} + t_i - t_j - T_{ij} \leq 0, \quad i \in J0, \ j \in J, j \neq i \tag{19}$$

$$\sum_{j \in J0} x_{ij} = 1, \quad i \in J0 \tag{20}$$

$$\sum_{i \in J0} x_{ij} = 1, \quad j \in J0 \tag{21}$$

$$s \geq 0, \tag{22}$$

$$t_0 = 0, \tag{23}$$

$$x_{ij} \in \{0,1\}, \quad i \in J0, \ j \in J, j \neq 0 \tag{24}$$

$$t_i \geq 0, \quad i \in J0 \tag{25}$$

where:

x_{ij} = 1 if job i directly precedes job j

x_{ij} = 0 otherwise

t_i is the start time of job i

s is the machine idle time

T_{ij} are chosen to make constraints (19) redundant whenever $x_{ij} = 0$

REMARK 1. This problem is NP-hard and in the case of zero ready times it reduces to the asymmetric travelling salesman problem (ATPS). Moreover, constraints (19) both prevent sub-tours in the ATSP solution and avoid having two jobs simultaneously processed.

REMARK 2. The above formulation can be considered as an ATSP with unlimited time windows ($r_i \leq t_i < +\infty$).

13. Outline of the solution algorithm

The solution of the above problem can be obtained by a pseudo-enumerative procedure exploiting both some peculiar properties of the problem and efficient lower and upper bounds.

Branching Phase

Denote by C(σ) the completion time of a subsequence σ. The following two dominance criteria, (see Bianco and others, 1985), are utilized for tree pruning:

Theorem 1. A subsequence $\sigma\pi k$ is dominated if there exists a permutation π' of π such that

$$C(\sigma\pi k) > C(\sigma\pi' k)$$

Theorem 2. A subsequence $\sigma j \sigma i$ is dominated if

a) $r_i > \max\ \{r_i, C(\sigma)\} + p_{ij}$

b) $p_{ki} + p_{ih} \geq p_{hk},\ \forall h,\ k \in J / \{i\}$.

Then, the following branching strategy can be stated.

With each node of the enumeration tree is associated a partial sequence σ in which the first k jobs have been fixed. As a first step, Theorem 2 is used to define, among the unscheduled jobs, a set of candidates for the $(k + 1)$th position in the optimal sequence. Subsequently, for each candidate i, the procedure checks whether or not it is dominated according to Theorem 1. If it is dominated, a new candidate is examined; otherwise, the lower bound is computed and, if it is less than an upper bound of the optimal solution, a new node, representing the subsequence σi, is generated.

At this point, the ready times of the unscheduled jobs which are smaller than $C(\sigma i)$ are set to $C(\sigma i)$.

Backtracking takes place whenever a complete sequence has been produced or all the candidates at a given level have been examined.

Bounding Phase

At each node of the enumerative tree we have a new problem with job-set $J/\{\sigma\}$, new ready times and new set-up times. To this problem two lower bounds and an upper bound are applied.

1) Lagrangean Lower Bound (LLB)
   It is obtained as the solution of the lagrangean problem obtained by dualizing constraints (17) and (18), and dropping constraints (19) in the original model.
2) Alternative Lower Bound (ALB)
   As the LLB tends to be weak in case of small variations in the matrix $\{p_{ij}\}$, the following heuristic rule can be utilized:
   (i) with each job $i \in J$ associate the processing time $p_i = \min_j\{p_{ij}\}$;
   (ii) schedule the jobs using the FCFS rule and take the completion time as the ALB of the original problem.

REMARK. Observe that ALB is the optimal solution of our problem when processing times do not depend on the sequence, and hence, in the general case, ALB is a lower bound on the optimal solution.

3) Upper Bound (UB)

As observed in the previous remark, the EST (Earliest Start Time) rule is optimal if the processing times are sequence-independent. Consequently, a reasonably good feasible solution can be obtained by associating with each job an average processing time $p_h = (\sum_{j\in J0} p_{ij})/n$ and sequencing the jobs according to the EST rule. The upper bound UB is then obtained by computing the completion time of the EST sequence using the original processing times. Next, we improve the UB value by applying two procedures based on Theorem 1.

The first one performs exchanges among pairs of adjacent jobs till a 2-exchange local optimum is reached.

To the resulting sequence, a second procedure is applied. It works as follows: Given a parameter m, for $i = 0,\ldots, n\text{-}m$, the current sequence is replaced by the best among the sequences obtained by permuting the jobs in the position from $i+1$ to $i+m$. Notice that the resulting sequence is not in general an m-exchange local optimum.

14. Computational results

To test the efficiency of the algorithm when applied to real-world problems, we must consider that aircraft, waiting to land, can be classified into a relatively small number m of distinct "categories" according to speed, capacity, weight and other technical characteristics.

As a consequence, the minimum interarrival times between two successive aircraft is a function only of the categories they belong to.

We have exploited this clustering of aircraft into categories to drastically reduce the size of the enumeration tree. In fact, it can be easily seen that, at the k-th iteration, only the earliest aircraft of each category, is eligibile to be scheduled at position $(k + 1)$-th.

The following Table 4 represents the minimum interarrival times relative to the main categories of commercial aircraft, while Tables 5 and 6 illustrate the results of two realistic large scale problems with 30 and 44 aircraft, respectively.

The CPU times required to find the optimal solution with the algorithm coded in Pascal and tests carried out on a Vax-11/780, have been respectively 373 sec. and 1,956 sec. Even though these times might seem not compatible with real-time requirements, we want to point out that our code is experimental and very little time has been spent to improve its efficiency.

Nevertheless, we believe that the algorithm could be implemented to fit into a real-time environment by using a faster machine, more sophisticated data structures and implementation techniques.

On the other hand, we want to stress that, as shown in Tables 5 and 6, the optimal solution allows saving up to about 20% on runway utilization.

Table 4. $t_{i,j}$ (sec); m = 4 1 = B 743; 2 = B 727; 3 = 707; 4 = DC 9

i \ j	1	2	3	4
1	96	181	200	228
2	72	70	100	130
3	72	70	80	110
4	72	70	80	90

Table 5. (n = 30)

Aircraft Number	Category	Nominal Landing Time (sec.)	FCFS Landing Time (sec.)	Optimal Sequence	Optimal Landing Time (sec.)
1	1	0	0	1	0
2	1	79	96	2	96
3	1	144	192	3	192
4	2	204	392	5	288
5	1	264	464	6	384
6	1	320	560	4	584
7	2	528	760	8	656
8	1	635	832	11	752
9	2	730	1032	7	952
10	2	766	1112	9	1032
11	1	790	1184	12	1104
12	1	920	1280	14	1332
13	3	1046	1461	10	1412
14	4	1106	1591	15	1492
15	2	1136	1671	16	1572
16	2	1166	1751	13	1642
17	2	1233	1831	18	1714
18	1	1642	1903	19	1810
19	1	1715	1999	20	1991
20	3	1770	2180	17	2091
21	1	2074	2252	23	2201
22	1	2168	2348	21	2273
23	4	2259	2576	22	2369
24	2	2427	2656	25	2465
25	1	2481	2728	24	2665
26	2	2679	2928	26	2745
27	3	2883	2998	27	2815
28	2	2982	3098	28	2915
29	1	3046	3170	29	2987
30	1	3091	3266	30	3083

15. Trends in ATC development

From a technological point of view, future ATC systems will be characterized by innovation in communication, navigation, surveillance and automation. The introduction of a new secondary surveillance radar, known as Mode-S, will provide digital data links in air/ground communication, with the ability to selectively address the aircraft and to transmit routine information and multiple messages. In fact, the performance of microcomputers enables digital signalprocessing with improvement of radar information quality, reduction of loading in speech channels and of interpretation errors (Ross N., 1986).

Table 6. (n = 44)

Aircraft Number	Category	Nominal Landing Time (sec.)	FCFS Landing Time (sec.)	Optimal Sequence	Optimal Landing Time (sec.)
1	1	0	0	1	0
2	1	79	96	2	96
3	2	144	296	3	296
4	2	204	376	4	376
5	2	264	456	6	448
6	1	320	528	7	544
7	1	528	624	8	640
8	1	635	720	5	840
9	2	730	920	9	920
10	1	766	992	11	1000
11	2	790	1192	13	1080
12	1	920	1264	14	1160
13	2	1046	1464	16	1240
14	2	1106	1544	17	1320
15	1	1136	1616	19	1400
16	2	1166	1816	10	1472
17	2	1226	1896	12	1568
18	1	1233	1968	15	1664
19	2	1286	2168	18	1760
20	2	1418	2248	21	1856
21	1	1642	2320	22	1952
22	1	1715	2416	24	2048
23	2	1749	2616	29	2144
24	1	1770	2688	30	2240
25	2	1809	2888	32	2236
26	2	1869	2968	34	2432
27	2	1929	3048	35	2528
28	2	1989	3128	20	2728
29	1	2074	3200	23	2808
30	1	2168	3296	25	2888
31	2	2229	3496	26	2968
32	1	2259	3568	27	3048
33	2	2326	3768	28	3128
34	1	2427	3840	31	3208
35	1	2481	3936	33	3288
36	2	2488	4136	36	3368
37	2	2565	4216	37	3448
38	2	2657	4296	38	3528
39	1	2679	4368	44	3608
40	1	2883	4464	39	3680
41	1	2982	4560	40	3776
42	1	3046	4656	41	3872
43	1	3091	4752	42	3968
44	2	3153	4952	43	4064

In the framework of automated systems, several programs have been proposed and investigated by FAA (AERA system) and Eurocontrol (ARC 2000), and an increase of automation in some functions is expected, especially for deconflicted 4D flight plans processing and for strategic management of traffic flows.

In the area of air flow control, following the network optimization approach, new models to deal with airport congestion prevention and flow management in control sectors have been developed recently and solved by nonlinear programming algorithms and supercomputing (Zenios S.A., 1991). A different approach was proposed by Andreatta G. and Romanin-Jacur (1987), based on a probabilistic model and dynamic programming algorithms in order to find optimal take-off delay policies.

Conversely, with regards to tactical control, the concept of intelligent assistance to controllers, based on advanced graphic display capabilities, is the most suitable and viable. In fact, automation does not have to take over the role of the human decision-maker, but it has to significantly reduce the controller workload, by improving monitoring functions and information presentation, issuing alerts for required actions, generating advice and providing the capability to execute routine functions automatically. On this particular topic, potential applications of Artificial Intelligence techniques have been identified and discussed (Gosling G.D., 1987). For example, expert systems can enhance safety by operating in parallel with the human controller in suggesting alternative actions with evaluation and explanation functions. In fact, the combinatorial explosion of the possible number of situations to be considered, clamors for heuristics and rule-based systems in order to find fast and feasible solutions to such problems as collision avoidance, failure management and exception handling.

16. Conclusions

The field of air traffic control is currrently characterized by increasing technology innovation toward automation and effective decision support to controllers. A new approach to the strategic planning and tactical operations of the ATC system is therefore necessary.

In particular, the multilevel model proposed in this paper shows that the ATC is well depicted by a control function with a hierarchical structure. Thus, a better balance between planning strategies and operational actions can be established. Moreover, in this framework, for each on-line function, the corresponding optimization models and computation algorithms seem to be helpful tools for better understanding traffic control phenomena, for increasing ATC system automation and, therefore, for improving air traffic management.

Acknowledgments

The authors are indebted to all colleagues mentioned in the references who contributed to earlier work.

References

AGARD 1988, *Knowledge based concepts and Artificial Intelligence applications to guidance and control.* Lecture Series n. 155.

AGARD 1990, *Aircraft trajectories: computation-prediction-control.* Agardograph n. 301, Vol. 1,2,3.

Andreatta G., Romanin-Jacur G., 1987, Aircraft flow management under congestion. *Transportation Science,* Vol. 21, n. 4, pp. 249-253.

Andreussi G., Bianco L., Cini M., Grippo L., 1981, *Un Approccio Strategico per il Controllo in Linea del Traffico Aereo.* IASI-CNR, RI-7.

Assad A.A., 1976, *Multicommodity Network Flows - Computational Experience.* Working Paper OR 058-76, Operations Research Center, MIT, Cambridge.

Bianco L., Cini M., Grippo L., 1978, *Pianificazione e Controllo Strategico del Traffico Aereo.* Istituto di Automatica - Università di Roma and CSSCCA-CNR, R. 78-28.

Bianco L., Nicoletti B., Ricciardelli S., 1978, *An Algorithm for Optimal Sequencing of Aircraft in the Near Terminal Area. Optimization Techniques* (Stoer J. ed.). Lecture Notes in Control and Information Science, Springer Verlag.

Bianco L., Cini M., Grippo L., 1979, *A Strategic Approach to Air Traffic Control.* 9th IFIP Conference on Optimization Techniques, Warsaw.

Bianco L., Ricciardelli S., Rinaldi G., Sassano A., 1979, *The Aircraft Optimal Sequencing as a N Job-One Machine Scheduling Problem.* Istituto di Automatica - Università di Roma and CSSCCA-CNR, R. 79-36.

Bianco L., Bielli M., Cini M., Grippo L., Ricciardelli S., 1980, *New Approaches to Air Traffic Control and Management.* Invited Paper to the World Conference on Transport Research. London.

Bianco L., 1985, *A Multilevel Approach to the ATC Problem: the On-Line Strategic Control of Flights.* Seminar CEE-CNR-PFT on Informatics in Air Traffic Control, Capri.

Bianco L., Ricciardelli S., Rinaldi G., Sassano A., 1988, Sequencing Tasks with Sequence Dependent Processing Times. *Naval Research Logistics Quarterly.* V. 35, pp. 177-184.

Bianco L., 1986, *Strategic Control to Improve Efficiency of Air Traffic Management.* AGARD CP-410.

Bianco L., Rinaldi G., Sassano A., 1987, A Combinatorial Optimization Approach to Aircraft Sequencing Problem. *Flow Control of Congested Networks* (Odoni A.R., Bianco L., Szegö G., eds) Springer Verlag.

Bielli M., Calicchio G., Nicoletti B., Ricciardelli S., 1980, Flow Control of Large Scale Air Traffic Systems. 2th Symposium IFAC, *Large Scale Systems* (Titli A. and Singh M.G. eds), Pergamon Press.

Bielli M., Calicchio G., Iozzia S., Nicoletti B., 1981, Air Traffic Control Systems: Modelling and Evaluation. IFAC *Control Science and Technology,* Kyoto (Akashi H. ed.) Pergamon Press.

Bielli M., Calicchio G., Nicoletti B., Ricciardelli S., 1982, The Air Traffic Flow Control Problem as an Application of Network Theory. *Computers and Operations Research,* Vol. 9, n. 4, pp. 265-278.

Dear R.G., 1976, *The Dynamic Scheduling of Aircraft in the Near Terminal Area.* FTL R76.9. Flight Transportation Laboratory, MIT Cambridge.

Erwin R.L. and others, 1974, *Strategic Control Algorithm Development.* U.S. Department of Transportation, Report n. DOT-TSC-OST 74-3, Vol. I-IV.

Erwin R.L., 1975, *Strategic Control of Terminal Area Traffic.* AGARD Conference Proceedings n. 188, Plans and Developments for Air Traffic Systems 3-1, 3-13.

Eurocontrol, 1977, *Traffic Flow Investigation System Progress Report.* International Air Transport Association. 6th Meeting of the ATM Group, Paris.

Gosling G.D., 1987, Identification of artificial intelligence applications in air traffic control. *Transp. Res. A,* Vol. 21A, n. 1, pp. 27-38.

Mesarovic M.D., Macko D., Tokamata Y., 1970, *Theory of Hierarchical Multilevel System.* Academic Press.

Odoni A.R., 1969, *An Analytical Investigation of Air Traffic in the Vicinity of Terminal Area.* Technical Report n. 46, Operations Research Center, MIT, Cambridge.

Park S.K., Straeter T.A., Hogge J.E., 1972, *An Analytic Study of Near Terminal Area Optimal Sequencing and Flow Control Techniques.* Agard CP. 105.

Park S.K., Straeter T.A., 1973, *Near Terminal Area Optimal Sequencing and Flow Control as a Mathematical Programming Problem.* Symposium on Non Linear Programming, George Washington University.

Psaraftis H.N., 1978, *A Dynamic Programming Approach to the Aircraft Sequencing Problem.* FTL R78-4, Flight Transportation Laboratory, MIT, Cambridge.

Psaraftis H.N., 1980, A Dynamic Programming Approach for Sequencing Group of Identical Jobs. *Ops. Res.,* 6.

Ratcliffe S., 1974, *Computer Representation of Air Traffic Route Structure.* Royal Radar Establishement Memorandum N. 2880, Malvern.

Ross N., 1986, Air traffic control systems. *Information Technology Applications in Transport,* Bonsall & Bell (Eds.), VNU Science Press.

Tobias L., 1972, *Automated Aircraft Scheduling Methods in the Near Terminal Area.* AIAA Paper n. 72. 120.

Trivizas D., 1985, *Optimization of Runway Operations.* Working Paper. Flight Transportation Laboratory, MIT, Cambridge.

Wismor D.A., 1971, *Optimization Methods for Large-Scale System with Applications.* Mc Graw-Hill.

Zenios S.A., 1991, Network based models for air traffic control. *European Journal of Operational Research,* Vol. 50, pp. 166-178.

SIMULATION AND OPTIMIZATION IN FLOW PLANNING AND MANAGEMENT

David E. Winer

Manager, Model Development Division
Operations Research Service
U.S. Federal Aviation Administration

This chapter deals with some of the computer tools that the FAA uses to design airspace, determine policy options, test new technologies, and improve overall management of the United States airspace system. The role of modeling in planning and supporting traffic management and training functions is examined. Additional topics are dynamic techniques, databases, and other technology for determining (and revising) optimum flow plans as the airspace system changes.
To achieve system efficiency, traffic flow planning must weigh the impact of constantly changing winds, storm patterns, scheduled and unscheduled demand, and interruptions from such causes as airport closures. Applications and functions under development for the FAA Air Traffic Control System Command Center are examined.

1. Introduction

Most of the aviation technical community is familiar with, "What if...", the typical question associated with simulation modeling. A simulation normally evaluates a series of proposed action plans, or scenarios, in which a computer program mimics reality by stepping through all interacting events as they happen. What happens depends of course on suggested strategies being modeled. By analyzing the aftermath, by being able to have a vision of what would happen should a scheme be implemented, planners can identify sensible solutions to problems. Computer simulation, by playing out a scenario to its end, can capture nearly all the complex interactions that would take place in real life, interactions too numerous to anticipate by routine thought processes. Thus figuratively, simulation provides a crystal ball view of the future that is beyond ordinary human insight.

There is yet another, more direct way to solve problems, one that attempts to answer the question, "What should I do?", without conscious iteration and testing of "What if?" proposals. In the operations research field this direct process is labeled optimization. The expression suggests that there is literally a best solution, and further, that "best" is quantifiable. Optimization routines normally fall short of the ideal goal of finding an absolutely best solutio since all possible data are not available and since there is a practical limit to precision imposed by the speed and size of computers. One can think of aviation optimization as a process involving hundreds, thousands, or even millions of "What if" tests in a search for the best schedule, design, or procedure that affects the way airplanes behave.

Whether we work with simulation or optimization techniques, these two operations research methods are maturing into extremely powerful decision support tools for designing air traffic projects and for planning and managing operational activities.

The Federal Aviation Administration (FAA) currently carries out its air traffic control mission in numerous ways that benefit from computer simulation and optimized planning. A number of initiatives that use these techniques are in progress. Successes in this area usually lead to further requirements for simulation and optimization as the reputation for effectiveness and high benefit/cost ratios associated with these techniques spreads through the user community.

2. Aircraft Performance Modeling

In order to be able to carry out simulation and optimization modeling involving flight efficiency, we developed a method to compute accurately the fuelburn of aircraft in all regions of the performance envelope [2, 3, 9]. Based on the fundamental physical principles of energy balance, the resulting equation is now capable of dealing with all phases of climb, cruise, and descent, as well as allowing for stages of gear and flap extension. This equation offers a continuous solution through any degree of resolution of its variables, and is thus particularly suitable for optimization using dynamic programming, discussed below.

The fully developed core performance equation balances potential energy, kinetic energy, thrust work, and drag work of an aircraft as it moves from one point to another. The variables considered are speed, altitude, weight, and path length in time or distance. A series of aircraft-specific constants, generated especially for the energy balance equation, is required for each type of aircraft. It is not necessary to determine cockpit oriented data such as thrust or throttle settings, an important consideration for algorithms used in ATC planning models.

Serious regard for optimum aircraft performance is crucially important for air traffic flow planning. Through fuelburn computation, developers may create real time models that realistically consider capabilities and needs of individual aircraft or fleets of aircraft: Can an aircraft actually meet climb schedules desired by ATC or realistically attain projected altitudes and speeds? What does an aircraft demand in terms of a truly fuel efficient clearance? Whether founded in simulation or optimization, models that consider performance are especially able to integrate aircraft operational efficiency in their calculations.

3. Wind and Temperature Modeling

The efficiency goal in flight planning is not to minimize the number of surface molecules flown over, but to minimize the number of air molecules flown through. This concept establishes a critical need for accurate and timely wind and temperature data to compute fuel consumption on each segment of flight considered in the optimization process [8]. The more detailed and accurate the atmospheric data available to optimization models are, the greater the possible resolution in determining the minimum fuelburn route and profile. Unfortunately, available geographical and temporal information about the changing upper atmosphere still lacks adequate precision despite years of research.

Still, some emerging technologies are promising, particularly the wind profilers developed by the National Oceanic and Atmospheric Administration [7]. These doppler radar instruments are being installed in the central U.S. under a scheme that is amenable for use in flight efficiency planning. Interestingly, this system is strikingly similar to an FAA proposal ten years ago to use profilers for improved flight planning capabilities [10]. Compared with balloon based observations, profilers' wind and temperature data can be determined within minutes instead of hours. Expansion of the system over wider geographic regions, combined with a data transmission system to promulgate wind profiles to the public, would unquestionably improve the ability of the FAA to plan for efficient flow of traffic.

The FAA is developing methods that will allow us to incorporate profiler data, or atmospheric data from any new source, with traditional forecasts for use in our traffic flow planning programs. The FAA has already developed techniques to use data not previously considered available for flight planning. These incorporate pilot reports directly in forecast products [4]. Each wind/temperature report received from an airplane is inserted automatically into a database used for efficient flow planning. The pilot report's effect is diminished gradually to zero in 200 nautical miles by a selected mathematical algorithm. These "kernels of truth", whether from profilers or from pilots, or from automatically gathered and transmitted airplane data, represent the best immediate technology for improving flight planning data in real time.

4. Visual Information Display

At some point people believed that new ATC computer display systems should emulate radar screens: these would serve controllers' needs through the presence of familiar working tools. An important developmental program in the FAA that subscribed to this concept, the Oceanic Display and Planning System (ODAPS) originally specified monochrome display screens with an appearance of ARTS radar screens, even though the aircraft position information came from non-radar sources. By contrast, in another, functionally different, oceanic program the FAA Dynamic Ocean Track System (DOTS) used as much color and resolution as was feasible at each stage of its development. The DOTS viewing screen also provided access to other verbal and numeric information not historically available for display. The DOTS visual presentation seems to have played a large part in the rapid acceptance of this system, even though its technology was founded only in commercially available computer hardware, and even though the display functions represented only a small part of the new system's planning capability.

Today's computers have changed the standards by which decision makers judge information. Not surprisingly, we now can demand that computer models reveal complex relationships graphically. Traffic flow optimization techniques seem unable to get beyond the laboratory unless they are presented in high resolution color, animated if possible. Three dimensional perspective views are even better. Fortunately, the technology to achieve such graphics is readily at hand. Unfortunately, short term budgets, lack of insight, and reluctance to change too often slow the pace of adoption of the latest graphics technologies.

Dramatic advancements in workstation class computers occur routinely as manufacturers seize upon each new electronic breakthrough. It is difficult to keep informed of the available display technology under these circumstances, much less to adopt the latest equipment. Yet costs for these advances are oftentimes moderate, and make them acceptable. Considering costs, one must keep in mind that effective display technology that promotes widespread adoption of more efficient aviation systems will likely produce benefits that far exceed the costs of the systems.

5. Optimization: Dynamic Programming

A common operations research challenge in aviation is to find the most advantageous way to fly airplanes, usually with the least expenditure of fuel or time. When confronted with this challenge, we have found that the logical first step is to reduce the infinite number of possible flight trajectories to a finite set. The existence of fixed navigational waypoints, routes, altitudes, and ATC requirements obviously curtails the number of possible alternatives that need be considered. Next, elimination of obviously inappropriate waypoints further cuts the computational load. Finally, dynamic programming methods described three decades ago [1] offer computer efficient means of evaluating a myriad of speeds, altitudes, and directions. The desired result is a plan for minimum expenditure of resources. To use dynamic programming in this manner, however, one must be able to compute fuelburn precisely throughout the aircraft performance envelopes. further, it is essential to determine the wind and temperature at all points in the matrix of navigational possibilities.

The dynamic programming process finds the path for minimum fuel or time, or a weighted combination of these, as discrete links through a series of nodes. We solve the energy balance equation described above repeatedly to identify the single path that yields the minimum total. For this application, three dimensional geographic intersections are the nodes for evaluating fuelburn for all the variables. Computing fuelburn and time for each link between nodes is a required step before sorting out the most efficient solution. It should be noted that this use of dynamic programming requires the use of a continuous function equation. That is, any incremental change in a variable such as weight or airspeed, however small, will be included in the computation. Traditional table-lookup and interpolation methods for determining aircraft fuelburn are not compatible with this dynamic programming application.

Our currently active dynamic programming techniques evolved from early FAA optimization investigations [6]. Along with computer hardware advancements, this state of optimum planning art results in convenient operational software that can accommodate sudden changes in the air traffic control situation. For example, if airports close or flight regions bar entry due to storms, we can devise highly efficient alternatives in time to consider the needs of traffic aloft.

6. Optimization: Dynamic Ocean Track System

DOTS evolved from optimization concepts, based on aircraft performance modeling, into operational prototypes to help manage oceanic air traffic (Figure 1). The current version, operating on Apollo workstations in Traffic Management Units at Oceanic Centers, began as a prototype on an Apple IIe computer, and grew quickly to exhaust the capabilities of networked IBM PCs. DOTS uses most of the technologies described above, but its principal purpose depends on dynamic programming.

This system has three major functions:

1. Track Generation uses forecast aviation weather, aircraft performance data, and site specific ATC flow requirements to produce a structured optimal route system. These tracks allow the most efficient use of oceanic airspace in a specific future period of hours.
2. Traffic Display accepts data from various flight data sources, then monitors and projects aircraft progress through oceanic airspace. The system includes logic to verify aircraft position data and to report discrepancies with other estimates. Pilot reports of position, wind, and temperature improve the DOTS weather forecasts. These reports are also used to compute and plot future aircraft positions by simple dead reckoning from reported positions.
3. Track Advisory identifies airspace availability from known and projected aircraft position data. This function provides traffic specialists with options that best accommodate users needs. These options go electronically to carrier dispatch offices for evaluation and flight plan revision.

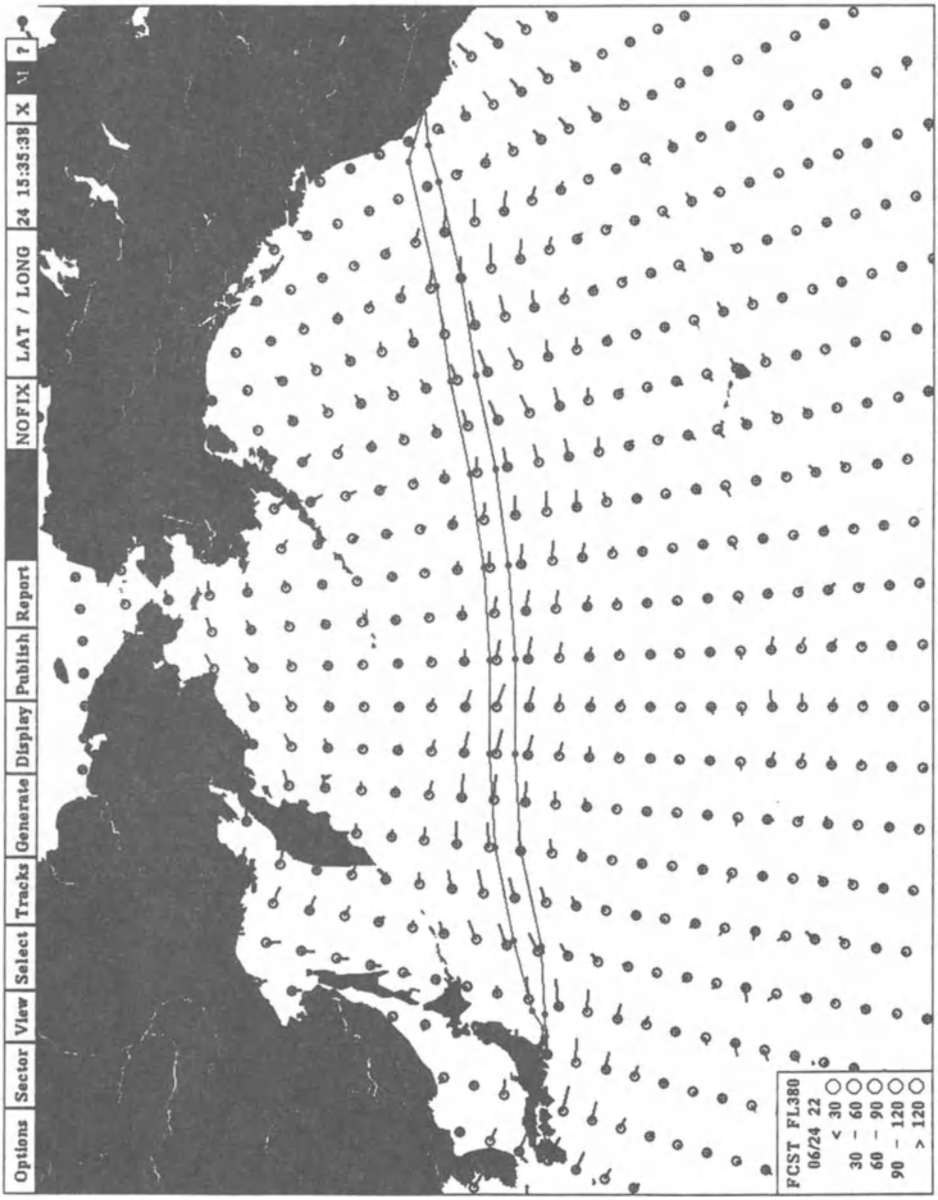


Figure 1. A selected view from the DOTS display showing fuel efficient tracks across the Pacific. Included are gridded wind vectors from the DOTS wind and temperature database used to optimize for scheduled traffic demand.

7. Optimization: High Altitude Route System

HARS is being developed as a dynamic traffic planning model that responds to user demands for improved continental U.S. (CONUS) airspace use (Figure 2). This highly automated optimizing system should soon augment, and eventually replace the "National Route Program" manual system currently used. Its goal is to provide improved fuel efficiency, decreased flight times, increased capacity, and reduced delay for long range flights. Similar in concept to the DOTS program, HARS will consider constantly changing weather and traffic demand in designing clusters of optimum routes. Functional parts of the HARS system, such as FLOWALTS described below, are made available for use by the FAA Air Traffic Control System Command Center (ATCSCC) during the time that other functions are being developed. All development is carried out in a rapid prototyping environment in which the ATCSCC personnel join with suggestions for improvement. The prototype design allows for convenient integration with DOTS and the agency's other long range traffic management programs, including especially the ATCSCC's Enhanced Traffic Management System and the rapidly developing FAA/NASA Center Tracon Automation System (CTAS).

It is premature to describe end-state functions in the HARS rapid prototyping environment. We can anticipate certain technological achievements, but cannot be sure that either the air traffic system operators or the aircraft operators will want to adopt all of these untried capabilities. Generally, the process is: install an incremental advancement for a limited trial, learn from the experience, modify the system, and move to the next increment based on new knowledge from the field.

At present, HARS combines the technology developed in the Dynamic Ocean Track System with improved weather reporting and display capabilities. The FLOWALTS technique allows planners in Central Flow to develop optimal reroutes around severe weather. As envisioned, these reroutes can then be "published" in real-time by Central Flow to the airlines to move traffic around severe weather.

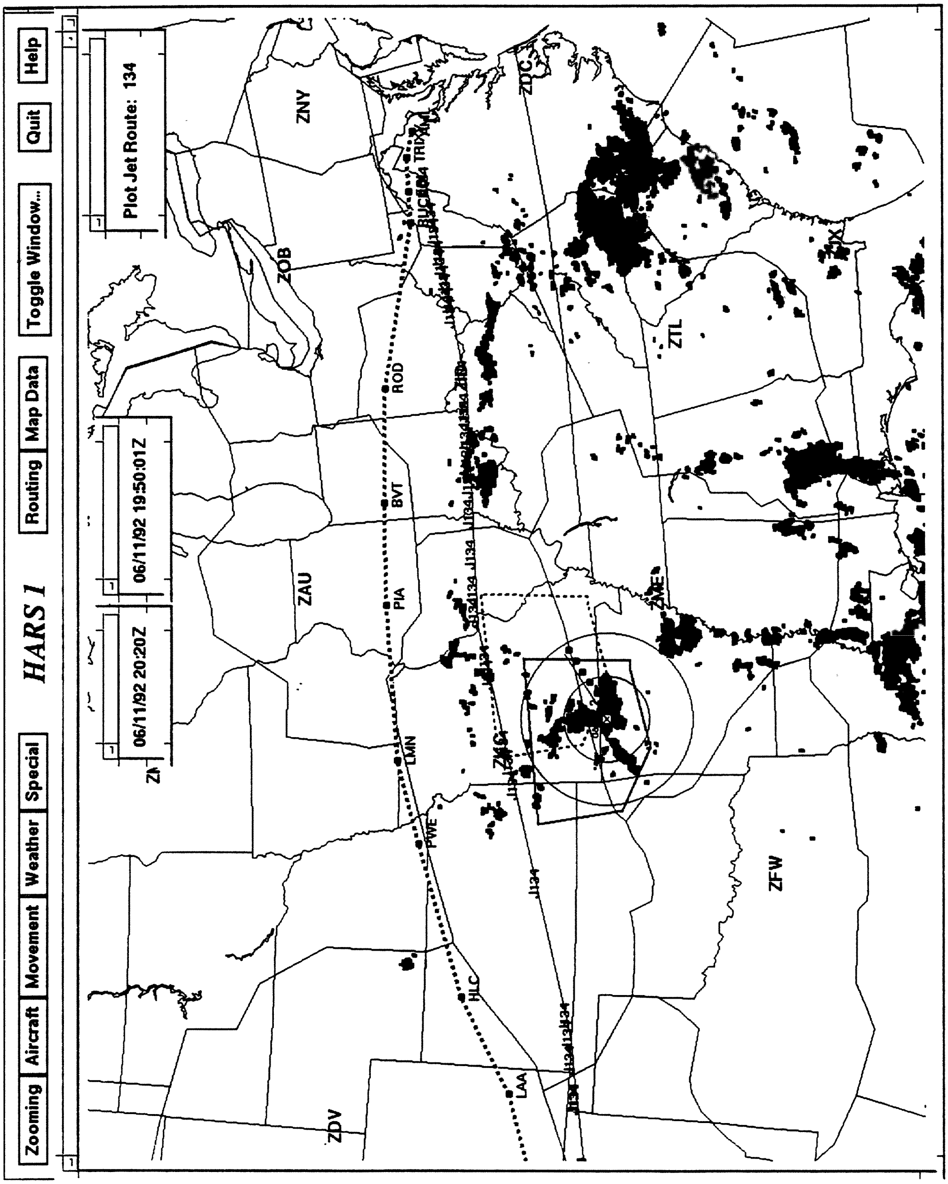


Figure 2. A screen from the HARS display. A fuel efficient re-route created by dynamic programming algorithms is to the north of the standard Jet Route 134. The HARS plan avoids the storm areas shown in black in this figure.

Current values of winds aloft data, storm data, special use airspace restrictions, traffic schedules, actual traffic load in the airway system, and the ATC separation rules are evaluated. HARS then determines (computes) and electronically publishes fuel efficient routes. HARS planning also considers flight time and capacity, and meets the scheduled demand between city pairs. Thesesystems of long range routes can be computed and distributed as many times during the day as desired.

Another important HARS capability is to review in real time the planned entry of individual airplanes into the system of routes between city pairs. HARS has information that the carriers don't have when they plan their flights: real time knowledge of where in the route system the competing traffic actually is. Thus, prior to commitment to a selected route, HARS can compare the projected flight plan with all other possible alternatives to determine which route is most efficient. The optimization algorithms take into account the individual airplane's need to climb enroute, then compare the proposed path and vertical profile solution with traffic separation requirements on all alternative routes. Although operational details have yet to be developed, each user's dispatch office will be given an opportunity to reroute most efficiently. The effect of this capability is to provide each airplane with its best altitude on a currently fuel efficient and currently conflict-free route.

HARS includes new information display functions as well as graphic information from the Enhanced Traffic Management System. At the present stage of development, most of our new information display deals with weather. We have concentrated on high resolution color coded maps that locate storms from weather radar and lightning strike data. The mapped storm positions can be forecast and further mapped for route planning.

8. Optimization: FLOWALTS

A sub-function within HARS, FLOWALTS (for flow alternatives) is a very rapidly built prototype to aid flow planners by employing previously developed technology. Highly graphical in nature, the system generates and displays alternate routes that are affected by weather, flow restrictions, or other constraints It then provides quantitative rankings of the economic impacts (fuel and time) associated with use of each route. The ranking algorithms consider specific airplane types, altitudes, user operating practices, and current weather information to determine fuelburn and flight time factors. Flow planners will use FLOWALTS graphically to delineate areas that should be avoided due to weather and other route restrictions. Levels of storm severity are displayed in intensity coded colors at a resolution of two kilometers. The technique results in a well defined picture of hazardous weather that is best to avoid in planning traffic flow.

The optimization function is invoked from a pull-down menu. Typically, the flow planner chooses regions not to fly through, either by drawing polygons around storms, or by selecting boundaries around named sectors or even entire ATC centers. Routines are being included to project the movement of enclosed storm boundaries, initially by manual entry of the speed and direction advised by ATCSCC meteorologists. Later, algorithms developed in conjunction with meteorologists will serve to automate storm projections. Once a flow specialist specifies origins and destinations of displayed jet routes that pass through undesirable regions, the computer employs dynamic programming to find fuel efficient substitutions for the jet routes. The alternate routes incorporate the effects of winds and temperatures everywhere and of course avoid entirely all airspace deemed unsuitable by ATCSCC planners.

9. Optimization: SMARTFLO

There is a corporate body of knowledge in the FAA System Command Center that is passed on to each generation of planning staff. This collective wisdom, dealing with weather and other traffic flow problems, increases with time. Our exploitation of this unwritten library depends on the vagaries of human memory and personality. SMARTFLO, an expert system computer program in the prototype stage, stores much of this "What I do routinely is what I should do" information in a database (Figure 3). It attempts to reproduce the behavior of experts who have dealt with similar problems in the past. The system will thus compare incoming information about weather, traffic demand, and other pertinent factors, with flow control strategies that have worked well before.

One can see that this is not a classic mathematical optimization scheme which evaluates all the possibilities. Rather, this method jumps right to a solution that has succeeded before in similar circumstances. It does not, and indeed cannot, consider all the possibilities. The purpose of SMARTFLO then is to provide very fast and consistent recommendations to planners. Such advice is likely to serve well in planning efficient traffic flow.

Heeding the principles of expert systems, we are developing the SMARTFLO software tools in close participation with flow management specialists. In this case particularly, the FAA's ATCSCC staff comprises an unusually able cadre of experienced operators from whom to draw the body of knowledge we wish to emulate.

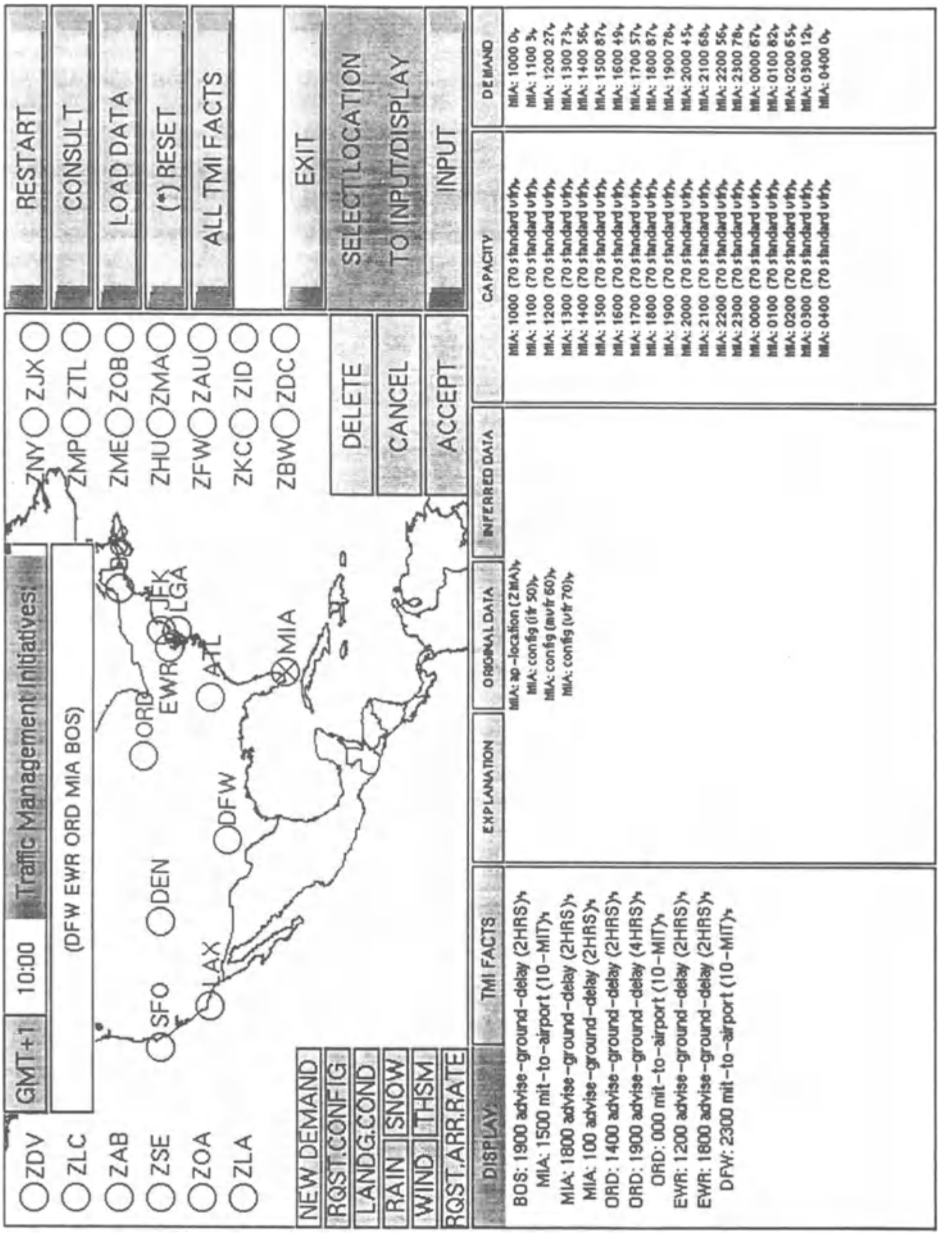


Figure 3. A preliminary version of a SMARTFLO screen. Here, the enphasis is on the Miami airport. Conditions at MIA that cause recommended traffic management advisories are listed in the center column; the advisories themselves (ground holds and miles in trail restrictions) are in the left column.

10. Simulation: FLOWSIM

The Daily Flow Simulation Model (FLOWSIM) project is developing a rapid simulation capability for traffic flow between all major US airports up to 24 hours in advance (Figure 4). The model will enable traffic flow managers to foresee traffic demands on the national air traffic system as well as allow them to evaluate the delay implications of their flow strategies. An initial version, currently restricted to a laboratory setting, is proving concepts and computation times for a rapid prototype version scheduled for installation in the ATCSCC. FLOWSIM will provide the ATCSCC with the ability to simulate, in fast-time, the day's scheduled air traffic flow, and to test various planning options. This should afford time to resolve many delay problems before they occur. FLOWSIM allows flow managers to explore the effects of alternative mixes of Estimated Departure Clearance Time (EDCT) programs and Miles in Trail (MIT) flow restrictions on traffic throughout the country. For this work, the model uses current data concerning airport configurations, airport traffic capacities, reported and forecast weather conditions, airport-specific operating procedures, and most importantly, flow management options.

The results of each scenario are graphically displayed to assist choices among alternatives.

To support real-time daily decision making, FLOWSIM was slated to run a scenario to completion in less than 15 minutes. This requirement for fast turnaround was to allow ATCSCC traffic managers to rerun and review the FLOWSIM simulation throughout the day using the latest inputs of traffic demand, weather, airport configurations, etc. In practice, the run time achieved has surpassed our goal considerably: typical scenarios finish in less than ten seconds using the fast workstation computer anticipated for use in ATCSCC.

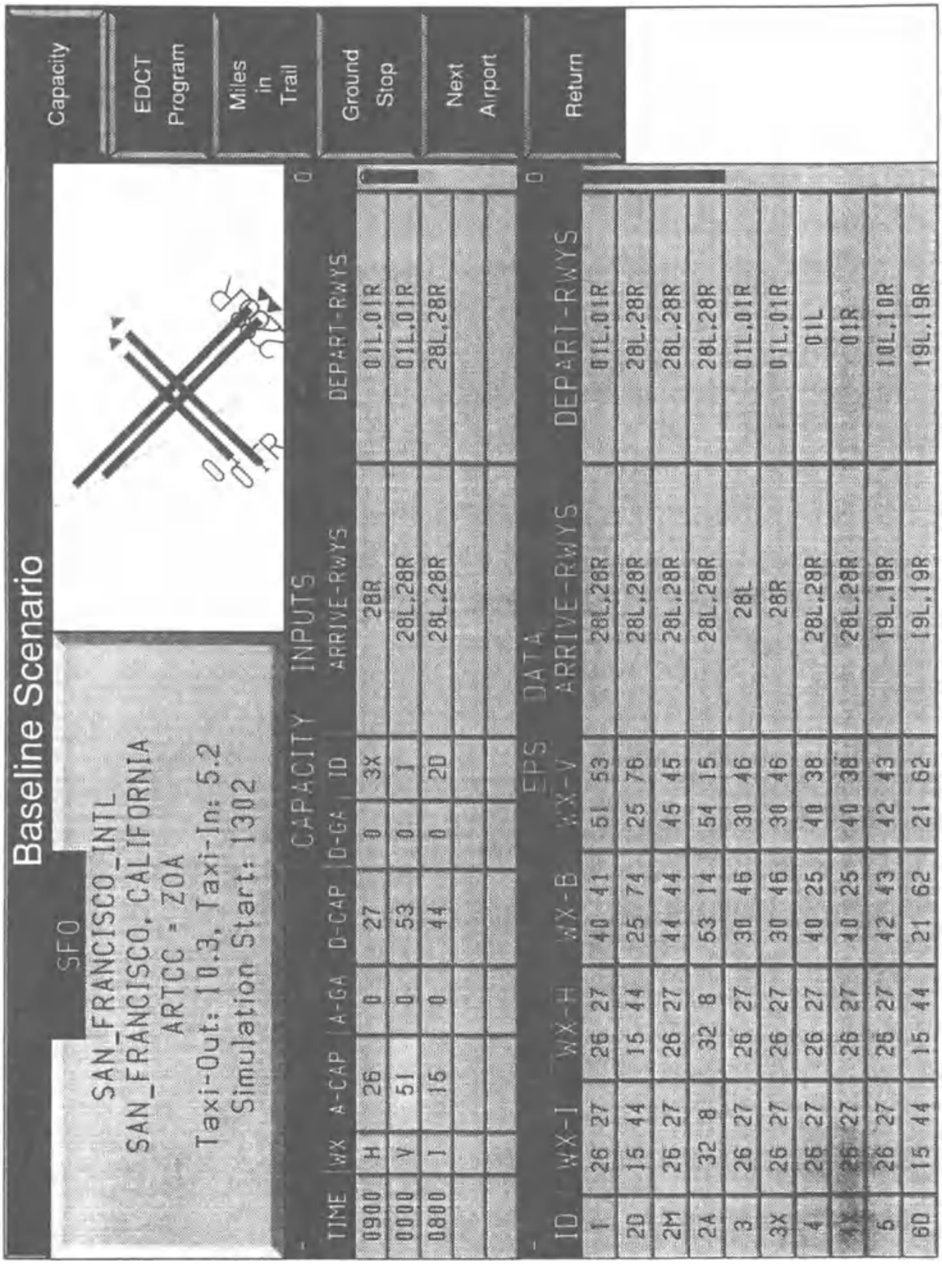


Figure 4. The user interface for FLOWSIM. In this example, the planner can review and change the runway conditions at San Francisco that affect the flow of traffic in the rest of the U.S.

11. Simulation: SIMMOD

Another FAA product that exemplifies the concept of simulation is SIMMOD, a model for detailed analysis of airport and airspace design and procedures (Figure 5). This large-scale model considers all the control options available for all airplanes as it moves the airplanes along their routes: as each plane competes for air- or ground-space ahead in its journey, the conflicts are resolved in the computer by the applicable air traffic separation rules. Just as in real life, some airplanes receive delay to maintain safe separation. SIMMOD keeps track of each element of delay for each airplane being modeled and stores this information in a large outcome file, or database. This file in turn, in a postprocessing stage, produces reports summarizing delay and other factors associated with the scenarios being analyzed.

Governmental and corporate users can evaluate runway designs, airspace routes, fleet schedules, air and ground procedures, fleet operating policies, and sector design; in fact, virtually any proposal that affects the movement of aircraft from parking gates through enroute airspace. The answers to "what if" questions describe the impacts of delay, travel time, capacity, fuel consumption, and noise. In effect, SIMMOD summarizes all factors that affect the economics of a proposed scenario.

The recent SIMMOD Version 2.0 is adapted to the powerful workstation class computers and makes use of advanced techniques to preprocess scenarios for efficient preparation of input files, and to postprocess the results of simulations to show high resolution animated graphics of the aircraft and aviation systems being modeled. Rule based, event-step models such as SIMMOD are ordinarily intended for studies of futures on the order of years or months away. However, recent advances in parallel and distributed processing technology have opened up the possibilities for such rapid simulation that large complicated SIMMOD scenarios can be run in time to assess daily traffic flow situations brought on by changes in weather or other flow management situations. We are participating in experiments that show abilities to consider such large networks as 50 major airports

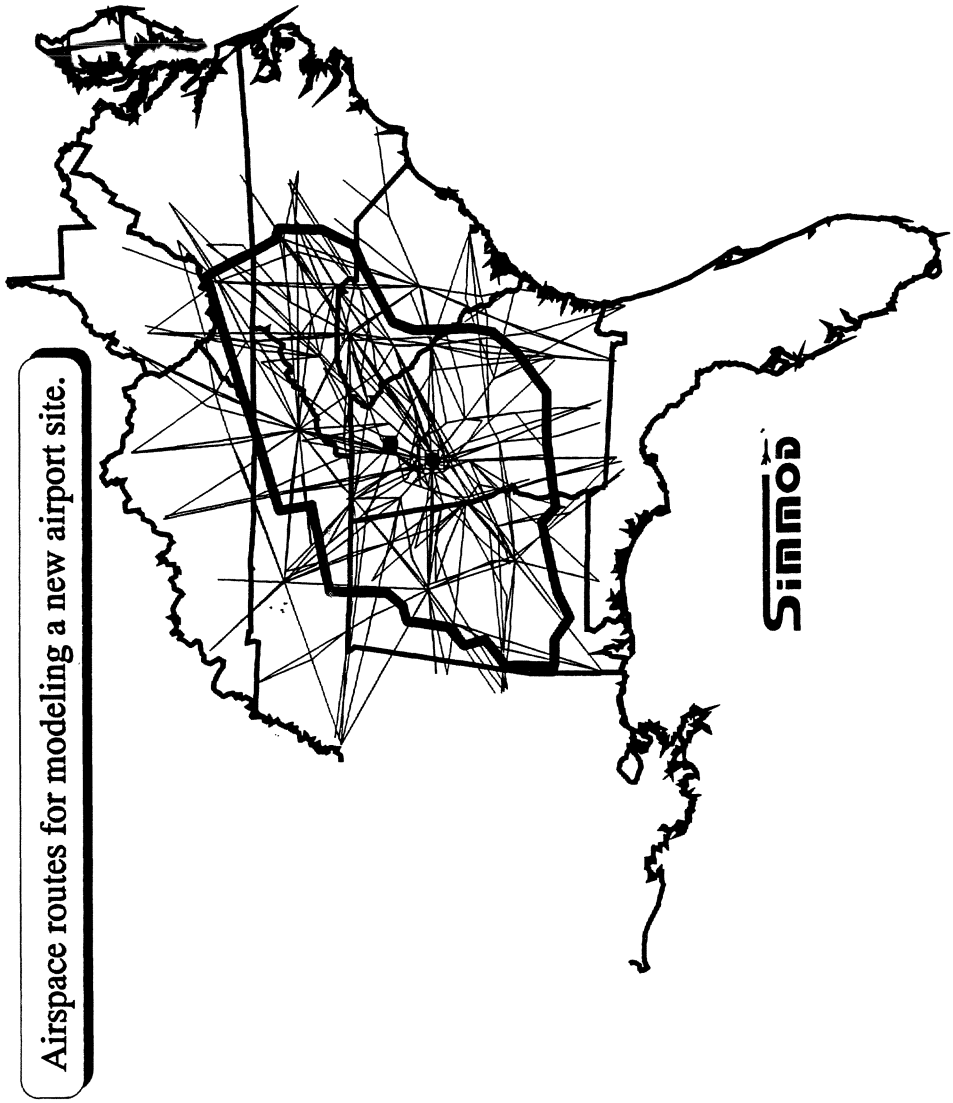


Figure 5. The airspace extending around Atlanta from a SIMMOD study. Here, the model was used to evaluate the site of a possible new airport; this concept requires re-design of airspace in the entire Atlanta Center and beyond in order to accomodate the flow of traffic.

interconnected in a simulation [5]. Such computer performance could lead to an early prototype system, using or based on SIMMOD, for use by national flow managers on a daily basis.

12. Optimization with Simulation: OPTIFLOW and NASSIM

Our blueprint for operational traffic flow planning includes even more sophisticated models. SMARTFLO and FLOWSIM, previously described, are expected to perform as fast-response models that can be developed quickly. They are not intended to provide highly detailed planning information. Their speed in reaction to flow management problems will likely continue to prove valuable even in the future when more ambitious and sophisticated programs become available to address traffic flow problems.

Our planned OPTIFLOW function may use dynamic programming but is likely to require some other effective optimization technologies, perhaps in combination. It will consider the requirements of aircraft performance modeling in planning for national traffic flows. In fact, this model is envisioned as the most sophisticated optimization system yet developed by the FAA: It will integrate and extend all technologies previously developed by the Operations Research Service. Its use is expected to decrease flight time and fuel consumption throughout the national airspace. Optimization analysis will include these variables for their interacting effects: traffic demand, flight plans, actual aircraft positions, hazardous weather patterns, aircraft performance by type, prevailing ATC facility limitations, and special use airspace availability.

OPTIFLOW will acquire much of its incoming data automatically by links to on-line databases containing information about weather, flight plans, airline schedules, and special use airspace restrictions. However, the system will also include person-to-machine interactive capabilities to help flow planners define special optimization problems posed by sudden closures of airfields, unusually heavy traffic loads, and other "pop- up" circumstances.

Proposed optimal solutions are expected to be evaluated further by playing out their recommendations in simulation. To carry out this function OPTIFLOW will formulate its best plan automatically as an input scenario to the National Airspace System Model, NASSIM. This is a detailed model for forecasting impacts of schedules and planning strategies throughout the United States. NASSIM, planned form development in 1993 and 1994, is intended to provide ATC with a complete picture of traffic delays, controller workload, and other impacts for hours ahead throughout the United States.

13. Simulation: NASSIM

This model is planned to be the next generation predictive tool for operational traffic flow planning. Its detailed traffic simulation will capture the interactions of individual aircraft throughout the continental U.S. NASSIM will be a "capstone" flow planning tool, incorporating, integrating, and extending many technologies, algorithms, data files and tools developed and proven in preceding projects such as FLOWSIM, HARS, and SMARTFLO. The model will be developed for continuous, real-time operation, and will be automatically refreshed with the latest available NAS data to achieve maximum possible realism and accuracy.

This amount of data, and the intended degree of simulation resolution, are challenging requirements. We are exploring the technology in computer hardware and languages to provide the highest level of performance possible for the model's users. The current expectation is that the model will require modern parallel processing techniques to predict the effects of the rapidly changing national airspace system.

NASSIM will dynamically forecast flow capacities for en route sectors, terminal areas, and airports. Principal driving forces will be projected system changes due to weather conditions, runway closures, airport configuration changes, and existing and proposed ATC traffic management programs. Insofar as controller workload constrains the system, the model will dynamically induce the required delay at airports and en route sectors according to accepted procedures. This means that NASSIM will not only determine ATC workload as an output parameter, but that delay "programs" or strategies for coping with excess workload must be fed back to the current scenario in order to maintain realistic simulation. Each aircraft's transit times will be recorded in detail. Delay statistics for all flights can then be accumulated and sorted by their impacts on various segments of the aviation community.

Finally, NASSIM will be required to provide replay and analysis of historical system data, and to operate on the basis of forecast or hypothetical system data. Thus, the model will serve as a policy, technology assessment, or procedures analysis tool.

14. Tying It All Together: A Common Platform and the CONDAT Database

Those familiar with the operational environment in "Central Flow" (ATCSCC) or any working ATC facility, will recognize that the specialists on the floor are not apt to embrace tools, however well intentioned, unless the tools are easy to use. In this regard, each of the programs described above could pose a separate problem in training and in coordination unless all are packaged as a single system. We originally believed that it would suffice to maintain a similar look and feel among the programs, but have come to realize that, for the system of tools to be acceptable, there really can be only a single look and feel.

Our strategy calls for a development team lead organization to coordinate user interface design among the various traffic flow planning tool developers. Under this concept, all of the programs are housed in the same computer system and can be accessed from standardized pull-down menus. The menu style is already familiar to anyone who uses any of the popular windowing operating environments, a factor that we trust will make new users comfortable from the start.

The various programs share much data in common. Interestingly, the output from some is needed as input data to others. All of these data are kept in one place, on an advanced technical database management system, we have named CONDAT (for Continental U.S. Data.) Data available through electronic communications networks, such as current weather reports and aircraft positions, load into CONDAT automatically. Other non-instantaneous data such as the Official Airline Guide are updated periodically. Users of a functional subsystem, for example FLOWSIM or SMARTFLO, may edit manually databases, such as the file for current airport configurations, from the keyboard or by mouse. Selected traffic flow plans from the optimization programs are stored as scenarios for simulation programs.

Even though our Operational Traffic Flow Planning models are developed to meet today's needs, we expect the technology, and perhaps even the modeled functions, will serve well in longer range FAA development programs. We particularly strive to establish and maintain compatibility with the FAA's Advanced Traffic Management System (ATMS) and the Terminal ATC Automation program (TATCA) [11]. Consequently, we arrange close liaison with those systems' developers with respect to software and hardware designs. We anticipate that the long range traffic management programs will, as we have chosen, evolve in the direction of Open Systems, in which the workstation computer architecture and operating systems are standardized. Should this occur, the programs described in this review can be treated readily as modules for incorporation in the upcoming systems of the future.

In some cases our traffic planning methodology will likely be integrated with other FAA research and development programs because the technology has been thoroughly tested and found advantageous. Interestingly, many of the capabilities will have evolved to meet needs that have also evolved since the time when the requirements for long range systems were written. In other cases, it seems likely that air traffic planners will embrace these models for the unique functionality derived from their operations research heritage: highly relevant mathematics and science applied to operational problems.

References

[1] Bellman, R.E. and Dreyfus, S.E., "Optimum Trajectories", *Applied Dynamic Programming*, Chapter 8, Princeton University Press, 1962.

[2] Collins, B.P., "*Estimation of Fuel Consumption of Commercial Jet Aircraft from Path Profile Data*", Society of Automotive Engineers, Paper 800746, International Air Transportation Meeting, Cincinnati, Ohio, May 20-22, 1980.

[3] Collins, B.P., Ford, D.W., and Bell, N.J., "*Concepts for Aviation Fuel Efficiency*", Proceedings of the Aviation Fuel Conservation Symposium, FAA, Washington, DC, September 10-11 1984.

[4] Collins, B. P., and Manning, W., "*Dynamic Traffic Flow Planning for ATC and Airspace User Requirements*", 1991 Winter Simulation Conference, Phoenix, Arizona, December 8-11, 1991.

[5] Cosby, Mark, "*Breakthroughs in the Application of Simulation to Aviation Planning*", Simulation Conference 17 (CACI, Federal Aviation Administration, Society for Computer Simulation), Arlington, Virginia, August 25-27, 1992.

[6] Ford, D.W., Collins, B.P., and Bell, N.J., "*Fuel Efficient Operations in the Oceanic Area*", Proceedings of the Aviation Fuel Conservation Symposium, FAA, Washington, DC, September 10-11 1984.

[7] FSL Forum, "*Profiler Network Status*", NOAA/ERL Forecast System Laboratory, Boulder, Colorado, February, 1991.

[8] True, H.C. and Winer, D.E., "*Meteorological Data Requirements for Fuel Efficient Flight*", Ninth Conference on Aerospace and Aeronautical Meteorology, Omaha, Nebraska, June 6-9, 1983.

[9] Winer, D.E., and Hoch, C.J., "*Energy Conservation in Terminal Airspace through Fuel Consumption Modeling*", Society of Automotive Engineers, Paper 800745, International Air

Transportation Meeting, Cincinnati, Ohio, May 20-22, 1980.

[10] Winer, D.E., "*Existing Wind Observation Network*", Sixth Annual Workshop on Meteorological Inputs to Aviation Systems, University of Tennessee Space Institute, Tullahoma, Tennessee, October 27, 1982.

[11] The Federal Aviation Administration Plan for Research, Engineering and Development, Chapter 2.1, "*Capacity and ATM Technology Project*"

MODELS FOR THE GROUND HOLDING PROBLEM

Giovanni Andreatta

Dept. of Pure and Applied Mathematics - University of Padova - Italy

Amedeo R. Odoni

Dept. of Aeronautics and Astronautics and Dept. of Civil Engineering - Massachusetts Institute of Technology - U.S.A.

Octavio Richetta

Dept. of Management Science and Information Systems- University of Massachusetts at Boston - U.S.A.

Airport congestion is one of the main causes of costly aircraft delays. Sometimes costs may be reduced by imposing on some aircraft a delay at take-off time in order to avoid a more expensive airborne delay later. The GHP (Ground Holding Problem) is concerned with finding optimal ground-holding strategies so that the total expected cost of delay is minimized. This paper presents a survey of models and algorithms for the GHP. These models have in common the simplifying assumption that congestion may arise at a single airport. A summary of an extensive set of computational experiments performed with these models is also presented. The results indicate that the models could provide the foundation for the development of real-time decision support systems for ATC flow management.

1. Introduction

Traffic congestion has become pervasive in the most developed air transportation systems in the world, i.e., those of North America, Western Europe and East Asia. While the available statistics on the magnitude of air traffic delays are not very precise, there is no doubt that the total direct and indirect costs of these delays are very large. If one combines the various numbers reported, worldwide direct airline costs may add up to $5 billion or more per year. By comparison the world's scheduled airlines suffered in 1990, one of the worst years in aviation history in economic terms, a total net loss of $4.3 billion. An equally telling statistic is that ground-holding delays (please see below for a definition) to airline aircraft in the United States averaged 2000 hours per day in 1986, approximately equivalent to grounding the entire fleet (250 aircraft at that time) of Delta Airlines, the third largest carrier in the United States. With significant increases in demand expected to continue, congestion is now cited by many experts and organizations as one of the principal impediments to the future growth of the airline industry.

Congestion occurs whenever the capacity of airport runway systems or of ATC sectors is exceeded over a period of time. Thus, it is mostly associated with peak traffic hours of the day, peak travel times in the year and, especially, periods of poor weather conditions when airport service rates can be significantly reduced. In the absence of the long-term capacity improvements that can be obtained through the construction of additional runways or through advances in ATC, flow management is the best available way to reduce the cost of delays. On a day-to-day basis, ATC flow management attempts to "match", dynamically, air traffic demand with the capacity of airports and airspace sectors of the ATC system. Ground-holding is the principal device used for this purpose, whenever delays are anticipated to be severe.

Ground-holding (or "gate-holding" or "ground-stopping") is typically imposed on aircraft flying to congested airports or scheduled to traverse congested airspace. Ground-holding is the action of delaying take-off beyond a flight's scheduled departure time. The motivation for doing so is that, as long as a delay is unavoidable, it is safer and less costly for the flight to absorb this delay on the ground before take-off, rather than in the air.

Prior to 1981, ground-holding was used sparingly in ATC, i.e., except for unusual circumstances, flights were allowed to take-off as soon as ready to do so. If delays were encountered, they were "taken" while the aircraft was airborne, typically by circling in the air ("stacking") near the airport of destination. However, during the 1981 air traffic controllers' strike in the United States, widespread use of ground-holding began, primarily because this was seen as a way to reduce controller workload by limiting the number of aircraft which were airborne at any given time. When it was perceived that ground-holding was also a fuel-saving practice, its use became permanent, while also being extensively adopted in Europe.

Unfortunately, deciding how much ground-holding delay to assign to a flight is far from simple, because it is difficult to predict how much delay a flight will actually suffer. The reason is that sector capacities and, especially, airport capacities are often highly variable and can change dramatically over time, as weather changes or other events occur. For example, it is not unusual in the United States to encounter a 2:1 or even 3:1 ratio between the highest and lowest capacities of a given airport. Moreover, small changes in visibility or in the height of the cloud-cover may translate into large differences in airport capacity. It is nearly impossible for meteorologists today to predict these changes to such a high level of accuracy, even over a very short time-horizon of an hour or less. Thus, ground-holding decisions must be made under uncertainty and must consider the trade-off between "conservative" strategies that may at times assign excessive ground-holds and more "liberal" ones that may result in more expensive airborne delays. Many airline representatives have complained recently that today's ATC systems may be erring too much on the conservative side, i.e., that there may be too many instances in which airport landing capacity (at destination airports) is being wasted, while aircraft stand waiting on the ground (at the airports of origin).

In this paper we shall present a sequence of models that we and our colleagues have developed for the Ground-Holding Problem (GHP), defined as the problem of assigning ground-holding delays to each of a set of aircraft in order to minimize the total (ground plus airborne) delay cost (Odoni [6]). The problem will be approached in the context of developing a real-time Decision Support System (DSS) for use by a central facility that oversees ATC flow management in a large geographical region. This context will be described in Section 2. Sections 3 discusses the common features and assumptions of the various models. The models themselves are presented in Sections 4 through 9, in a roughly chronological order. For each model, we outline the underlying assumptions, applicability, solution approaches and computational characteristics. Section 10 contains a quite detailed description of an extensive set of computational experiments performed with the last three models in the paper. These experiments underscore the potential effectiveness of the models as decision support tools. Conclusions and topics for further research are summarized in Section 11.

2. Real-Time Decision Support for Solving the GHP

In the United States, the ATC System Command Center (ATCSCC) in Washington initiates, develops, co-ordinates and monitors the execution of flow management strategies (including ground-holding decisions) for the entire country and is supported by regional "traffic management units" at a local, more tactical level. ATCSCC is equipped with outstanding capabilities for acquiring traffic-related information, including the Aircraft Situation Display (ASD), a real-time data processing and display system that shows the status and location of every airborne aircraft with a filed flight plan in the United States, along with real-time weather maps, anticipated traffic loads at ATC sectors and airports, etc. The ATCSCC currently relies almost exclusively on the expertise of its controllers for decisions concerning flow management, including ground-holding.

In Western Europe, where the problem of air traffic delays has reached the proportions of a crisis, there is no existing counterpart to a central co-ordinating facility such as the ATCSCC. A decision has now been made to establish such a facility in Brussels by 1994, with Eurocontrol acting as the organization leading this effort.

In view of the current state of affairs, the ATCSCC and the planned European facility in Brussels will certainly require the development in the future of a large variety of computer-based decision support tools. In particular, there is general agreement on the need for better real-time DSS and for fast algorithms that would assist ATC flow management specialists in devising strategies for ground holding. The objective of the research described here and of parallel efforts elsewhere is to make a contribution in this direction. Given the practical importance of the GHP, remarkably little has been accomplished along these lines to date.

Desirable Model Characteristics

Based on the preceding discussion, it is clear that, in order to properly address the GHP in its fullest generality, a model/DSS should have the following characteristics:

(i) Multi-period: Airport and ATC sector capacity changes over time and so does demand for landings and for take-offs so the dimension of time is essential; in keeping with planning practices in flow management, the time horizon (typically one day from early morning to late night) in the models of interest is subdivided into smaller periods of time (typically 5 to 15 minutes long) each of which can be viewed as relatively homogeneous with respect to demand and capacity characteristics;

(ii) Dynamic: Information is constantly being updated in the ATC system; for example (and of special relevance to flow management) weather forecasts and therefore airport capacity predictions improve as we approach any given period of time. To take full advantage, one should adopt dynamic decision strategies, which may be continually revised as new information becomes available;

(iii) Stochastic: As noted earlier, the capacity of many important ATC elements is inherently stochastic, due to its dependence on often unpredictable or partly predictable variables such as detailed weather characteristics. ATC demand is also partly stochastic on time horizons of more than one hour. However, because the level of uncertainty for demand is lower than that for capacity, at least in the short run, demand will be assumed to be deterministically known throughout this paper.

(iv) Multi-airport: A ground-hold imposed on a given flight at a given airport may also have consequences for other flights: not only are flights which were to use the same aircraft later on affected, but other connecting flights, as well. The latter is a particularly serious concern in the case of airlines that emphasize tightly interconnected hub-and-spoke networks.

In this paper, we shall review only models dealing with the single airport GHP. The reason is that, on one hand, work on the multi-airport GHP is beginning only now (Wang [13] and, especially, Vranas [12]) and, on the other, the single-airport GHP can be considered as an essential building block for the more complex multi-airport version of the problem.

The most general version of the single-airport GHP (multi-period, dynamic and stochastic) may not be appropriate for some practical situations. Depending on how much information is available and on how this information is updated, alternative versions of the GHP may have to be solved. Therefore, several versions of the problem will be addressed. For example, deterministic (rather than stochastic) versions will be preferable for locations where either the weather or the airport capacities are stable enough to be approximated as perfectly predictable quantities. Moreover, it should be noted that existing ATC systems never deal explicitly with "probabilities" and thus deterministic models may approximate today's practice better than stochastic ones. Similarly, static (rather than dynamic) versions may be more appropriate for environments where (i) there are significant lags in updating information concerning weather or capacities at a set of geographically dispersed locations or (ii) a "strategic" ground-holding plan is prepared at a single point in time (typically at the beginning of the day) and that plan is revised only in a marginal way ("tactical" changes) from that point on. Finally, even a single-period model may be useful in illustrating certain conceptual themes fundamental to ground-holding strategies under stochastic conditions. In fact the first model to be described below (Section 4) is a single-period one.

3. The Simplified GHP Model

The air traffic network model considered for the single-airport GHP can be described in reference to the single-destination network shown in Figure 1. The model is macroscopic in nature, yet it captures the essential elements needed to solve the GHP:

(i) N aircraft (flights $F_1 ..., F_N$) are scheduled to arrive at the "arrivals" airport Z from the "departures" airports.

(ii) Airport Z is the only capacitated element of the network and thus the only source of delays. All other elements in the network (departure runways, airways, etc.) have unlimited capacity.

(iii) The departure and travel times of each aircraft are deterministic and known in advance.

(iv) The time interval of interest is [0, B], with the earliest departure for Z scheduled at 0 and the latest arrival scheduled at B. The time interval [0, B] is discretized into T equal time periods numbered $1,2,...T$.

(v) Ground and air delay cost functions for each flight are known.

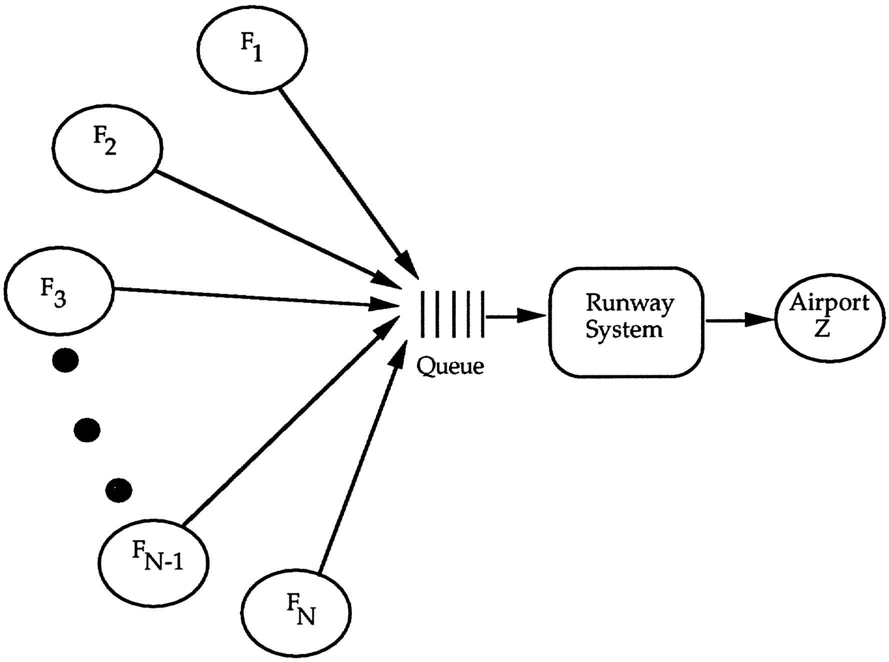


Figure 1: "Star" Configuration Network

Items (ii) and (iii) amount to an assumption of perfectly predictable travel times between the airport of departure of each flight and airport Z (for additional details see Odoni [6]). Thus, this ignores the effect of such tactical actions as speed control, "metering" and path-stretching that may sometimes take place during the "en route" portion of a flight in response to local ATC conditions. It is assumed that the impact of such actions on operating costs is entirely secondary to that of the delays due to congestion at the airport of destination Z - and, thus, to the ground-hold versus air delay trade-off. This is fully justified for the United States ATC system: although no specific statistics are collected on the matter, an overwhelming proportion of delays (possibly in the order of 95%) are undoubtedly due to airport, not en route, capacity limitations. This is not the case, however, in Western Europe, where the en route airspace imposes just as severe capacity constraints as the airports - primarily as a result of institutional and political factors. We shall return to this particular point in the final section.

4. Single-Period Probabilistic GHP

Andreatta and Romanin-Jacur [1, 2, 3, 4] seem to have been the first to attempt to solve a probabilistic version of the GHP. They considered a greatly simplified model which highlights some important conceptual aspects of all probabilistic versions of the GHP. This model assumes that congestion may arise only during a (single) given period of time at the specified "arrival" airport Z. The model therefore is neither multi-period nor dynamic, but takes explicitly into account the stochastic nature of airport capacity.

The GHP addressed can be mathematically stated as follows:

Model A

$$\text{MIN} \left\{ \sum_{i=1}^{N} C_i^g \; X_i + \sum_{q=0}^{Q} P_q \; \sum_{i=1}^{N} C_i^a (1- X_i) \; f\,(i,q \mid X_1,\ldots,X_N) \right\}$$

where:

N	is the number of flights scheduled to land;
C_i^g	is the "cost" suffered by aircraft i when held on the ground (necessarily for exactly 1 period);
C_i^a	is the "cost" suffered by aircraft i when delayed in the air (necessarily for exactly 1 period);
Q	is the maximum value of the airport capacity;
P_q	is the probability that the capacity assumes value q;
X_i	is the decision variable concerning aircraft i: X_i=1 if a ground-hold is imposed on aircraft i whereas X_i=0 if no ground-hold is imposed on it;
$f(i,q \mid X_1,\ldots,X_N)$	is a function that indicates if aircraft i will suffer an airborne delay ($f(i,q \mid X_1,\ldots,X_N)=1$) or not ($f(i,q \mid X_1,\ldots,X_N)=0$) given that the capacity is q and the decision variables take values $X_1,\ldots,X_N$.

Two different algorithms were proposed for this GHP, both using a Dynamic Programming (DP) approach that finds an exact optimal policy in $O(N^2)$ computational time. This is a generalizable result: DP-based approaches can be used, at least in theory, to find exact solutions for multi-period versions of the stochastic GHP, both for static (Terrab [10]) and for dynamic models (Richetta [9]).

The costs C_i^g and C_i^a may be arbitrary and the optimal solution depends upon any given priority rule that determines the order of landings when two or more aircraft are airborne in the terminal area simultaneously. If such a priority rule is not given *a priori*, but may be arbitrarily selected, then it is shown in [3] that landing priority should be given to the aircraft with the highest airborne delay costs. If no priority rule is being used and landings occur in a random order (as is the case in practice when landings are authorized in a first-come, first-served manner) then $f(i,q|X_1,\ldots,X_N)$ represents the probability that aircraft i will suffer an airborne delay given that the capacity is q and the decision variables take values $X_1,\ldots,X_N$.

Define $H=N-X_1-\ldots-X_N$. We then have:

$f(i,q|X_1,\ldots,X_N) = f(q|X_1,\ldots,X_N) = \frac{H-q}{H}$ for $H>q$, and $f(i,q|X_1,\ldots,X_N)=0$ otherwise. Once again, analogous patterns to the one suggested by this last statement can also be found in optimal solutions to multi-period static problems (Terrab [10]). Such patterns can also be used to increase the efficiency of heuristic algorithms for the GHP [2, 10].

5. Multi-Period Deterministic Model

The simplest multi-period model [10, 11] assumes that the capacity of airport Z is a deterministic function of time, known in advance for the entire period of interest. The time horizon consists of T periods in which capacity may be limited (and take on a different value K_j in period j) and an extra period, $T+1$, in which capacity is large enough so that all the aircraft that are still in the air can land during $T+1$. This assumption about period $T+1$ is, of course, true in practice: late at night there is always enough capacity to accommodate all the remaining requests for landings. The model can be formulated mathematically as follows:

Model B

$$\text{MIN} \sum_{i=1}^{N} \sum_{J=t(i)}^{T+1} G_{ij} X_{ij}$$

subject to:

$$\sum_{J=t(i)}^{T+1} X_{ij} = 1 \quad \forall i \in \{1,\ldots,N\}$$

$$\sum_{i=1}^{N} X_{ij} \leq K_j \qquad \forall j \in \{1,\ldots,T\}$$

$$X_{ij} \in \{0,1\}$$

where:

N is the number of flights scheduled to land;

T is the number of periods when capacity may be limited;

K_j is the capacity in period j;

X_{ij} are the decision variables: $X_{ij}=1$ means that aircraft i will be assigned to land in period j and $X_{ij} = 0$ otherwise;

G_{ij} is the cost incurred by aircraft i when assigned to land in period j;

$t(i)$ is the period of time during which aircraft i was originally scheduled to land.

The following two important observations can be made:

(i) If the capacity K_j is known with certainty for all $j=1,\ldots,T$ (as is the case in this deterministic model), then, under any optimal ground holding policy, aircraft may suffer ground holds but never airborne delays. As long as the cost of ground delay is less than the cost of airborne delay per unit of time for any given aircraft, it will always be better to hold an airplane on the ground rather than in the air. For this reason the relevant costs G_{ij} in Model B are the ground-holding costs for flight i; for the deterministic GHP, detailed information on airborne delay costs is not necessary.

(ii) The above formulation proves that this version of the GHP is of polynomial computational complexity: in fact, Model B is a particular case of the so-called "Transportation Problem" and can be transformed into an "Assignment Problem". Many numerical instances of Model B have been solved using standard Minimum Cost Flow algorithms [10].

The experimental results reported in [10] show that, even with a deterministic knowledge of airport capacity, large savings in total delay costs can be achieved through solutions that assign available capacity optimally. However, the implementability of such optimal strategies is made questionable by the systematic biases they exhibit: typically, they assign most ground-holds to aircraft with low delay costs per unit of time (e.g., general aviation aircraft and regional airlines) while giving priority to aircraft with high delay costs (e.g., wide-body aircraft). It may therefore be necessary to impose additional constraints that force a more equitable distribution of ground-holds. This can be done quite effectively and still yield strategies with significant savings in the cost of delay (Terrab [10]).

It is also important to anticipate at this point one of the major results in this paper by noting that large savings can also be achieved, <u>without discriminating at all</u> among various types of aircraft, through models that are dynamic and take into consideration the uncertainties (stochasticity) that may be present. This will be demonstrated in Section 10 through many examples in which all aircraft have identical ground-holding unit costs and identical airborne delay unit costs ("single class of aircraft").

6. A "Fine-Grain" Stochastic (Static) Model

At the next level of complexity is a multi-period, stochastic and static version of the GHP (Terrab [10] and Terrab and Odoni [11]). The capacity of the airport Z is assumed to be a random variable with a probability distribution known before the beginning of period 1. Without loss of generality, it is also assumed that there is some given *a priori* priority rule for the order in which aircraft will be assigned to the T landing periods in airport Z. (An additional period T+1 is still assumed to have unlimited capacity, as in Model B.) Then the objective function becomes the following:

<u>Model C</u>

$$\text{MIN}\left\{\sum_{i=1}^{N} EC_i\,(X_i\,, H_i^1\,, H_i^2, \ldots, H_i^T)\right\}$$

where:

X_i is the decision variable and represents the number of ground-hold periods of delay imposed on flight i;

H_i^j represents the number of flights with *a priori* priority higher than flight i already assigned to arrive in period j: obviously H_i^j depends only on those X_k such that flight k has priority over i;

$EC_i(.)$ is the expected cost to be incurred by aircraft i if decision X_i is taken, given that in period $1,2,\ldots,T$ there are respectively $H_i^1\,, H_i^2, \ldots, H_i^T$ aircraft with priority over i assigned to land.

The details of Model C can be found in [9]. There are two points, however, that need to be stressed here: First, Model C optimizes in the expected value sense, taking into consideration the entire joint probability distribution of the capacities K_j for $j = 1,2,\ldots T$. This means that, even in an optimal solution, some aircraft may have to suffer airborne delays, in addition to the assigned ground-holds, X_i, computed by solving Model C. (Of course, the higher the cost of airborne delay per unit time is by comparison to the cost of ground delay per unit time, the more "conservative" the optimal values of the X_i will be, in order to reduce the chance of costly airborne delays.) Thus Model C requires both ground-delay and airborne-delay cost functions for each flight as "inputs".

Second, this version of the GHP, may be solved through a Dynamic Programming approach, but requires a computational time of the order $O[N(M+1)^T]$, where M denotes the maximum possible value of the airport capacity in any single period. For instance, in a realistic case with N=500 flights scheduled to arrive over T=50 periods of time and assuming that the maximum capacity is $M=N/T=10$, the exponential factor $(M+1)^T$ is equal to $(11)^{50}$. The conclusion is that the DP approach is impractical for real problems, without further refinements. Note that the exponential factor $(M+1)^T$ is equal to the total number of possible different capacity profiles, if we assume that in any period of time all capacity values from 0 to M are possible. However, it is quite unlikely that a capacity profile with, for instance, capacity equal to 0 in the odd-numbered periods and capacity equal to M in the even-numbered periods will ever materialize. This suggests that by reducing the number of capacity profiles to be considered to only a few representatives ones, it might be possible to obtain a more manageable model, without losing much relevant stochastic information about the airport's capacity. This line will be pursued further in the model presented in Section 8. In the next section we will present various heuristic algorithms designed to find in reasonable time hopefully good solutions for Model C.

7. Heuristics

In [10] and [11] several families of heuristics are presented for solving Model C, along with extensive experimental results with these heuristics. We first outline below four types of heuristics:

7.1. LLA: Limited Look Ahead Heuristics

Limited Look Ahead consists of (*i*) subdividing the time horizon of T time periods into smaller subsets (of 3 time periods each, for example) and (*ii*) running the DP algorithm on each subset of time periods consecutively, taking into account the flights "left over" from the previous sub-problem (i.e., those landing in the "infinite capacity" final period relative to the previous sub-problem). The motivation for doing this is a reduction in time complexity as well as storage space requirements. As is the case for the general DP formulation, LLA Heuristics can be used with arbitrary cost functions and assume that a landing priority rule fixed *a priori* is in effect.

7.2. MMR: Maximum Marginal Return Heuristic

The MMR Heuristic also works with general cost functions and assumes a fixed landing priority rule. The basic idea for this Heuristic comes from the observation that the expected cost for flight i, $EC_i\ (X_i, H_i^1, H_i^2, \ldots, H_i^T)$, depends only on X_i and the status of flights of higher priority represented by the vector $(H_i^1, H_i^2, \ldots, H_i^T)$. Suppose that flights have been indexed, so that flight $i+1$ has priority over flight i, for all i. By definition, the flight with highest priority (flight N) is such that:

$(H_N^1, H_N^2, \ldots, H_N^T) = (0,0,\ldots, 0)$. Therefore the expected (ground plus air delay) cost for flight N, $EC_N\ (X_N, 0, 0,\ldots,0)$ does not depend on the status of any other flight; it depends only on the ground hold X_N that we impose on flight N. Thus, we can find the "optimal" ground hold, X_N^*, that leads to the lowest expected delay cost for flight N. Once we have determined the ground hold for flight N, we can proceed, in an analogous way, to assign ground holds to flights N-1, then N-2, etc. down to flight 1. The MMR Heuristic therefore computes ground holds by minimizing the expected cost for each flight individually, starting with the flight with highest priority. The MMR Heuristic may be viewed as a "myopic" strategy consisting of going after the lowest possible marginal cost increase at each step (thereby the appellation Maximum Marginal Return).

7.3 ECH: Equivalent Capacity Heuristics

The strategy for this family of heuristics is to reduce the stochastic problem to a deterministic one by reducing the probabilistic forecast (joint probability distribution) of the capacities: $P(K_1, K_2,...,K_T)$ to a single set of "equivalent" deterministic capacities $(EK_1, EK_2,...,EK_T)$ and then solve the resulting deterministic version of GHPP by using one of the approaches outlined in Section 5. If we adopt as "equivalent" deterministic capacity EK_t, the expected value of the capacity in period t (which seems to be the most natural choice) then it is not difficult to realize that the resulting strategy will not depend on the cost of airborne delay singe, as pointed out in Section 5, in a deterministic GHP there are no airborne delays. Thus this heuristic may be overly "optimistic" in the sense that it does not impose enough ground holds to minimize total expected costs. Another Heuristic in this family that is intended to compensate for this bias is the Weighted Capacity Heuristic (WCH) where the "equivalent" deterministic capacity EK_t is a weighted expected value of the capacity in period t, where the weights may be selected so as to hedge against unfavorable capacity cases by giving more weight (compared to the actual probability) to these cases.

7.4 EPH: Equivalent Policy Heuristics

The strategy for this family of heuristics is to compute ground holding policies for each of a number of capacity scenarios separately (using any deterministic algorithm) and to aggregate these policies into a single one. Again the most obvious such heuristic would be to impose on each flight i a ground hold equal to the expected value of the ground holds imposed under each separate case. The same observation as for the Equivalent Capacity Heuristics about the "optimistic" character of this policy can be made. One can, however, use a similar modification (a Weighted Policy Heuristic) to try to compensate for this, namely use "adjusted" weights instead of actual probabilities in the computation of the final ground holds.

7.5. Experimental Results

A large number of experiments have been performed comparing the heuristics described in this Section. Here we present the main conclusions emerging from these experiments (additional details and the actual numerical results can be found in [10]):

(i) The MMR Heuristic seems to perform better than all the other heuristics proposed.

(ii) The LLA Heuristic provides results that are comparable to those obtained with the MMR Heuristic in terms of total delay costs, but it is significantly slower. This is because the exact DP approach is very demanding computationally even for small instances of the GHP.

(iii) The exact DP approach can outperform all the proposed heuristics by a considerable margin when a general cost function is considered. This was demonstrated on several small examples. The exact DP approach typically gave solutions with total expected delay costs that were 20% lower than the costs of solutions obtained using the heuristics.

(iv) All the heuristic solutions have a systematic bias tending to assign more delays to smaller aircraft because of their lower cost of delay per unit of time. This raises the same questions of equitability discussed in Section 5.

8. A More Aggregate Optimal Stochastic and Static Model

The model to be discussed in this section (Richetta [9], Odoni and Richetta [7]) has been the first one to solve, optimally and with very reasonable computational effort, realistic instances of the multi-period, stochastic GHP under only mildly restrictive assumptions. The main feature of this model is that it simplifies the structure of the control mechanism by making ground hold decisions on groups of aircraft (i.e., on aircraft classified according to cost class, and schedule) rather than individual flights and by considering few rather than many airport capacity profiles. Although it will be described here for a static environment, the model can be extended to the dynamic case, as will be explained in the next section.

This model is motivated by our earlier observation that, in practice, the number of alternative capacity profiles (henceforth called scenarios) for airport Z, that can be forecast and dealt with on any given day is small. The model assumes explicitly that there are Q such scenarios (where Q is a small number), each having a given probability of materializing. Other important assumptions which permit further reductions in the computational complexity of the model are:

(i) Aircraft can be classified into a small number of different classes (typically 3 or 4) with aircraft in each class having essentially identical ground-holding delay costs. Let $C_g^k(i)$ be the function representing the cost of ground holding an aircraft of class k for i consecutive time periods

(ii) The cost of delaying one aircraft in the air for one time period is a constant c_a, independent of the type of aircraft. Thus, c_a may be considered as an overall average cost of waiting in the air. This assumption might seem unnatural at first, but is actually based on the following "operational" characteristics of the ATC system:

(1) Aircraft which are already airborne are sequenced by ATC in an approximately first-come, first-served (FCFS) way; therefore, there is no need to draw distinctions among different classes of aircraft while airborne.

(2) Within reasonable airborne delay levels (i.e., for up to the largest airborne delays observed in practice, which are in the order of one hour) delay cost functions are approximately linear, since "non-linearities", due to factors such as safety, do not yet set in.

Furthermore, computational results (see Section 10) have shown that the relative magnitude of average ground and air delay costs affects the ground hold strategy selected much more significantly than modeling air delay costs in greater detail. This observation provides further support for the "constant c_a" assumption.

We shall use the following notation next:

N_{ki} is the number of aircraft of class k scheduled to arrive at the destination airport during period i (k=1,...,K; i=1,...,T);

M_{qi} denotes the airport capacity in period i under scenario q (q=1,...,Q; i=1,...,T)

X_{qkij} represents the number of aircraft of class k originally scheduled to arrive at the destination airport during period i, and rescheduled to arrive during period j under capacity scenario q, due to a ground delay of j-i time periods (q=1,...,Q; k=1,...,K; i=1,...,T; $i \leq j \leq T+1$);

W_{qi} are auxiliary variables representing the number of aircraft unable to land at the destination airport during period i under capacity scenario q, i.e., the number of aircraft incurring airborne delay during period i (q=1,...,Q; i=1,...,T);

It should be noted that the decision variables X_{qkij} defined above are more "aggregate" than the decision variables of the previously discussed Models A-C. In the latter models, we were concerned with ground-hold delays at the individual flight level (i.e., A-C are "fine-grain" models) whereas we have now defined somewhat more "aggregate" decision variables: how many flights scheduled to arrive in period i will instead be rescheduled for period j.

8.1. Deterministic Model.

A first step toward developing a stochastic model is to write a formulation for the deterministic case. Assume that it is known with certainty that, on a given day, a particular capacity scenario q will materialize, i.e., there is only one scenario to be considered. We then have:

Model D

$$\text{Minimize} \qquad \text{Cost}(q) = \sum_{k=1}^{K} \sum_{i=1}^{T} \sum_{j=i+1}^{T+1} C_g^k \,(j-i)\, X_{qkij} + c_a \sum_{i=1}^{T} W_{qi}$$

subject to:

$$\sum_{j=1}^{T+1} X_{qkij} = N_{ki} \qquad k=1,\ldots,K;\ i=1,\ldots,T$$

$$W_{qi} \geq \sum_{k=1}^{K} \sum_{h=1}^{i} X_{qkhi} + W_{qi-1} - M_{qi} \qquad i=1,2,\ldots,T+1$$

X_{qkij} , W_{qi} $\geq$ 0 and integer.

The objective function in Model D minimizes total (ground plus air) delay costs. The first set of constraints states that all flights scheduled to land during period i must be rescheduled to arrive at i or later. The second set represents the flow balance at airport Z at the end of each time period i.

One can easily check that the coefficient matrix of Model D is totally unimodular. So one can relax the integrality constraints, since they will be satisfied by any basic feasible solution of the corresponding Linear Programming problem. In fact, it is convenient [7, 9] to model this as a Minimum Cost Flow problem on a network and to solve it through specialized algorithms.

8.2. Stochastic Static model.

Suppose now that, in the situation described above, each capacity scenario q has probability Prob(q) of materializing. Clearly, the objective function must be "weighted" over all possible capacity scenarios. Furthermore, since one still has to make ground-hold decisions at t=0, before knowing which airport capacity scenario will materialize, the decision variables corresponding to each capacity profile must be "coupled" with those corresponding to all the other capacity profiles (see coupling constraints in Model E below). The following stochastic model of the GHP is obtained (see [7] for additional details):

Model E:

Minimize $\sum_{q=1}^{Q} \mathrm{Cost}(q)\ \mathrm{Prob}(q)$

subject to:

$\boxed{\text{set of constraints for } q=1}$

.

.

.

.

$\boxed{\text{set of constraints for } q=N}$

$\boxed{\text{Coupling constraints:} \quad X_{1kij} = \ldots = X_{Qkij} \quad \forall\ k;i;j}$

where, for each value of q from 1 to Q, there is a set of constraints identical to those of the deterministic Model D.

Model E can be viewed as a Stochastic Programming problem with one stage. It is suitable for application of decomposition techniques and lends itself well to parallel computation. Richetta and Odoni [7] make the conjecture that the coefficient matrix is unimodular, since, in all the numerical experiments where integrality constraints were relaxed (Section 10 below), they obtained integer solutions. The proof of this conjecture, however, is still an open question.

As an alternative to decomposition, one can directly substitute the coupling constraints into the rest of model, hence reducing both the number of variables and the number of constraints. A summary of the extensive computational results available for Model E is given in Section 10.

9. Optimal Stochastic and Dynamic Model

Before describing the extension of Model E to the dynamic GHP, an observation about a fundamental difference between dynamic and static models must be made. In the dynamic problem, ground-holding strategies are revised over time as capacity forecasts are updated. Strategy revisions take into consideration the "current state" of the ATC system, including any earlier decisions regarding ground-holds. Thus, the expected cost of ground plus air delays is minimized by deciding, <u>at the beginning of each time period</u>, whether eligible flights will be allowed to depart or will be held on the ground. By contrast, the static solution imposes "once and for all" ground-holds at time zero (i.e., at the beginning of the first time period of the day). In terms of modeling, this means that, in addition to scheduled arrival times, <u>scheduled departure times</u> must also be considered explicitly in developing dynamic strategies.

The dynamic version of Model E (Odoni and Richetta [8]) can now be described with reference to Figure 2. The dynamic evolution of the capacity forecasts and the implicit updating of the associated probabilities is modeled through a probability tree. Taking a forecast consisting of three capacity scenarios for airport Z as an example, Figure 2 shows that the forecast is updated three times during the interval $[0, B]$. These three instants (denoted t_1, t_2 and t_3 in Figure 2) define three <u>stages</u> comprising the time intervals $[t_1, t_2)$, $[t_2, t_3)$, and $[t_3, B]$ (a capacity forecast consisting of Q capacity scenarios would consist of at most Q stages). Within each stage, the probability of each of the scenarios for future airport landing capacity does not change. Therefore, an optimal dynamic solution to the GHPP assigns and/or revises ground-holds at the beginning of each stage. The time at which stage s starts will be referred to as t_s below.

As in the static Model E, an implicit assumption here is that the number of alternative scenarios at the beginning of each day, Q, is quite small - probably 4 or less. This assumption is important for obtaining quick numerical solutions. A small value of Q is, once again consistent with current weather forecasting technology which has advanced to the point where the type of weather conditions in a specific geographic area can be predicted with reasonable accuracy, but the exact timing of weather fronts and their local severity are uncertain. A typical example of the type of situation that can be addressed through this approach is shown in Figure 7 of Section 10 where, at the beginning of a day, there is an expectation of some deterioration in weather conditions in early afternoon which may result in severe loss of landing capacity (profile 1) limited loss

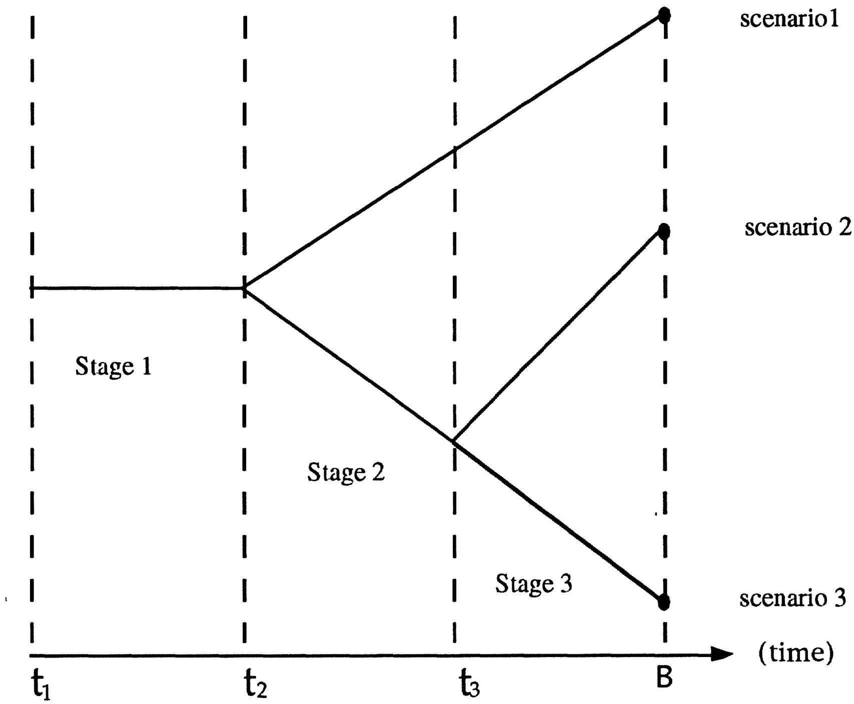


Figure 2: Number of Stages Defined by the PMF of Airport Capacities

(profile 2) or no loss at all (profile 3). For that example, of course, Q=3, t_1=6:00, t_2=14:00, t_3=15:00 and B=24:00 hours.

In the static solution to the GHP, the time interval [0, B] comprises a single stage, resulting in a "here-and-now" solution which assigns ground-holds at t_1. In the dynamic case there are up to Q stages at which we make ground hold decisions. In general we can have Q or fewer stages (fewer than Q stages when three or more capacity scenarios become distinct at the same point in time).

Model F

Due to the role of scheduled departure times, the dynamic Model F requires a modification in some of the notation defined in Section 8. Specifically:

N_{ksi} is the number of aircraft of class k scheduled to depart during stage s and arrive at airport Z during period i (k=1,...K; s = 1,...,Q; i = t_s+1, t_s+2,..., T)

X_{qksij} is the number of aircraft of class k scheduled to depart during stage s and arrive at airport Z during time period i which are rescheduled to arrive during period j, under capacity scenario q (k=1,...,K; q, s = 1,...,Q; i = t_s+1, t_s+2,..., T; $i \leq j \leq T+1$).

After substitution of the decision variables X_{qksij} and of the demands N_{ksi} into the objective function and corresponding network model (see Model D above), Model F is entirely analogous to Model E, except that the coupling constraints are as shown in Figure 3. Note that there is one set of coupling constraints for each stage s of the problem. This reflects the fact that, at the beginning of each stage, we must assign ground-holds without knowing which of the possible capacity forecasts will actually materialize. This reasoning is again similar to that for Model E. The reader is referred to Richetta [9] and to Odoni and Richetta [8] for a detailed description of Model F.

Coupling Constraints:

$X_{sksij} = X_{s+1\,ksij} = \ldots = X_{Qksij}$;

$s = 1, \ldots, Q-1;\ k = 1, \ldots, K;\ i = t_s + 1, \ldots, T;\ i \leq j \leq T+1$

Network Component for q = 1

Network Component for q = Q

Figure 3: Constraint Matrix Structure for the Multistage Problem

The resulting optimization problem may be solved by using standard techniques of multistage Stochastic Programming, with the stages corresponding to the time instants, t_s, when new or updated information may become available.

Finally, we note that additional constraints, such as placing limits on the maximum acceptable ground-holds and/or airborne delays, can be introduced into Models E and F easily.

A Simple Example

The difference between dynamic and static strategies is illustrated by the following idealized example, which involves only two flights. Figure 4 shows a diagram of the flight "schedule". Flight F_1 is scheduled to depart at time 1, F_2 is scheduled to depart at time 2, and both flights are scheduled to arrive at an airport Z at period 3.

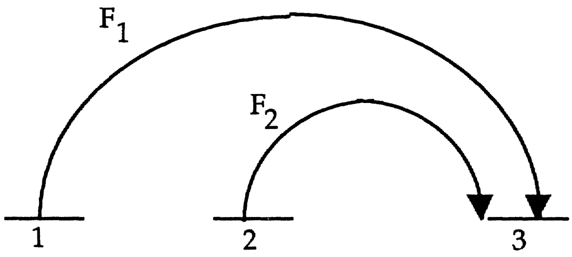


Figure 4: Flight Schedule

Landing capacity, M, at Z during the arrival period, time 3, is limited to one or two flights according to the probability tree shown in Figure 5. We assume that capacity during the next time period, 4, is unrestricted. The probability tree of Figure 5 shows the possible evolution of capacity at Z over time, as perceived at time 1. As happens in practice, airport capacity during time period 3 is partially

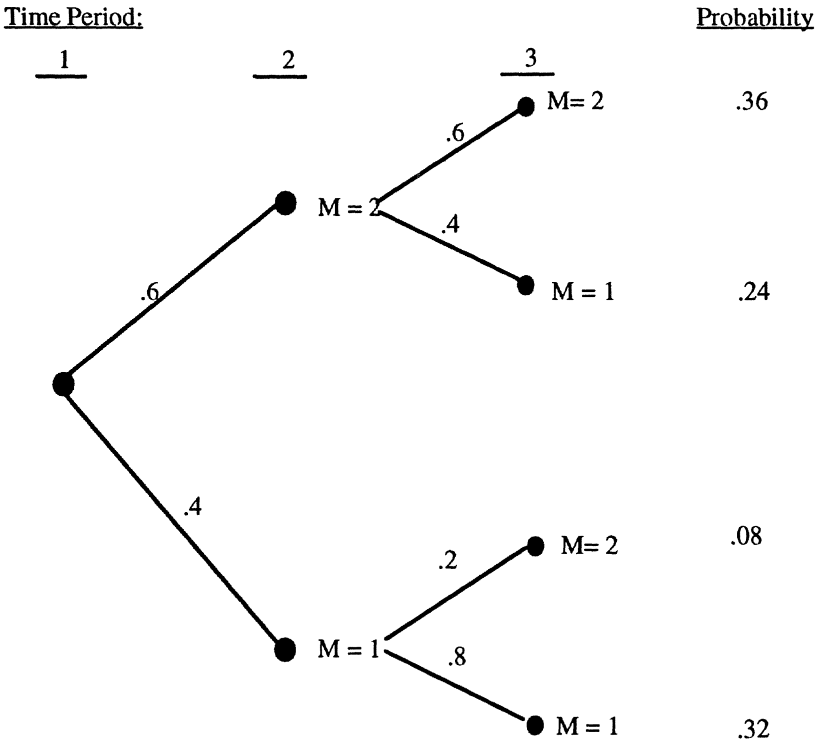


Figure 5: PMF of Airport Landing Capacities

correlated to capacity available during time 2. If time 2 capacity is 2, then there is a greater chance of having a high capacity during time 3; while if time 2 capacity is 1, there is a greater probability of having limited capacity during time 3.

Next we specify the ground and air delay costs for F_1 and F_2. Since F_1 and F_2 are both scheduled to arrive during time 3, and time 4 capacity is unrestricted, we only need to consider the cost of one period of delay:

Flight	Ground Delay Cost	Air Delay Cost
F_1	cg_1 = \$1000	ca_1 = \$2000
F_2	cg_2 = \$1100	ca_2 = \$2200

In line with what we would expect in a real situation, the cost of air delay is higher than that of ground delay reflecting the higher operational cost of airborne aircraft. The two aircraft have different costs reflecting factors such as aircraft type, passenger load, fuel efficiency, connection schedules, etc.

The static solution assigns ground-holds to both flights, F_1 and F_2, at the beginning of time 1 based only on the information available then. Since the probability of having capacity limited to one landing during time 3 is .56, the optimal static strategy is to let the more costly flight, F_2, depart according to schedule and delay F_1 one time period for an optimal static cost of \$1000 (if flight F_1 were also allowed to depart on time, the expected cost would be (.56)(2000) + (.44) (0) = \$ 1,120).

In the dynamic problem we make ground-hold decisions on a period by period basis, using the history of airport capacities to produce an updated capacity forecast regarding future airport landing capacities. Consider the following dynamic strategy:

Let F_1 depart at time 1 (i.e., according to schedule). At time 2, if time 2 capacity is 1, delay the departure of F_2 one time period; otherwise, let F_2 depart according to schedule. By conditioning on the value of airport capacity at time 2 we see that the expected delay cost of this strategy is (.4)(1,100) + (.6)(.4)(2,000) = \$920, representing a significant cost improvement vs. the optimal static strategy. The reader can verify that this is indeed the optimal dynamic strategy. Note that if both aircraft are allowed to depart on time under this strategy (the probability of this event is .6), and the capacity at time 3 turns out to be equal to 1, then the less expensive flight, F_1, is the one which is held in the air for one time period.

The example also points to what is indeed a systematic bias in dynamic decision-making for the GHP: optimal strategies favor long-range flights over short-range ones, in the sense that long-range flights are more likely to be allowed to take-off on time (i.e., with no or little ground-holding). Intuitively, good dynamic strategies would tend to be more "active" with short-range flights (i.e., impose more ground-holds on them) in order to take advantage of the improved state of knowledge at the time when short-range flights are scheduled to depart. Current practice partly reflects this tendency: for example, flights to the United States from Europe or non-stop coast-to-coast flights in the United States are typically exempt from ground-holding.

10. Computational Experiments

A large number of computational experiments have been carried out with Models E and F using data from Boston's Logan International Airport, which in 1990 was the 15th busiest airport in the world in terms of number of passengers served. In this section, we describe these experiments and their results. (Detailed data regarding all aspects of the computational experiments can be found in Richetta [9].)

10.1 Instances of the GHP for Logan Airport

We used a schedule of aircraft arrivals for a typical weekday of operations at Logan Airport during the Fall of 1988 based on information taken from the November 1988 issue of the *Official Airline Guide*. It is worth noting that, there are also approximately 50 unscheduled daily flights into Logan (less than 10 percent of the total) which are not subject to ATCSCC ground-holding. Aircraft were classified into the three classes defined by their maximum take-off weights (MTOW), with aircraft classes 1, 2, and 3 corresponding to small, medium/large, and heavy aircraft respectively. There are a total of 551 scheduled arrivals between 6 a.m. and midnight. This time period was subdivided into 72 fifteen minute intervals and a 73rd representing the "dummy" final period $T+1$.

Figure 6 shows hourly landings by aircraft type for scheduled flights. Small and medium/large aircraft, each with approximately 45% of scheduled flights, account for about 90% of Logan traffic. During the busiest periods (8 to 11 and 16 to 19 hours) landing demand for scheduled flights averages 36 and 43.5 landings per hour respectively, representing 60% and 74% of the "good weather" maximum landing capacity of 60 aircraft per hour. Thus, in good weather this schedule yields little congestion. However, bad weather

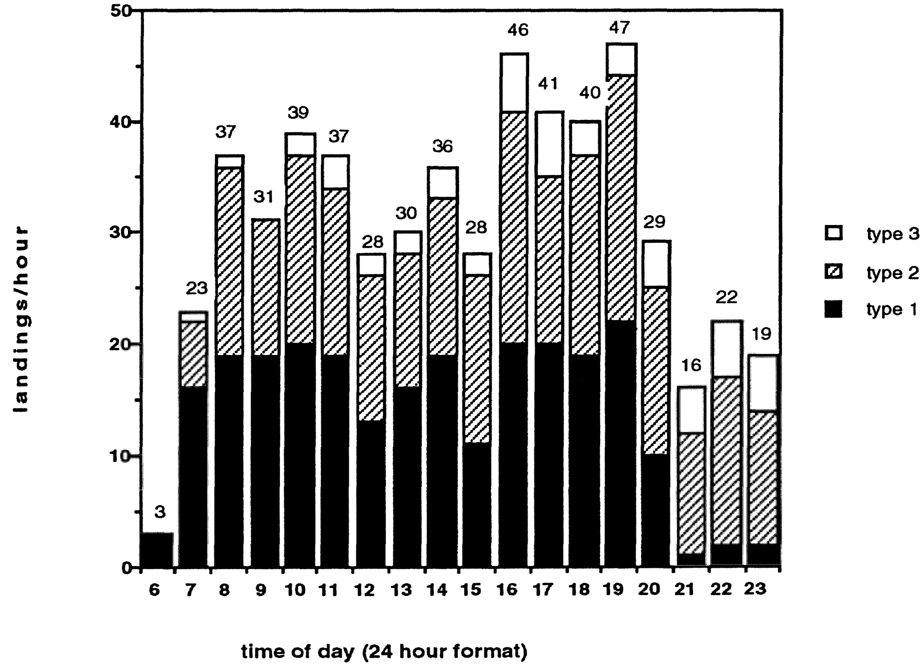
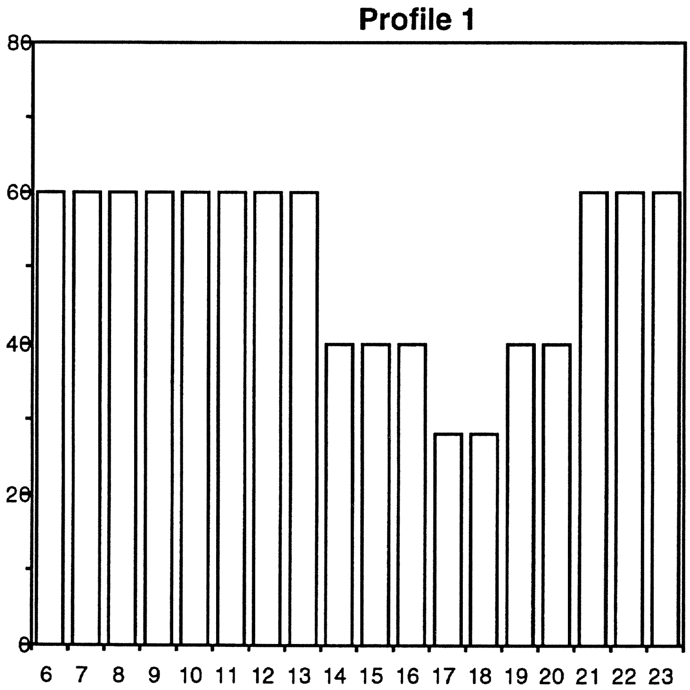
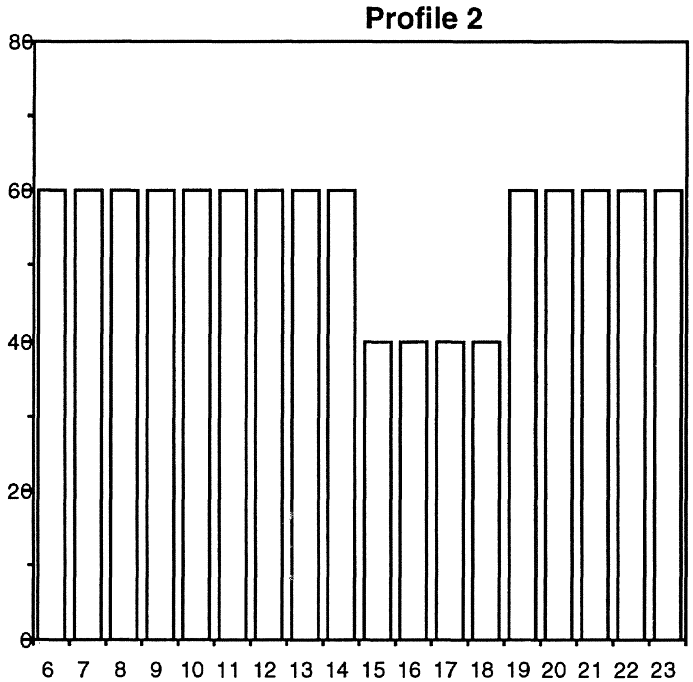


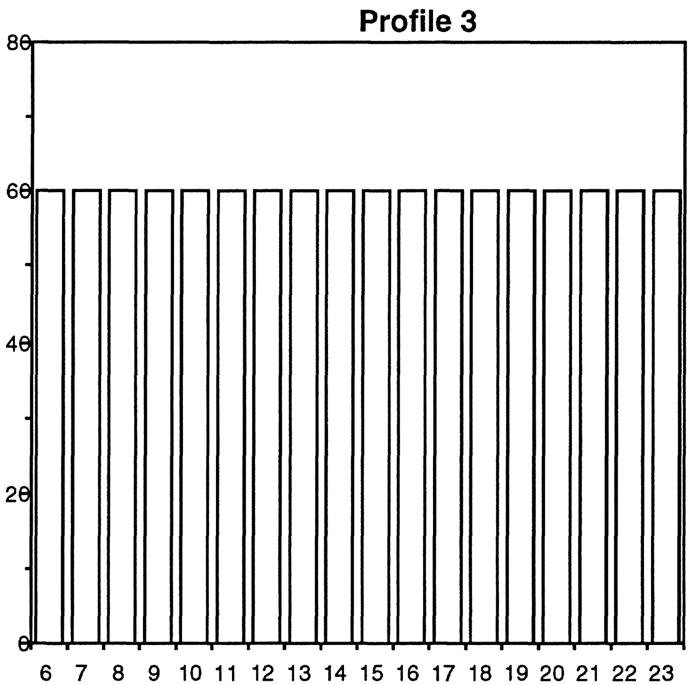
Figure 6: Hourly Landings By Aircraft Type for Scheduled Flights

can reduce landing capacity significantly - and it is precisely during bad weather days that uncertainty about landing capacity is the greatest. This is the motivation for pursuing probabilistic approaches to the GHP.

A total of 10 different capacity cases, consisting of three capacity profiles each, were studied. Four different probability scenarios were explored for capacity cases 1-3, and a single probability scenario for capacity cases 4-10, for a total of 19 different capacity forecasts. The capacity forecasts cover a wide variety of conditions with regard to the levels, timing and duration of periods of restricted capacity, and reflect operating conditions typically prevailing at Logan during bad-weather days. Figure 7 shows the three capacity profiles for capacity case 1 and Table 1 the corresponding four probability scenarios. For example, for capacity case 1, under probability scenario 1, profiles 1, 2, and 3 have probabilities of .5, .3, and .2 respectively.







(Vertical axis: landings/hour. Horizontal axis: time of day - 24 hour format)

Figure 7: Capacity Profiles for Landing Capacity Case 1

	Probability Scenario			
Profile #	1	2	3	4
1	.5	.3	.3	.34
2	.3	.5	.2	.33
3	.2	.2	.5	.33

Table 1: Probability Scenarios for Capacity Case 1

The capacity levels used in preparing the forecasts were 60, 40, and 28 landings per hour, corresponding to VFR1, VFR2/IFR1, and IFR2/IFR3 conditions respectively. VFR stands for visual flying rules and IFR for instrument flying rules. Airport landing capacity under IFR conditions decreases versus VFR conditions as aircraft minimum separation rules are enforced strictly, increasing the time between landings. As well, some landing runways available for VFR operations may not be equipped with an instrument landing system (ILS), further reducing capacity during IFR conditions. Logan historical data indicate that VFR1 weather conditions prevail about 80% of the time, VFR2/IFR1, 12% of the time, and IFR2 and worse during the remaining 8%. We have not included extreme cases such as shut down of operations, that are likely to require flight cancellations due to unacceptable delay levels.

In order to assure FCFS within each aircraft class, the ground delay cost functions used are slightly increasing. For the case of a single aircraft class the ground-hold costs per aircraft are $250/period (i.e., $1000/hour) for the first period of ground-holding and then increase by $10/period. In the case of three aircraft classes, the ground-hold cost function used yields the same average ground-hold cost of $1,000/hour for a single class of aircraft, based on a 45%-45%-10% aircraft class split. The marginal rate of ground-hold cost increase is $10/period. The cost for the first period of ground delay by aircraft class is shown in Table 2 and is typical of the true direct costs to operators of these types of aircraft.

	Aircraft Type		
	class 1	class 2	class 3
Aircraft Split	(45%)	(45%)	(10%)
Ground delay Cost (First Period)	\$430/hour	\$1,300/hour	\$2,225/hour

Table 2: First Period Ground-Hold Delay Cost

As noted already at several points, we can adjust the bias of our models towards conservative (liberal) ground-holding strategies by increasing (decreasing) the cost of air delay, c_a. For single aircraft class algorithms we explore marginal air delay costs of \$1,200, \$1,600, \$2,000, and \$3,000 per hour, representing cost premiums of approximately 20%, 60%, 100% and 200% vs. the average cost of ground delays. We solved the GHP for everyone of the problems defined in Table 3 using the algorithms described below (except for algorithms with three aircraft classes for which we only solved cases with marginal air delay cost of \$3000/hour so that the cost of air delays is always higher than the cost of ground delays for all aircraft classes).

Capacity Case	Number of Probability Scenarios	Number of Forecasts	Air Delay Costs	Number of Problems*
1-3	4	12	1200, 1600 2000, 3000	48
4-9	1	6	1600	6
10	1	1	3000	1
		19		55

* A problem is defined as a capacity forecast - air delay cost combination.

Table 3: Problems Generated by the Different Capacity Forecast- Air Cost Combinations

10.2 The Models

We evaluated the performance of 6 models, including the passive strategy of no ground-holds, by solving the GHP's defined above.

1. *Deterministic*: This model provides a static deterministic solution to the GHP by treating the most likely capacity profile in the probabilistic forecast as the only capacity profile, i.e., as the deterministic capacity forecast, and disregarding completely all other profiles. Available capacity is then assigned on a FCFS basis with all delays assigned as ground-holds. This model is denoted DETERM in the Figures and Tables that follow.
2. *Passive*: This model allows aircraft to depart according to the original schedule, i.e., does not impose any ground-holds, and calculates expected air delays using the probabilistic landing capacity forecast. The model is denoted PASSIVE.
3. *Static*: This model provides the optimal probabilistic static solution for the GHP with a single aircraft class. It is solved by using Stochastic Linear Programming with one stage (i.e., Model E). This is the "once and for all" optimal solution to the GHP at time t_1. The model is denoted STATIC.
4. *Static with Three Aircraft Classes*: The same as 3 above, but with three aircraft classes . This model is denoted STATIC3C.
5. *Dynamic*: This model provides the optimal dynamic solution to the GHP with one aircraft class via Stochastic Linear Programming with Q stages (i.e., Model F). It The model is denoted DYNAMIC.
6. *Dynamic with Three Aircraft Classes*: The same as 5 above, but with three aircraft classes. This algorithm is denoted DYNAMIC3C.

PASSIVE was implemented using a C programming language code that utilizes the input schedule to calculate expected air delays using the probabilistic landing capacity forecast. DYNAMIC, STATIC, DYNAMIC3C, and STATIC3C are Stochastic Linear Programming models. Stochastic Linear Programs of the size we have described can be solved on a personal computer. DETERM was implemented as the deterministic linear program presented in Section 3.1.

To solve the linear programs resulting from DETERM, DYNAMIC, STATIC, DYNAMIC3C, and STATIC3C we used LINGO on a 386 machine with a 80387 co-processor and 4mb of RAM memory.. LINGO combines an exterior point linear programming algorithm (LINDO) and a modeling language for problem input. Our version of LINGO is able to solve problems with a constraint matrix size of up to 5000 x 15,000, for moderately dense matrices.

Running times ranged from under 10 minutes for the optimal STATIC solutions with a single class of aircraft to more than 3 hours for the optimal DYNAMIC solutions with three classes of aircraft. With a fast workstation and a somewhat customized solution software, it is safe to expect that these running times can be reduced by at least a factor of 10.

10.3 Performance Evaluation

The performance of the models was evaluated with respect to the following statistics for each solution: total expected cost; expected cost of ground and air delays; expected ground delay and expected air delay measured in aircraft-periods. Recall that DYNAMIC3C and STATIC3C apply only to problem instances with marginal air delay cost of $3000/hour. For that reason, these two models are not included in the overall performance assessment immediately following.

(a) Overall Performance:

Figures 8 and 9 show the average performance of the single aircraft class models tested (i.e., DETERM, PASSIVE, DYNAMIC and STATIC). Figure 8 is on a percentage basis (i.e., equal weight for each of the 55 problems shown in table 4) while Figure 9 shows the average expected total delay cost of the solutions.

Figure 8 indicates that STATIC provides a 6.6% average savings vs. DETERM, while DYNAMIC provides average savings of 29.3%, showing a significant advantage for the dynamic solution to the GHP.

It can also be seen that all the models that include ground-holding perform significantly better than the PASSIVE strategy of no ground-holds.

The importance of dynamic strategies was further demonstrated in [9] through a series of experiments with a heuristic, DNM-HEUR that computes ground-holds dynamically (i.e., at the beginning of each stage defined by the probabilistic forecast) using DETERM. (Note that some of the information conveyed by the probabilistic forecast is taken into consideration by the heuristic which utilizes the most likely capacity forecast at each stage for DETERM.) The performance of DNM-HEUR was remarkably close to that of DYNAMIC, showing 25% savings relative to DETERM.

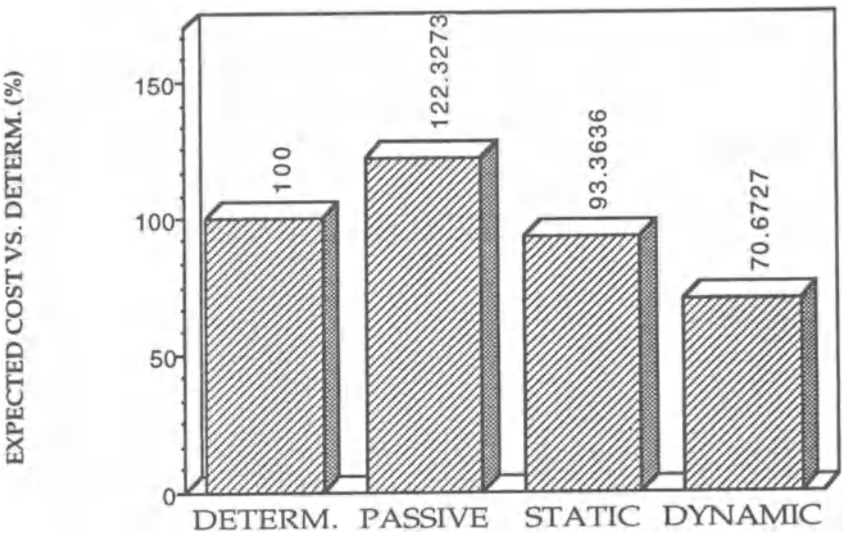


Figure 8: Average Cost Performance (% basis)

Figure 9 shows that the average cost savings provided by the DYNAMIC model are traceable mainly to significant reductions in the expected cost of air delays. On the other hand the STATIC model shows an increase in expected air delay costs which offsets most of the savings in ground-holding cost. The reason for the significant improvement in the expected cost of air delays for the DYNAMIC model is that, by updating the capacity forecast at each stage, this model generates ground-holding policies that reduce expensive air delays significantly. In the case of PASSIVE, we see that the higher cost of air delays results in significantly higher expected delay cost, despite the fact that PASSIVE minimizes the total expected delay, by allowing all aircraft to take off on time (and thus never "wasting" any available airport capacity).

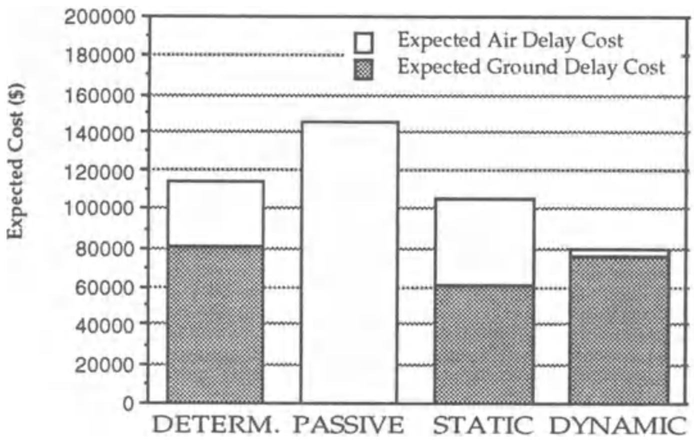


Figure 9: Average Cost Performance ($ basis)

Figure 10 illustrates this last point by showing the expected number of aircraft-periods of air and ground delay averaged for all the problems solved. We see that total expected delays for PASSIVE are indeed the lowest of all the algorithms tested. Unfortunately, these delays are all in the form of expensive air delays. Remarkably, DYNAMIC has the advantage of producing very low air delays while maintaining the total number of delay periods within 10% of the minimum achieved by PASSIVE. In the case of STATIC, total expected delays are significantly lower than those for DETERM, but only within 15% of the minimum given by PASSIVE, demonstrating again the advantage of the dynamic solution vs. the static solution.

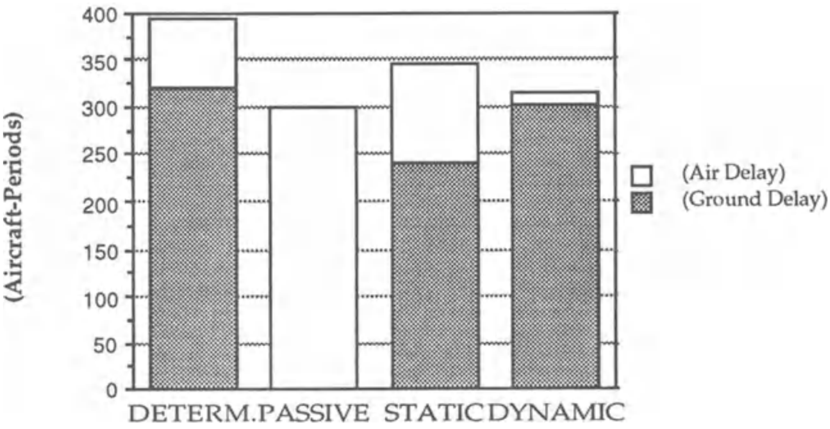


Figure 10: Average Expected Ground and Air Delay

(b) Effect of Marginal Air Delay Costs:

Figure 11 shows the average relative performance of the models for each marginal air delay cost value tested for capacity cases 1-3 (i.e., the capacity cases for which we tested every marginal air delay cost as shown in Table 3). With no exception, the performance of DYNAMIC improves as the air cost increases, showing a 22.2% savings versus DETERM for marginal air delay cost of $1200/hour while for marginal air delay cost of $3,000/hour the savings are 38.2%.

On the other hand, STATIC only provides savings versus DETERM in the 3% to 12% range, with the best relative performance achieved when the air delay cost is $1,200/hour. The advantage of STATIC over DETERM seems to deteriorate as the cost of air delays increases. This is because, as the marginal air delay cost increases - particularly for cases in which the most pessimistic capacity scenario is the most likely scenario - the solution for DETERM improves (i.e., for high enough marginal air delay cost the optimal static solution is the strategy of assigning available capacity for the most pessimistic capacity profile on a FCFS basis with all delays taken in the form of ground-holds).

The very significant savings achieved for an air delay cost of \$3,000/hour (Figure 11b, bottom) by DYNAMIC3C and STATIC3C, versus DYNAMIC and STATIC respectively, are due to the assignment of ground-holds to the lowest cost aircraft eligible for delay. We also see that the performance of PASSIVE deteriorates with increasing air delay cost as we would expect.

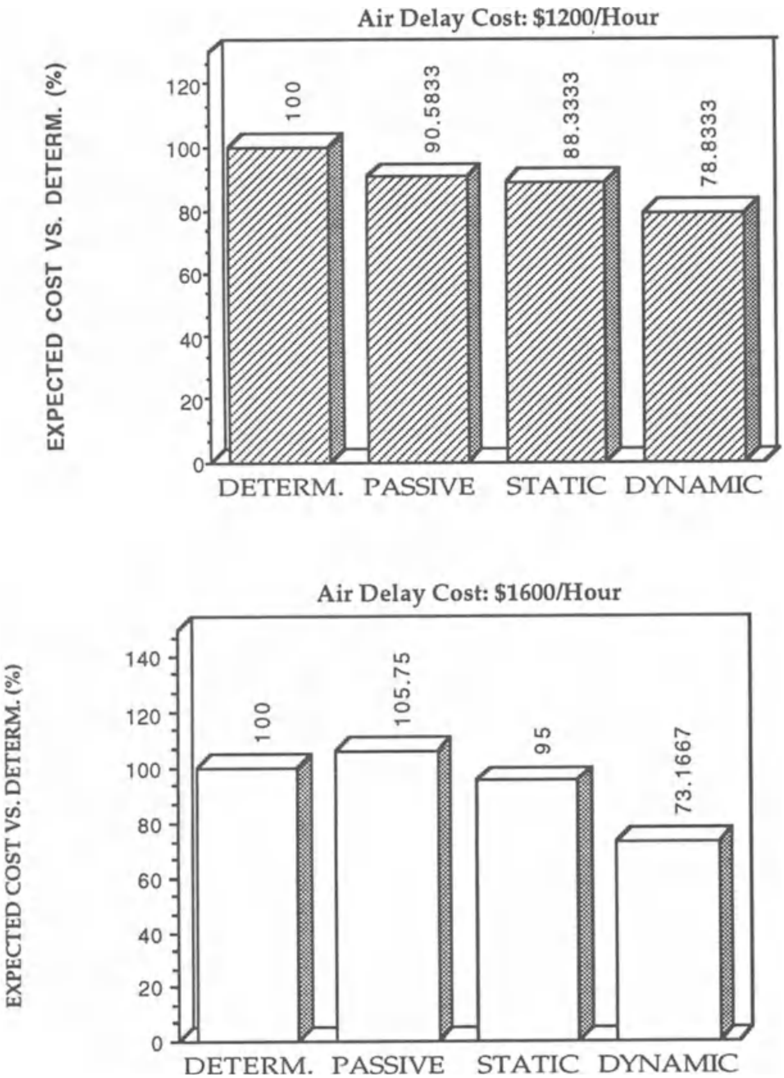


Figure 11a: Average Cost Performance by Air Delay Cost: Capacity Cases 1-3

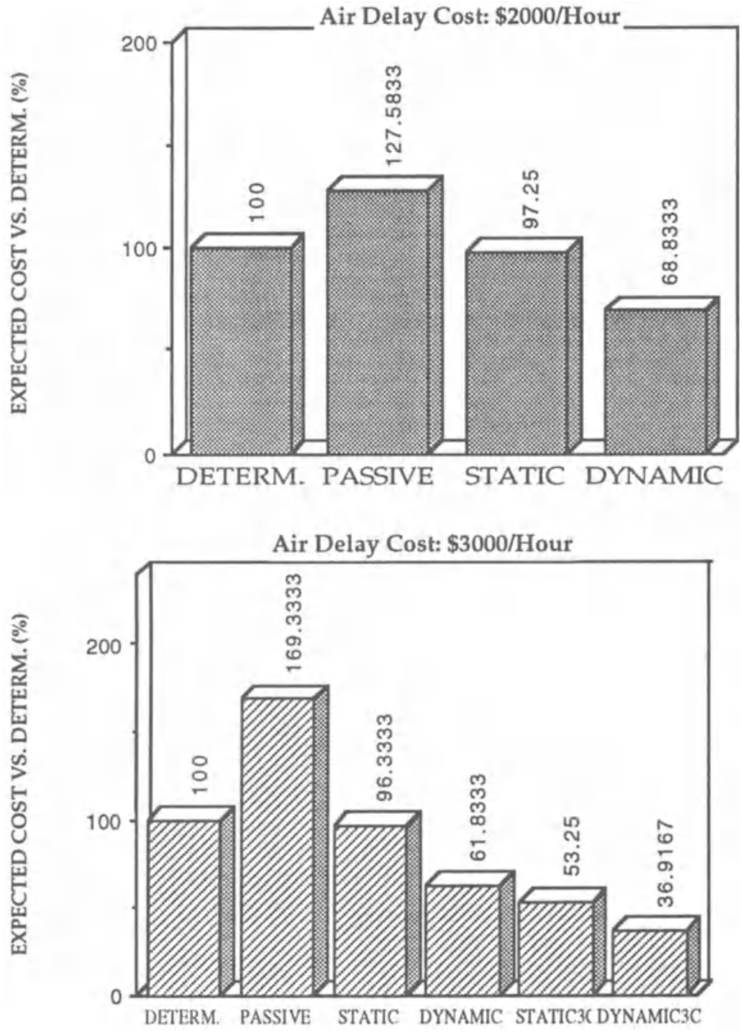


Figure 11b: Average Cost Performance by Air Delay Cost: Capacity Cases 1-3

(c) <u>Effect of Ground-Hold Cost Function For Three Aircraft Classes:</u>

Figure 12 shows the average expected ground delay for each aircraft class for STATIC3C and DYNAMIC3C. Unlike models with a single aircraft class, STATIC3C and DYNAMIC3C result in ground-holds that assign to the lower cost aircraft classes ground-holds which are substantially higher than the 45%-45%-10% split for aircraft classes 1,2, and 3 respectively. In Figure 12 small aircraft absorb over 90% of the ground delays for STATIC3C, and over 80% of the ground delays for DYNAMIC3C. This is the reason why any ground-holding model that may eventually be implemented, at least in the United States, will probably be for a single class of aircraft, due to the FAA's policy of "equal access" to airports for all users who meet the relevant equipment and training requirements.

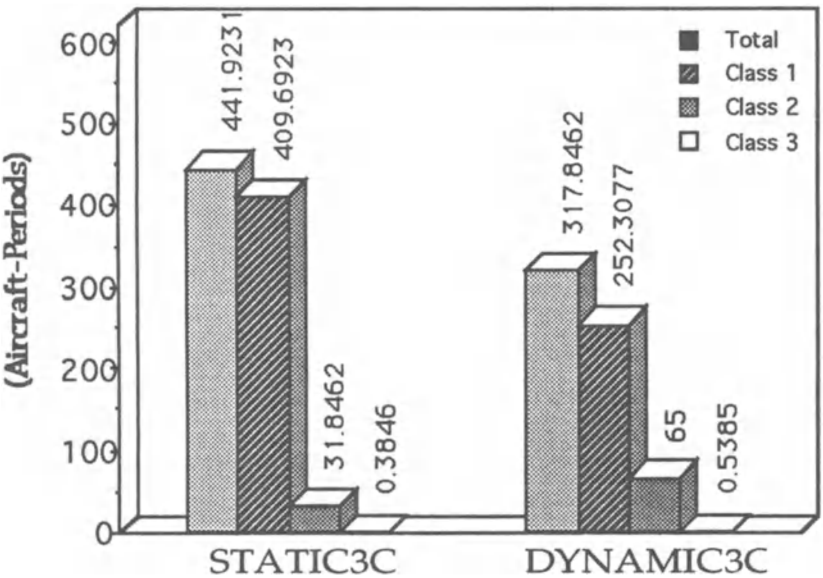


Figure 12: Average Expected Ground Delay By Aircraft Class

11. Conclusions

We have reviewed a sequence of models for the single-airport ground-holding problem (GHP). The models discussed were increasingly generalized, ranging from a singlé-period, stochastic and static model, to a multi-period, deterministic model, to a multi-period, stochastic and dynamic one. The most appropriate model for each practical case will depend on the decision-making environment and on how much information is available to an ATC flow management unit.

Many computational experiments have indicated that these models may be able to suggest ground-holding strategies that could result in significant reductions of delay costs. These same experiments were also encouraging with regard to the feasibility of employing the models as real-time decision support tools for ATC flow managers. Clearly, a natural next step would be to conduct a more realistic set of experiments that may include professional air traffic controllers in a simulated environment such as the one provided by the FAA's National Simulation Capability (Zellweger [14]). Such an effort would undoubtedly spur further research on (i) the desirable type of "interfacing" between controllers and the decision support tools and (ii) variations on the models described here that might provide additional flexibility and options to controllers.

A second direction for future work is to extend GHP models along two lines. The first is a multi-airport environment that captures the possibility that delays at one airport will also spread to other airports (Vranas [12]) because of "interactions" among flights, as described in Section 2. Second, it is also desirable to have models in which not only airports, but en route sectors, as well, could act as major bottlenecks. (Of course, en route sector delays are taken in the air and therefore their costs are different than those taken on the ground at airports.) Several teams of researchers, including the authors of this paper, are currently working on this "capacitated network" problem (see also Mulvey and Zenios [5]) that contains many complicating considerations.

Acknowledgments

The research of the first author was supported in part by research grants from the Progetto Finalizzato Trasporti 2 of the Italian National Research Council (contract no. 92.01845.PF74).

The research of the second and third authors was supported in part by research grants from the C. S. Draper Laboratory and the National Science Foundation (grant no. 9014751).

References

[1] G. Andreatta and G. Romanin-Jacur, *The Flow Management Problem*, LADSEB-CNR, Report 03/86, Padova, Italy (April), 1986.

[2] G. Andreatta and G. Romanin-Jacur, *Flow Management Problem and Optimal Priority Rule*, LADSEB-CNR, Report 04/86, Padova, Italy (May), 1986.

[3] G. Andreatta and G. Romanin-Jacur, "Aircraft Flow Management under Congestion", *Transportation Science*, 21, No. 4, 1987, pp. 249-253.

[4] G. Andreatta and G. Romanin-Jacur, "Aircraft Flow Management Problem", in *Operational Research* '87 (G.K. Rand, editor) Elsevier, 1988, pp. 626-634.

[5] J. M. Mulvey and S. A. Zenios, "Real-time operational Planning for the U. S. Air Traffic System", *Applied Numerical Mathematics*, 3, 1987, pp. 427-441.

[6] A. R. Odoni, "The Flow Management Problem in Air Traffic Control" in *Flow Control of Congested Networks* (A. R. Odoni, L. Bianco and G. Szego, editors) Springer-Verlag, Berlin, 1987, pp. 269-288.

[7] A. R. Odoni and O. Richetta, "Stochastic Linear Programming Solution to The Static Ground-Holding Policy Problem (GHPP) in Air Traffic Control", *Transportation Science*, 27, 1993 (forthcoming).

[8] A. R. Odoni and O. Richetta, "*Dynamic Solution to The Ground-Holding Policy Problem (GHPP) in Air Traffic Control*", Working Paper, Operations Research Center, MIT, Cambridge, MA, July 1992.

[9] O. Richetta, *Ground Holding Strategies for Air Traffic Control Under Uncertainty*, Operations Research Center Technical

Report No. 198, MIT, Cambridge, MA, July 1991.

[10] M. Terrab, *Ground Holding Strategies in Air Traffic Control,* Operations Research Center Technical Report No. 196, MIT, Cambridge, MA, February 1990.

[11] M. Terrab and A. R. Odoni, "Strategic Flow Management for Air Traffic Control", *Operations Research*, 61, No. 1, January-February 1993, forthcoming.

[12] P. Vranas, "*The Multi-Airport Ground-Holding Problem in Air Traffic Control*", Ph.D. thesis, Massachusetts Institute of Technology, June 1992.

[13] H. Wang, "A Dynamic Programming Framework for the Global Flow Control Problem in Air Traffic Management", *Transportation Science*, 25, No. 4, November 1991, pp. 308-313.

[14] A. Zellweger, "*FAA's National Simulation Capability*", this volume.

AN APPLICATION OF EXPERT SYSTEMS AND PARALLEL PROCESSING TO ATC PLANNING PROBLEMS

Robert W. Simpson
Director, Flight Transportation Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139
U.S.A.

Expert Systems may be used for non-critical planning processes of ATC such as generating planned A-D trajectories for aircraft arriving at an airport, and for creating a schedule plan for the runway configuration to be used throughout the day at the airport. Two such applications have been implemented and tested at MIT/FTL. The first one was implemented in a language called Id which is designed for parallel processors. It used a set of Arrival Patterns to the runway to create a consistent set of conflict-free trajectories. By using a parallel processor with 128 parallel units, it was shown that a new trajectory could be calculated in 0.1 seconds if the processor speed was 40 MIPS. The second implementation required the Expert System to do planning in a time domain. It was implemented in LISP, and required an extension of Expert System technology which could generated time-based inferences.

1. Introduction

Operations in the terminal area airspace around major airports are usually the bottleneck in attempting to increase ATC capacities. To improve the safety and efficiency of future high density terminal area ATC operations, it is desirable to introduce automated interactive planning tools to assist the ATC managers and controllers. In major terminal areas, there may be more than one airport and probably several runways simultaneously in operation. Their geometric layouts are always different, and even in one terminal area the operational procedures and traffic paths will change over time. It would be expensive to develop a special set of planning tools for every terminal area, and to adapt them quickly to various changes in operational procedures from one year to the next. Because operational situations are changing dynamically as aircraft arrive for both landings and takeoffs, it is clear that these planning tools must be capable of quick response in safely re-planning all operations. This suggests that the software technology called "Expert Systems" and the hardware/software technologies of Parallel Processing might play a role in the development of these advanced ATC planning tools.

2. The Terminal Area Air Traffic Problem and Processes

The core activity in a terminal area airspace is the landing and takeoff operations on several runways. Aircraft arrive from all directions, descending through the surrounding airspace to enter the terminal area, and begin the merging processes which end up in an efficient spacing between all landing traffic destined for the same runway. On the airport surface, departing flights are assigned to their runways, and then taxi to reach each runway's threshold where they will be inserted between the landing operations. As they become airborne, their climbing departure paths must be planned to avoid conflicts with each other and with all the arrival paths. Thus, there must be a Runway Scheduling Process that sets a timing to within a few seconds for takeoff and landing operations which is dynamically changing as new traffic arrives; and a Flight Path Generation Process, that generates a set of descending arrival and climbing departure flight paths in four dimensions to support the dynamic schedule. These flight paths must be conflict-free (i.e., they must meet specified safe separation constraints in time and space). These two processes can be called Terminal Area operations planning processes and would be utilized on a minute to minute basis.

There is also a longer term strategic Terminal Area planning process which is concerned with planning the configuration of the runways to be used at each airport in the terminal area over the next several hours. A Runway Configuration consists of the set of runways at an airport and their usage for takeoffs or landings by type of aircraft. Given knowledge about the future traffic, equipment status, weather (in the form of wind, visibility, ceiling, precipitation), and requirements for runway and equipment maintenance, noise abatement, snow removal, etc.) a Runway Configuration Planning Process produces a plan, (or schedule for runway changes to the nearest 15 minutes) for feasible runway configurations at each airport in the terminal area. This process is needed to forecast the airport's operational capacity for the Traffic Flow Management Process which regulates the traffic flows into the airports and terminal area.

There are also much more critical and dynamic Separation Assurance Processes. Given that an efficient and safe operational plan can be generated, ATC monitors the 4-D conformance of every aircraft to its planned path to ensure safety (i.e. there is a Conformance Management Process); and also imposes backup processes called Hazard Detection and Resolution Processes to ensure that no errors or unexpected events can cause a collision. These processes are real time Separation Assurance processes with a requirement for very short response times (to a few seconds) and any automated versions must be fully validated to be operational. They are unlikely candidates for the application of Expert Systems Technology.

But the Terminal Area planning processes have longer response times and are not safety critical so that the requirement for full validation is not severe. Their outputs can be scanned by the human controller and modified, and the Separation Assurance process, whether automated or manual, will still ensure the safety of operations. These planning processes are candidates for the application of Expert Systems software.

3. Software Technology for Expert Systems - 1

Computational algorithms have two disjoint components:

a) a set of rules expressing relationships between the variables of a problem;
b) a solution strategy which orders the operation of the rules.

For ordinary algorithms, the solution strategy is predetermined by the creator of the algorithm who is interested in efficiency and the proof of convergence, and perhaps optimality. A "flow chart" can be created to describe the computational path(s) which will be followed.

But for an "Expert System" the user will only input knowledge in the form of rules, statements, constraints, relationships, etc. The "Expert System" will pursue its own solution strategy using a "core" process created by its inventor. The computer code for an Expert System does not have a fixed Flow Chart defining the sequence or paths for executing its rules or statements, and its methods of finding a solution may surprise even its inventors from one problem to the next. The behaviour of the Expert System is dependent upon the exact combination of knowledge and data that it possesses at any point in time, and it is impossible to validate reliable behavior in all circumstances. As indicated above, this makes it an unlikely candidate to handle Separation Assurance Processes in ATC since complete safety of the automation process cannot be assured.

The advantages of an Expert System are its transportability amongst different hardware environments, its adaptability to different operational problems in a generic area, and the maintainability of the software over time. These are very desirable qualities for automation of the Terminal Area planning processes. We shall now present the results of two research projects which investigate the application of Expert Systems and Parallel Processing technologies to automated planning processes in the terminal area.

4. An Expert System for Terminal Area Flight Path Generation

Reference 1 describes an Expert System for the Flight Path Generation Process developed at the Flight Transportation Laboratory, MIT by M.L. Sadoune. It uses the Dataflow Architecture for parallel computation developed at the MIT Laboratory for Computer Science. It was programmed in a declarative language called "Id" which allows implicit parallel processing; i.e., the programmer need not worry about pre-specification of computational parallelism, or the number of parallel processors which might be available. While it was developed on the Multiprocessor Emulation Facility of the Laboratory for Computer Science, there now exist prototypical versions of computer hardware which can run this Expert System.

The Flight Path Generation process which was implemented by this Expert System can be described as follows:

Given the following information:

1. The spatial organization of the terminal area in terms of runways, navaids, geography, airways, etc.
2. A current forecast for winds (over altitude and area).
3. Current positions, groundspeeds and directions for all aircraft and any performance limitations.
4. A proposed SCHEDULE for both landing and takeoff operations on all runways/airports in the Terminal Area specified to the nearest second.

Then, the Expert System computes:

A set of planned 4-D flight paths for all aircraft from their current position to the runway which are operationally feasible and efficient, and conflict-free. (Reference 1 did not achieve the integration of departure paths.)

The planned flight paths have the following characteristics:

1. Runway operations occur at scheduled times;
2. They are familiar to both pilots and controllers;
3. They are adaptive to tactical corrections by controllers to maintain the SCHEDULE in the face of errors from piloting and winds;
4. They can be changed radically if operational deviations occur (e.g., blocked runway, missed approach, change of runway direction).

5. The Computational Approach

A basic concept for path generation is called a "Pattern." A Pattern is a family of paths created between all the runways and all the entry/departure points at the boundary of the terminal airspace. In computational terms, each path is a sequence of 15-20 "operators" which execute the path. In operational terms, each operator is a "vector" which causes a turn, or speed change, or specifies a new heading or altitude, etc. Each segment of a path has its operator which is invoked at a certain planned time and place in order to define the path.

It is possible to place constraints on a number of segments of any path before generation. A computational strategy called "Constraint Propagation" allows the paths to be built forward/backward from any constrained segment, and parallel computation allows simultaneous generation of multiple paths which meet all constraints.

To reduce combinatorial complexity and avoid issuing frequently revised flight plans to aircraft, a strategy of "Sequential Generation" was followed. This would keep the prior paths fixed while a new flight path was generated which was conflict-free from all prior paths. This was always possible in our research; but in the event that it were not possible, it was also arranged to identify any conflicting path as a candidate to be re-generated after the new path was generated in the absence of that conflicting path.

There is a basic set of Patterns which are used at all major airports today. These are shown in Figures 1 through 4. They are familiar to pilots and controllers, and can be tailored to suit most Terminal Area airspace procedures. Note that they consist of a dozen or more adjustable controls in directions, speeds, and altitudes.

Normally, there are an infinite number of feasible paths which can be selected from each Pattern, so that a variety of "Preference Rules" can be created for each Pattern. These could be re-ranked by individual controllers to suit their style of traffic handling.

A "Generate and Test" approach meant that multiple paths were generated and then tested to see if they were conflict-free. If not, the conflict identification was used to place a constraint on the conflicting segment and the path generation was then repeated. As stated earlier, the conflicting segment of the prior path was also identified. If multiple conflict-free candidates existed, a preference ranking scheme was used to select the "most preferred" path.

FIGURE 1 - PATTERN 1, ARRIVAL-TROMBONE

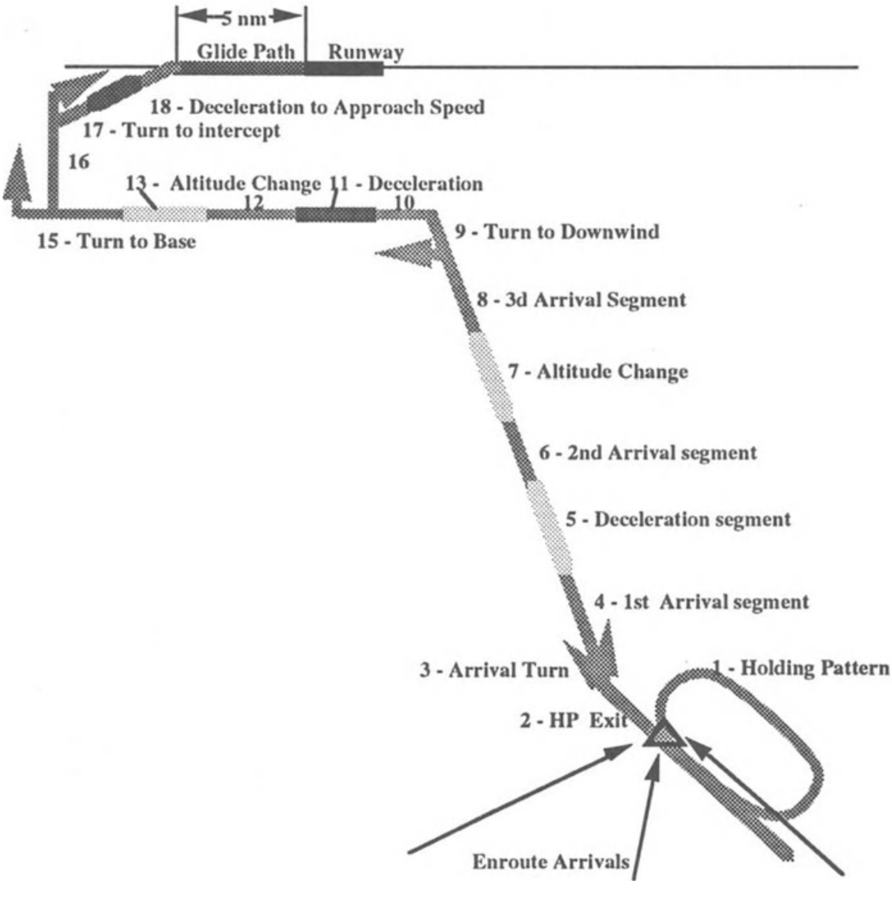


FIGURE 2 - PATTERN 2 - OVERHEAD -TROMBONE

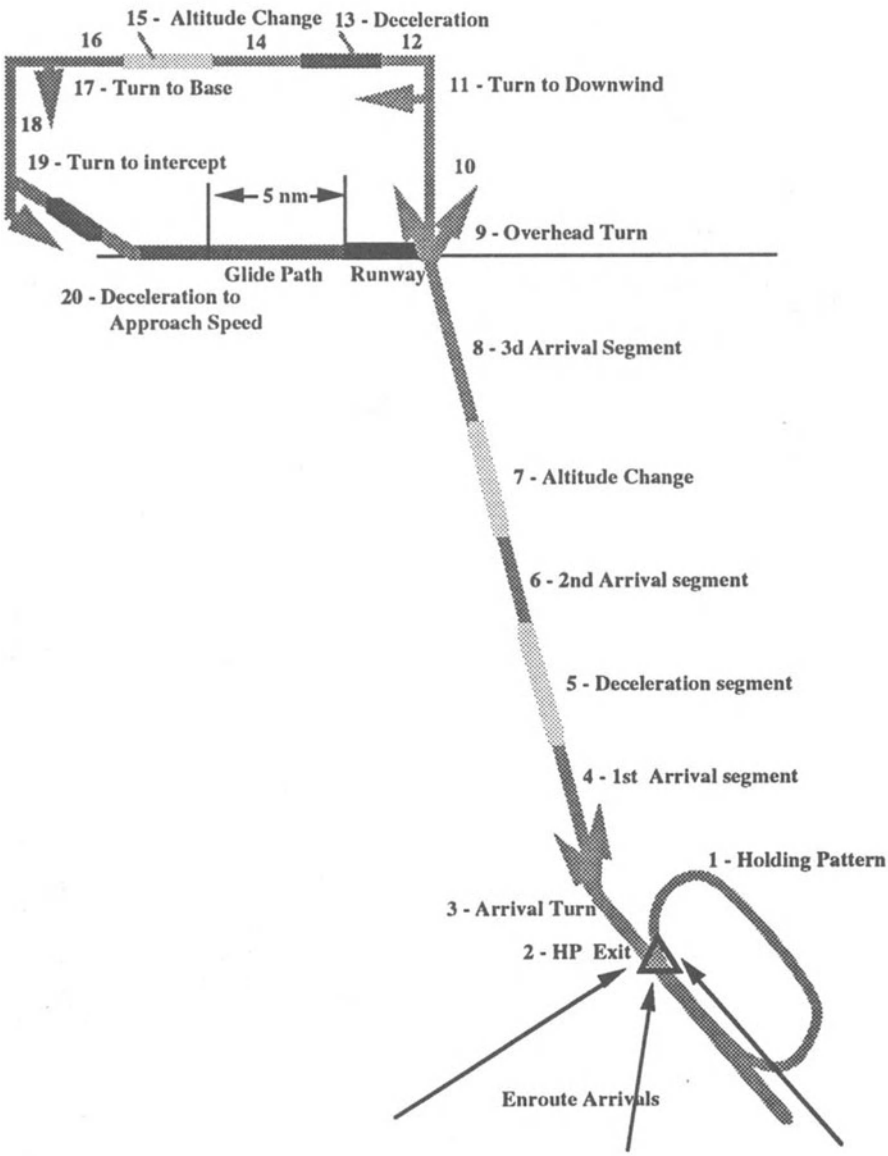


FIGURE 3 - PATTERN 3 - ARRIVAL -TO - BASE

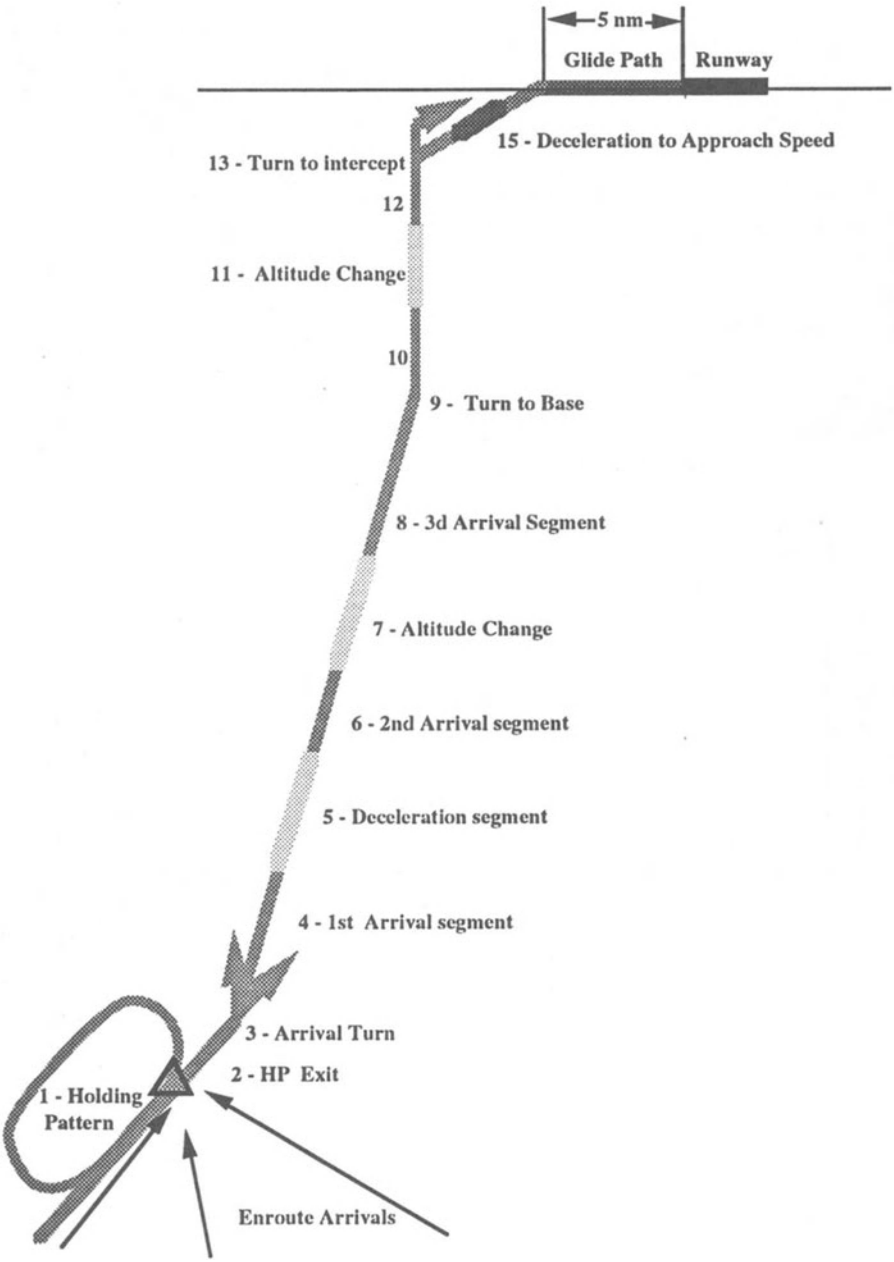
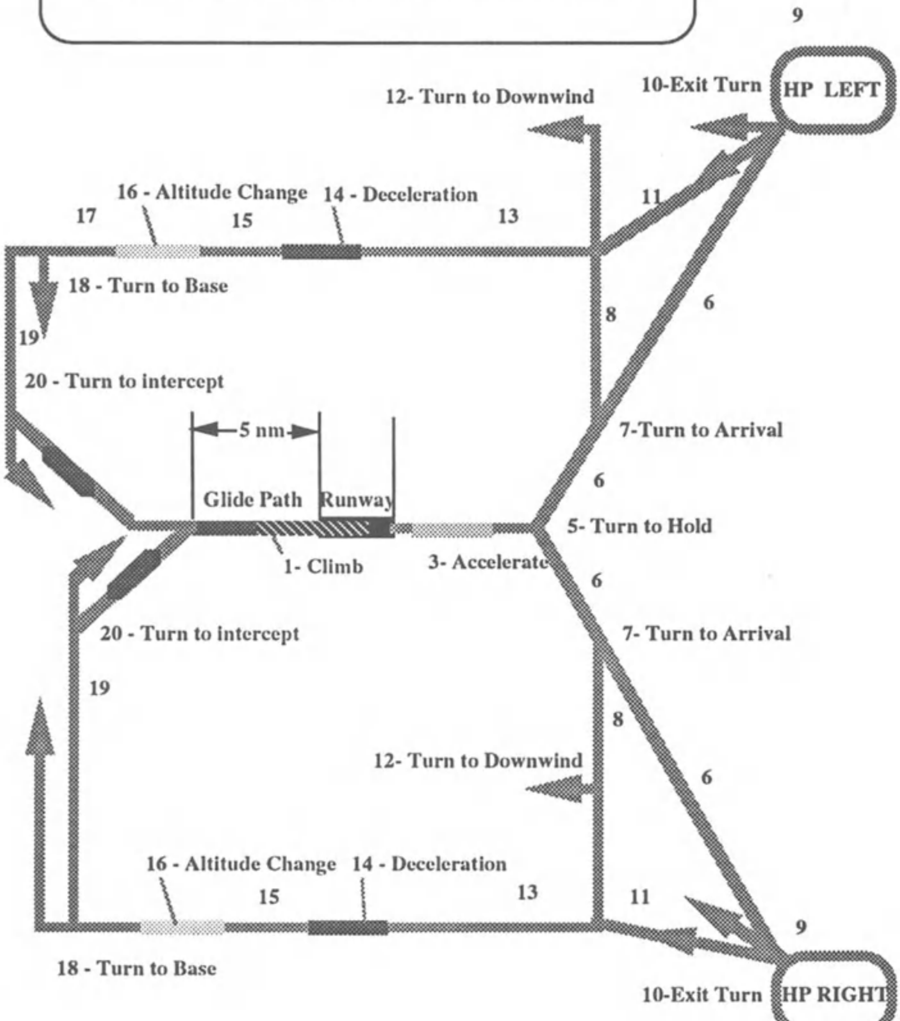


FIGURE 4 - PATTERN 4, MISSED APPROACH



6. Constructing an Operator in Id

Each segment of a path is a transition from an initial state to an end state. State variables are position, time, altitude, speed, direction, and altitude rate. Each path consists of a sequence of Operators which can effect each segment or transition by specifying constraints on the kinematics of the transition.

Each constraint is a declarative statement expressing the kinematic and geometric relationship between the path state variables. Each Operator is a small network of basic Id operations such as add, subtract, divide, less than, etc. Each pattern is a larger network of Operators.

Consider an example Operator which provides a "Constant Speed, Straight and Level" transition from state 1 to state 2 with a minimum duration D. The transition relationships are:

$t_2 = t_1 + dt$	where dt is the segment duration
$x_2 = x_1 + dx$	where dx is the segment x shift
$y_2 = y_1 + dy$	where dy is the segment y shift
$z_2 = z_1$	(constant altitude)
$V_2 = V_1$	(constant speed)
$\Psi_2 = \Psi_1$	(constant direction)

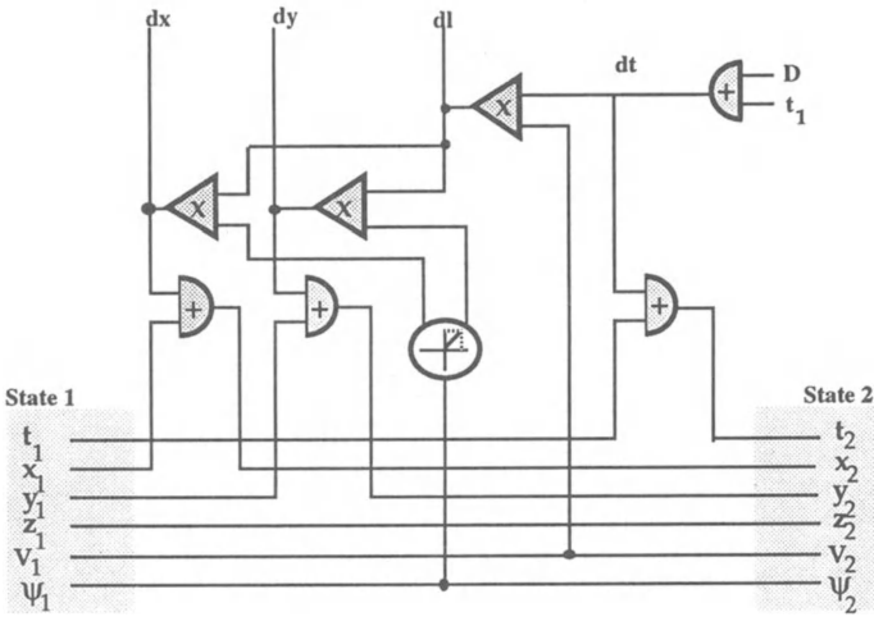
The constraints are:

$dt \geq D$	(duration of segment)
$dl = V_1 \bullet dt$	(distance flown)
$dx = dl \bullet \cos \Psi_1$	
$dy = dl \bullet \sin \Psi_1$	

Given initial speed, direction, and altitude, this Operator will move the path from its initial position to its final position and ensure that the segment has a minimum length or duration. The network representation of this Operator is shown in Figure 5 where each box represents one of the relationships or constraints expressed above. A box represents a computational device which expresses a relationship

FIGURE 5 - CONSTRAINT PROPAGATION

Network Representing a Straight and Level Operator



- each box is a basic operation which will operate in either direction whenever there are enough inputs to define the remaining outputs

- a Pattern is a collection of such Operators, i.e., a larger network

- boxes are automatically assigned to a processor by Id, and can be operating in parallel

- knowledge from either State 1 or State 2 will cause operations to occur in the network; any State information will automatically propagate in the larger network of a Pattern.

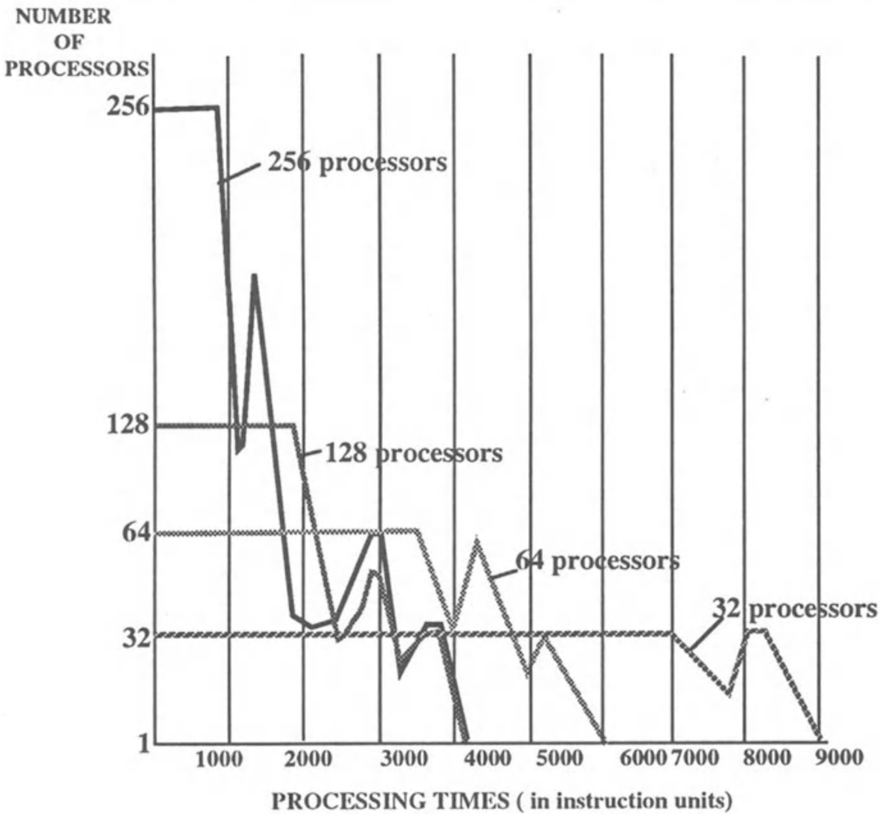
between its variables. There is no direction to the computational flow and no input or output variables. Local Inference is performed at each device such that. as soon as sufficient variables are specified, the device infers the remaining variable(s), and propagates their values to the rest of the network. Thus, the computational sequence depends on the order in which variables are inferred, and consequently on the control or assignment of parallel operations in the Id operating system. It is possible to have redundant but consistent relationships which can determine any variable in different ways.

7. Computational Implementation

There are many operators which are used in all the Patterns; e.g., Turn, Deceleration, Acceleration, Descend, Climb, Intercept Glide Path, Hold, etc. These have to be coded just once. Each Pattern is constructed as a sequence of States connected by Operators, and thus becomes a larger "Dataflow Graph." For a Dataflow Architecture computer, a dataflow graph constitutes a "machine language" and is directly compiled using Id. There are 600 procedures and 5000 lines of code in this Expert System which implements the generation of a Flight Path for landings.

In the programming environment called "Id World" at MIT/LCS, it is possible to measure the "parallelism profile" (i.e., the number of simultaneous processors in parallel computation at any point in time), the instructions performed, memory requirements, etc., for an emulated dataflow architecture parallel computer with any number of parallel processors. Given a processing time for an instruction, it is then possible to estimate the computation time. One set of parallelism profiles is shown in Figure 6 where typically all available processors are initially busy. The number of parallel processors then gradually decreases until the computation is completed. There is a computational critical path length for any computational problem. In this case it is around 4000 instruction periods which can be seen in Figure 6 when the 256 and 128 parallel processor computers both complete the task after roughly 4000 instruction units. There is no point in using a processor with more than 128 parallel units. At a processor speed of 40 MIPS this corresponds to 100 microseconds to compute a new flight path. Several alternative paths can be generated in less than one millisecond, and the preferred path selected. Obviously, Flight Path generation can also be implemented with a much smaller number of parallel processors as shown in Figure 6 for the 64 and 32 processor profiles.

FIGURE 6 - PARALLELISM PROFILES FOR GENERATING A FLIGHT PATH



8. Summary-Flight Path Generation

1. By creating a set of Arrival/Departure Patterns for a particular terminal area, it is possible to design a generic Expert System which can quickly provide feasible, conflict-free Flight Paths for any schedule of runway operations.
2. The use of Patterns ensures that any path is familiar and acceptable to both pilots and controllers.

3. If a modification of Terminal Area procedures is required in the field, it is easy for the local ATC managers to create new Patterns without any re-coding.
4. By adopting the use of dataflow architecture computers and the Id language, it is possible to use implicit parallel processing for path generation. This ensures that paths can be computed in any required minimum time for operational acceptability by simply using hardware which has more parallel processors.

9. An Expert System for Runway Configuration Planning

Reference 2 describes an Expert System for the Runway Configuration Planning Problem developed at the Flight Transportation Laboratory, MIT, by L. R. Hazelton. It was written in Common Lisp and can run on a variety of desktop computers and workstations. Its novelty lies in the fact that it extends the usual domain of Expert Systems by working in a time-dependent logic typical of scheduling systems.

The statement of the Runway Configuration Planning Problem is given below

Given the following input information:

1. A forecast of takeoffs and landings over the next several hours.
2. A forecast of winds and visibilities.
3. A forecast of snow removal requirements.
4. Desirable rules for noise abatement around the airport.
5. Requirements for equipment repair and maintenence.

Then the Runway Configuration Planner delivers;
"a dynamically adaptive schedule plan for airport runway configurations to be operated over the next several hours" (which provides a forecast of runway capacities for landing and takeoff operations).

This schedule plan has the following characteristics:

1. Sufficient runway capacity is provided to minimize delays
2. Number, duration, and timing of runway changes is controlled
3. Noise relief i§ provided at off-peak traffic times
4. Snow removal and maintenence is pre-planned dynamically

10. SoftwareTechnology for Expert Systems -2

Most Expert Systems operate in a very restricted domain. There is a controlled, deterministic, irreversible and static environment where a single operator follows a sequence of steps to find a single solution. However, if the Expert System is used to solve a scheduling problem, it is possible that the usual static logical relationships are replaced by a set of temporal logical relationships which are substantially different.

For the runway scheduling problem there are multiple operators, and a continuous planning process as new information arrives. The controllable actions of the runway schedule plan require time to execute, and consume resources that are not replenished. There are uncertain forecasts of uncontrolled occurrences, and usually some unexpected events or facts. Finally, the consequences of actions may persist for various periods of time until affected by subsequent actions, events or occurrences. This created a need to construct a new, extended form of an Expert System which handled the introduction of time into symbolic logic. These time effects on logic meant that:

1) All knowledge (or facts) must have a time interval of truth.
2) All actions must have an interval in which the occur and they may be irreversible after some "committment time".
3) All inferences have an interval of validity, and may be reversed by a later observation of new facts.

The differences between static and temporal logic can be illustrated by the following:

a) for usual static logical inference, one can make the following statement;

If A and B are true, or if D is true - then C is true.

b) for temporal logic, the inference may be stated as;

If A is true in the interval (t_1,t_2), and if B is true in (t_3,t_4), or, if D is true in the time interval (t_5,t_6) - then C is true in some interval (t_7,t_8).

This can be illustrated in Figure 7 for a typical situation where the logic says that if C becomes true it may persist to be true thereafter. For example, if A states that the temperature is below freezing, and B states that it is raining, then C infers that ice is forming on the runway. C persists until the knowledge that the temperature is above freezing appears. Notice that the inference is directional; no logical rule is present to state that if D is not true, then C is not true, nor is it assumed that if C is true, then D must be true.

The details of an Expert System called Tower Chief designed for handling temporal logic of the type needed for runway configuration planning are described in detail in Reference 2. A simple diagram of the modular elements of Tower Chief are shown in Figure 8. Note that there are separate modules for Truth Maintenence (Time Map Manager), for Runway Scheduling (Scheduler) and for Inference Generation (Temporal System Analyser).

In summary, the requirements for constructing an Expert System to handle the Runway Configuration Planning problem has caused an innovation in software technology. An Expert System has been achieved which can handle "Domain Independent Temporal Planning" where there are simultaneous parallel actions, time bounded actions, a persistence of consequences, and which maintains an historical "Truth Maintenance" over the planning period (i.e., it explains the existence of inferred facts in any time period)

FIGURE 7 - AN EXAMPLE OF TEMPORAL LOGIC

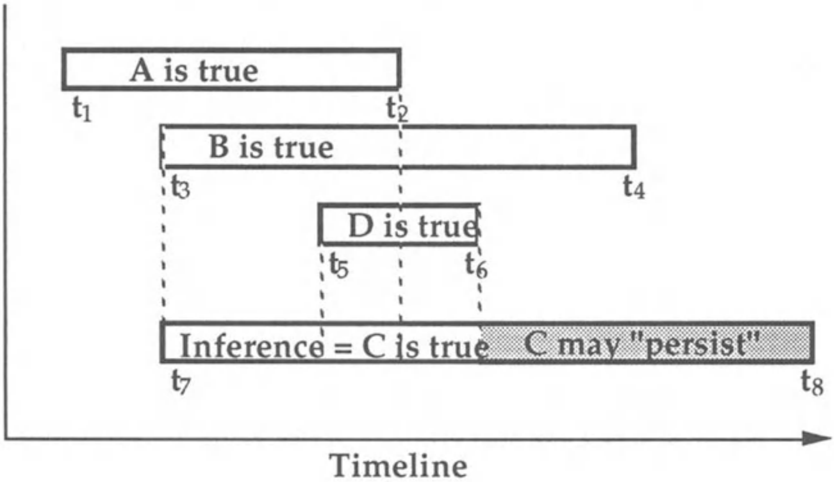
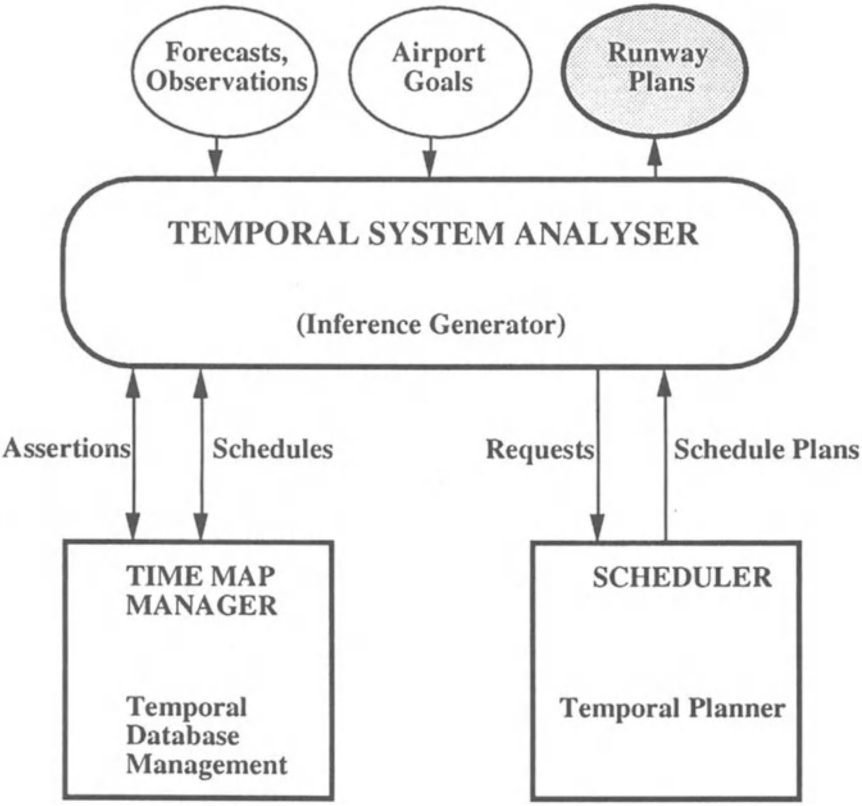


FIGURE 8 - AN EXPERT SYSTEM FOR TEMPORAL PLANNING



References

[1] Sadoune, Michel Marc, *Terminal Area Flight Path Generation Using Parallel Constraint Propagation*, Flight Transportation Laboratory, Report R89-1, MIT, Cambridge, Ma. 01239, USA, May 1989.

[2] Hazelton, L. R. *An Expert System for Temporal Planning with an Application to Runway Configuration Managment*, Flight Transportation Laboratory Report R 91-1, MIT, Cambridge, Ma. 02139, USA, February 1991.

FAA'S NATIONAL SIMULATION CAPABILITY

Andres G. Zellweger

Director, Operations Research Service
Federal Aviation Administration
Washington, DC
USA

Significant progress has been made in testing new air traffic control (ATC) concepts through the use of rapid prototypes. These given us a better idea about how a particular new feature or system ought to be built and how the human operator (pilot, air traffic controller, air traffic flow manager, etc.) should interact with the feature. These prototypes of individual components of the ATC system of the 21st century do not give us an understanding, before a production investment decision is made, of how the new feature interacts with and affects the operation of the total aviation system. As automation complexity on the ground and in the air increases to achieve FAA's vision of the 21st century, this need for better early understanding interaction is becoming more and more critical.
The FAA has established a National Simulation Capability (NSC) to study the horizontal integration of future system components during concept development and research stages. Horizontal integration brings together diverse ATC components (airport, terminal, en route, and oceanic automation; central flow control; flight management systems; cockpit presentations; etc.) in a flexible, real-time simulation environment. The feasibility of NSC has been demonstrated and initial experimental capabilities have been established at FAA laboratories at the Mitre Corporation and the FAA Technical Center. Experiments are being conducted to evaluate new operational concepts (safety enhancements, capacity improvements, productivity tools, changing pilot and controller roles, etc.), human computer interaction and failure modes in a realistic interactive ATC environment of the future.

1. Introduction

The United States Federal Aviation Administration (FAA) is beginning a major thrust to define and develop the 21st century aviation system for the United States. During the 1980's the FAA began a major upgrade of the communication, navigation/landing, surveillance, and air traffic control (ATC) automation systems that make up the aviation system's infrastructure. The National Weather Service has invested billions of dollars in upgrading its infrastructure and will, in partnership with the FAA, be able to provide much more timely and accurate weather data and forecasts to the aviation community in the years ahead. Satellite navigation and communication systems are coming into place and their use in aviation is predicted to increase dramatically by the turn of the century. Increasingly sophisticated avionics in the cockpit will lead to dramatic changes in the way aircraft interact with the rest of the aviation system. Finally, the ever increasing power of the computer will let us give the air traffic controller and flow manager of tomorrow more sophisticated tools to manage traffic more efficiently and to control the traffic with greater safety.

These advances in the aviation system infrastructure provide the basis for meeting tomorrow's aviation needs. Demand continues to grow at a rapid pace, and today's aviation system elements are already showing signs of saturation. Aircraft operators, facing increasing aircraft operating cost, especially cost of fuel, are demanding that the ATC system accommodate more efficient flights (preferred profiles, better flow management, improved flight in instrument conditions, etc.). The complexity of today's ATC system is already such that small perturbations to one part of the system impact the entire system. For example, curtailment of operations at Chicago O'Hare due to bad weather can be felt on both coasts. The challenge of the designers of the 21st century aviation system is to harness the available technology and create a system of people and machines that meet these many demands safely and efficiently.

The FAA has, over the past two years, initiated an effort to define the U.S. air traffic management system of the 21st century. This system will be compatible with the International Civil Aviation Organization defined future air navigation system. The system is characterized by such things as: user preferred profiles; higher levels of automation in the ATC computer system; integrated traffic flow management across airport, terminal, en route, oceanic, and central components of the ATC system; the integration of flight management system with ATC automation system; greater pilot involvement in ATC decisions, quite possibly using information from the Traffic Alert and Collision Avoidance System (TCAS) display; extensive use of satellite navigation and communication services; the use of improved landing systems; and airport surface automation for both safety and efficiency. A description of this system was published in a 1991 FAA report entitled *Concepts and Description of the Future Air Traffic Management System for the United States* (ref. 1).

2. Mission of the National Simulation Capability (NSC)

In early ATC system improvement programs, the diverse elements of the aviation system could be designed, built, tested, and fielded in a straightforward manner with little regard for interoperability among components. This is no longer the case because the technologies that provide the foundation of the future ATC system and the human interaction with the technologies introduce much higher levels of complexity and interaction among system elements. The delays in getting new capabilities to the field and the cost growth of more recent upgrades clearly demonstrate that the system development methods of the past are no longer adequate in today's more complex environment.

The FAA needs an approach to building systems that will ensure the timely development of ATC system elements that interact well to deliver safer, more efficient ATC system services. Such an approach should enable developers of future system components to make use of analyses, simulations, and prototyping throughout the development process. The approach should also allow these components to be studied in the context of the entire system as it will exist when the component is implemented. The context must be the "then current" National Airspace System (NAS) plus all of the planned and proposed improvements that will become operational before the idea being evaluated is fielded. This requires a flexible, dynamic, and robust capability with which candidate systems can be modelled quickly and then evaluated on the basis of their interaction with those other existing and planned NAS components, and their ability to perform their intended function as part of the integrated, future ATC system. Prototyping at the system level accomplishes this. Prototyping at the system level will also support the definition of future system requirements, reduce system interoperability risk, support end-user evaluations of emerging functionality, and expedite the delivery of needed systems to the field.

The National Simulation Capability (NSC) will provide the FAA with this system prototyping capability (see Figure 1). The NSC will be a distributed simulation capability, relying heavily on existing simulations, that will link together people, processes and tools, as required, to perform:

- Concept validation at the system level, ensuring that Research and Development (R&D) projects are identified early and focused on user needs;
- Future systems engineering, planning and development, studying how system elements, including those functional elements allocated to humans, will work together, and how they will deal with unplanned events; and
- Human factors assessments in a highly integrated future automation environment.

Thus, the NSC will enable the FAA to prototype the future NAS itself as a means to understanding and guiding its development.

3. Use of the NSC

One of the things that has become quite apparent over the past decade is that prototyping is a critical step in the development of new ATC system elements. Prototyping, or real-time human in the loop simulation, has been used successfully for many different aspects of system development: design of the computer human interface (CHI); design of algorithms and system functionality; laboratory and field evaluation of more mature new system elements; training and

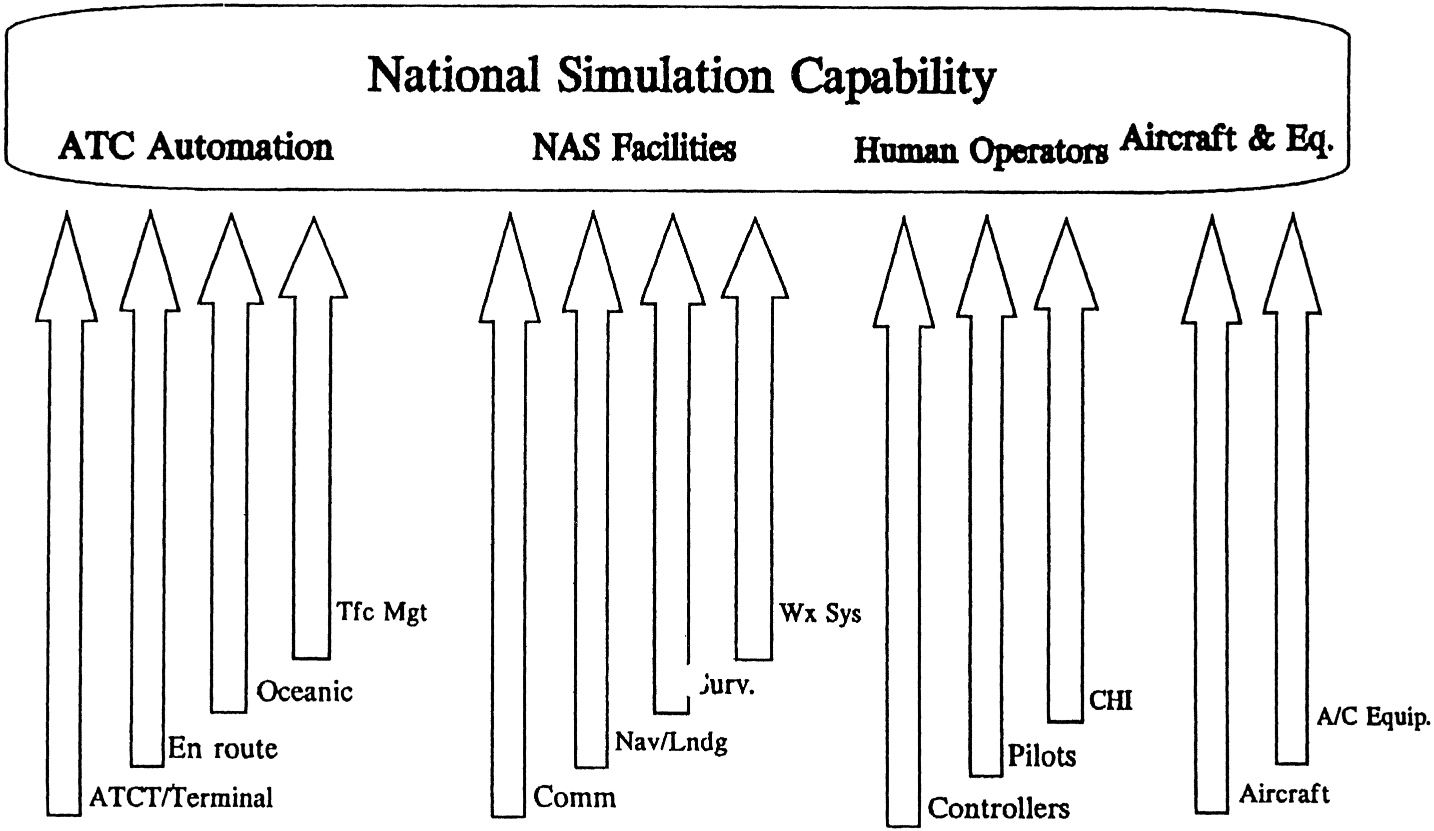


Figure 1. NSC Role: NAS Horizontal Integration

facilitating user acceptance of new features; procedure development; and even for demonstrations to gain political and financial support for programs. Figure 2 lists just some of the many recent successful uses of prototyping in aviation system development.

The NSC will go a step beyond these traditional simulations. The one underlying characteristic of previous simulations was that each individual simulation was oriented to a single system element, such as the design of a controller display for presenting flight data or the integration of conflict alert into the en route ATC system. Such "vertical" simulations will not be adequate to develop the next generation aviation system because of the complexity and far reaching impact of the interactions of the different system elements. A change of a function in one part of the system impacts functions in many other parts of the system. For example, management of traffic flow will require the correct interaction of computers and people in the cockpit, in the terminal and en route ATC facilities, and in a centralized flow control facility. The NSC will provide the capability to simulate the interactions of many system components. Thus the NSC is a laboratory to study the "horizontal" integration of people and machines that will comprise the 21st century aviation system (Figure 3).

The NSC will provide the means to study and solve problems in three different areas to complement the traditional "vertical" or project-oriented R&D. First, it will help determine the proper allocation of functions to the interacting system elements and the proper definition of the interfaces between the elements. Second, it will allow the exploration of how well the system elements (people and machines) interact to form a coherent total system. The interactions of the human operators (pilots, controllers, etc.) with each other and with their automation aids represents an important aspect of this second area of R&D. Third, the NSC will be a vehicle to study how the system behaves under failure conditions. A variety of failures in human and machine behavior and interaction can be simulated in the NSC to ensure that aviation safety will not be compromised when new features are added to different parts of the aviation system.

- MAESTRO and COMPAS development
- Pre-Departure Clearance at FAA
- MLS evaluation at NASA Ames
- AERA 2
  - CHI development
  - Conflict resolution rule development
  - Procedure development
- CTAS development at NASA Ames

Figure 2. Examples of ATC Research and Development Prototypes

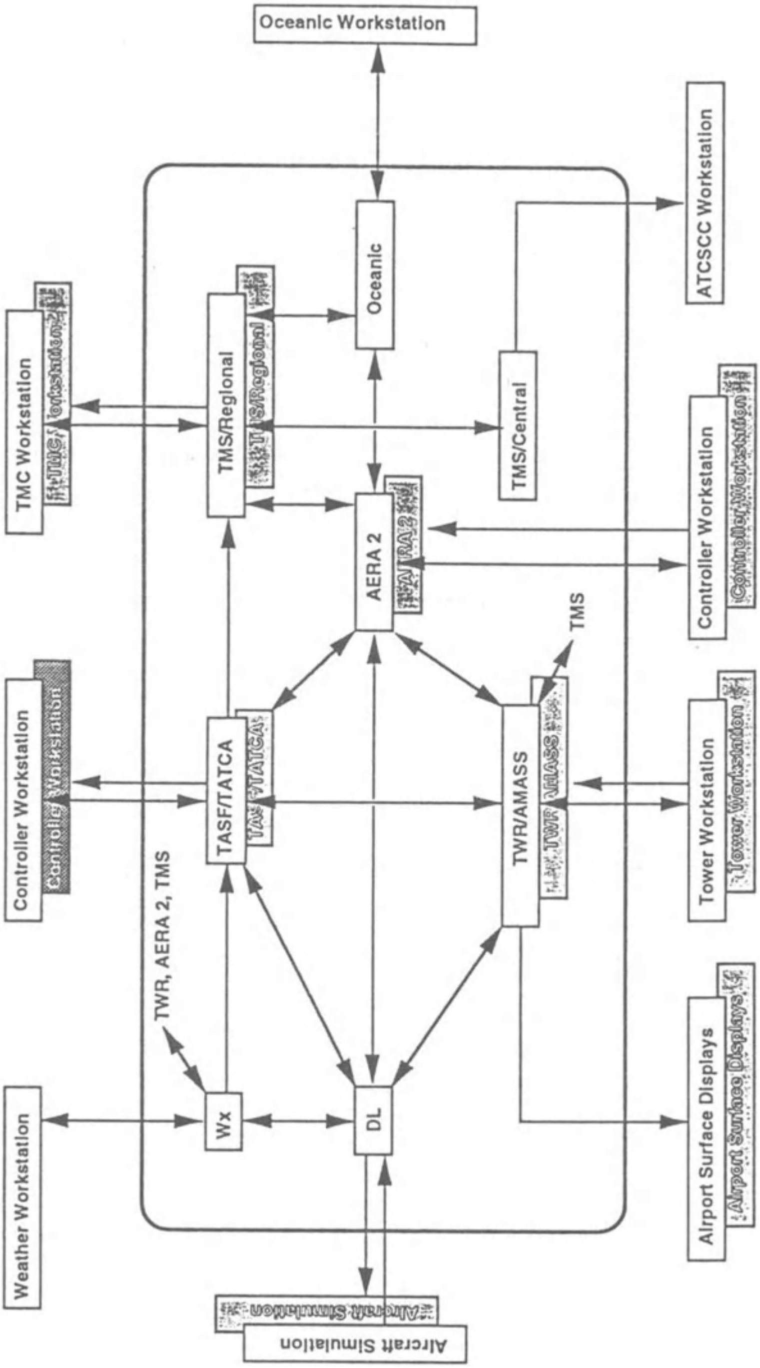


Figure 3. Horizontal Integration of Components

The NSC will be used throughout the R&D phase of a project. Early in the conceptual stage, the NSC provides the opportunity to explore a concept in the context of the future ATC system in which it might be used. Thus the value of a concept can be determined and decisions made about the potential of further R&D investment. As concepts are refined and R&D progresses, FAA will conduct system engineering to understand the operation of the R&D products in the total system. The NSC will complement this analytic work by addressing the interoperability, CHI, and failure mode analyses described above. Thus NSC will ensure that problems will be found and resolved early in the life cycle rather than during the system procurement or operational test phases when correction of mistakes is much more costly in time and money.

An example of the use of the NSC during the early (requirements definition) phase can be drawn from the FAA weather program. Better actual and forecast weather information will be useful to all parts of the ATC system. Better wind data will permit better route allocation and more precise metering of aircraft. Knowledge of when and where severe weather cells need to be avoided will allow better planning of traffic flows. More precise airport forecasts will permit traffic flow planners to develop more efficient solutions to potential bottlenecks. FAA system engineers must decide the parameters that define the weather data (type of data, accuracy, cell size, frequency of forecasts, update rates, etc.) for each of these interacting ATC system elements to balance optimal system performance with cost of the weather products. The NSC by simulating the delicate interaction of the people and machines involved in the use of weather data can support the system engineers in determining the proper weather data requirements. The NSC will also permit system designers to explore how weather information should be presented to and transferred among the controllers, flow managers, pilots, and flight dispatchers and how these people will subsequently deal with that information.

A second, quite different, example of the unique capability of the NSC to support the design of the 21st century aviation system relates to the introduction of new types of aircraft. Tiltrotor and tiltwing aircraft represent a unique challenge for the future. With their ability to take-off and land vertically coupled with their ability to cruise at the same altitudes and airspeeds as present day turbo-prop aircraft, they do not fit entirely into the patterns of operations of either rotary wing or fixed wing aircraft. The challenge will be to design airspace, procedures, and facilities which take maximum advantage of their capabilities in developing the inter-urban transportation networks of the future. NSC will be able to address these issues and explore the many options available for integrating these aircraft into the future NAS.

4. Initial NSC Activities

In early 1990, FAA started an ambitious 2-year activity to demonstrate the feasibility of the NSC concept. A prototype of an NSC laboratory was developed for FAA by the Center for Advanced Aviation System Development (CAASD), an FAA sponsored Federally Funded Research and Development Center. The objective of this laboratory, called the I-Lab, was to demonstrate the feasibility of the concept of "horizontal" integration simulation and to gain some early technical experience with the building of such a laboratory and to illustrate and clarify the NSC experimentation process. Emphasis was placed on documenting lessons learned that illustrate how to build an ATC Simulation Lab and to provide guidance for the development of the ultimate geographically distributed NSC. This included investigation of hardware and software architectures, communications mechanisms, database structures, incorporation of legacy software (i.e. software previously developed for other purposes), experimentation requirements, and establishment of guidelines for functional capability and fidelity. The latter was important in gaining an understanding of the tradeoff between the degree of fidelity required for a particular experiment and the cost and time necessary to incorporate the necessary features for a particular experiment into the lab.

In August of 1990 CAASD demonstrated the successful integration of six different human-in-the-loop real time simulations, developed independently, on different platforms, in different languages, and for different purposes. A single aircraft generator which "flew" an aircraft scenario and was also interfaced to a simulated pilot station drove the entire simulation. Aircraft flew through the entire system and one could clearly see how changing of control parameters in one of the simulation elements affected other elements. This milestone established the technical feasibility of the NSC concept.

The most difficult aspect of this initial NSC effort was the development of the dynamic interfaces among the components. New software had to be developed to provide central simulation control, synchronization, and time reference and for the distributed inter-process communications.

The second phase of the two year feasibility demonstration was the extension of the I-Lab to the point where it could be used to perform a substantive set of ATC integration experiments. The hardware suite was extended (Figure 4) and a number of new legacy components were added (Figure 5). Figure 6 presents a graphic depiction of these I-Lab components. In addition to the legacy components a cockpit/graphic services module, based on commercial, off the shelf (COTS) software was added to I-Lab (Figure 7). An interesting aspect of this is that the 3D graphics viewer could be used both to give an out the cockpit view and a view of the ATC tower (Figure 8). It is fascinating to "look" out of the tower and to see the realistic representation of aircraft arriving, departing, taxiing, and at the gate, as simulated by the SIMMOD component.

Some of the legacy components required significant changes, others were used as developed by other FAA R&D efforts. The en route component, for example, took a single sector Automated En Route ATC (AERA) prototype (Figure 9) and extended it to a multi sector simulation (Figure 10). This was necessary because some of the early metering experiments planned for I-Lab required 50 en route AERA sectors to provide the appropriate geographic extent and volume of traffic. The lab is able to run 2 or 3 manned AERA sectors and the rest

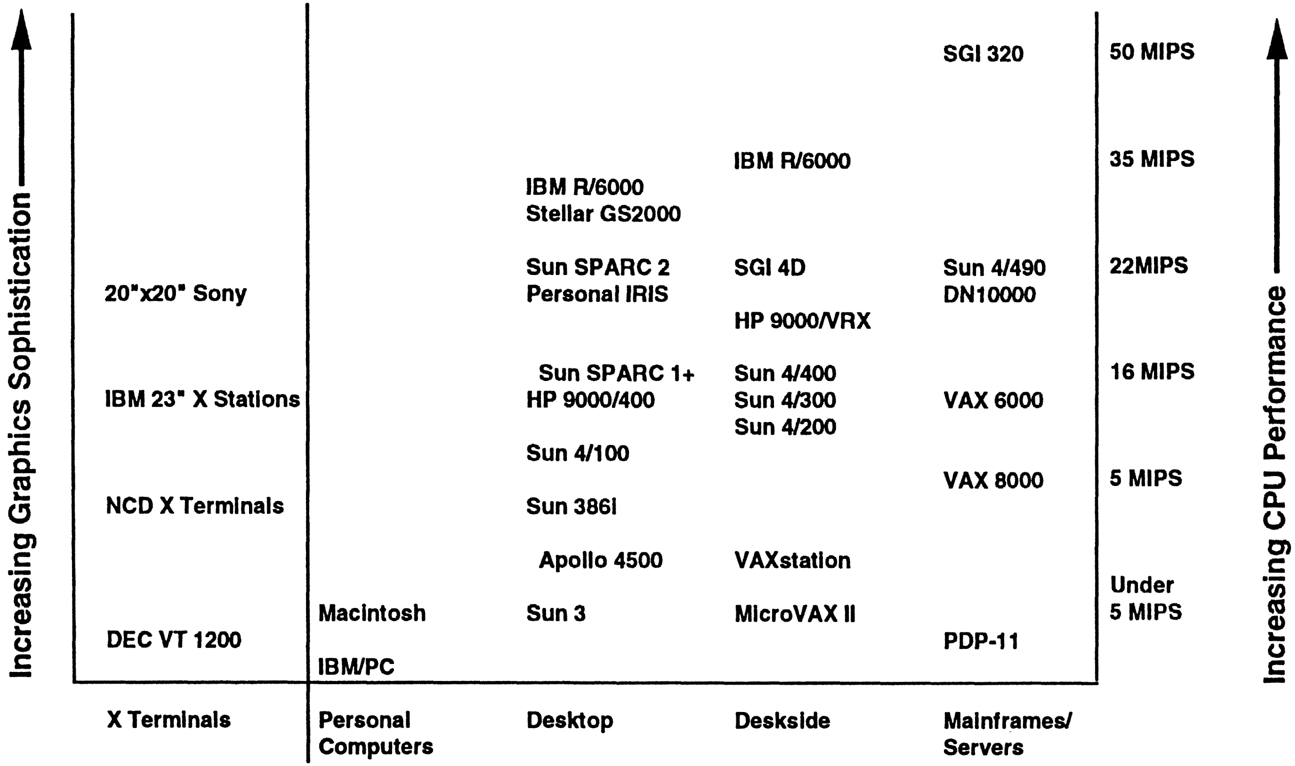


Figure 4. I-Lab Hardware Suite

Component	Type *Platform/Language*	Purpose of Development/Use
AERA 2 Automated En Route Air Traffic Control	Prototype *VAX/PL1*	Prototype and refine CHI and Conflict Prediction/Avoidance components. Simulates traffic movement.
TASF Terminal Area Simulation Facility	Simulation *Apollo/Pascal*	Build Simulation platform to support terminal automation development.
SIMMOD FAA's Airport Surface Simulation Component	Simulation *Sun/Simscript*	Developed by CACI and the FAA to be a general purpose simulation of airport. Ground movement simulation.
CTAS Center-TRACON Advisory System	Prototype *Sun/C*	Developed by NASA Ames to prototype terminal traffic management concepts. Provides metering and controller aids.
DSP Departure Sequencing Program	Operational Prototype *Apollo/Pascal*	Operational evaluation and requirement specification. Coordinates departures from multiple airports.
PDC Predeparture Clearance	Opnl. Proto. *Sun/C*	Operational evaluation and requirement specification. Issues auto-clearance.
AGD Adaptive Ground Delay	Prototype *Apollo/Pascal*	Proof-of-Concept. Automatically updates ground delay

Figure 5. I-Lab Legacy Software Components

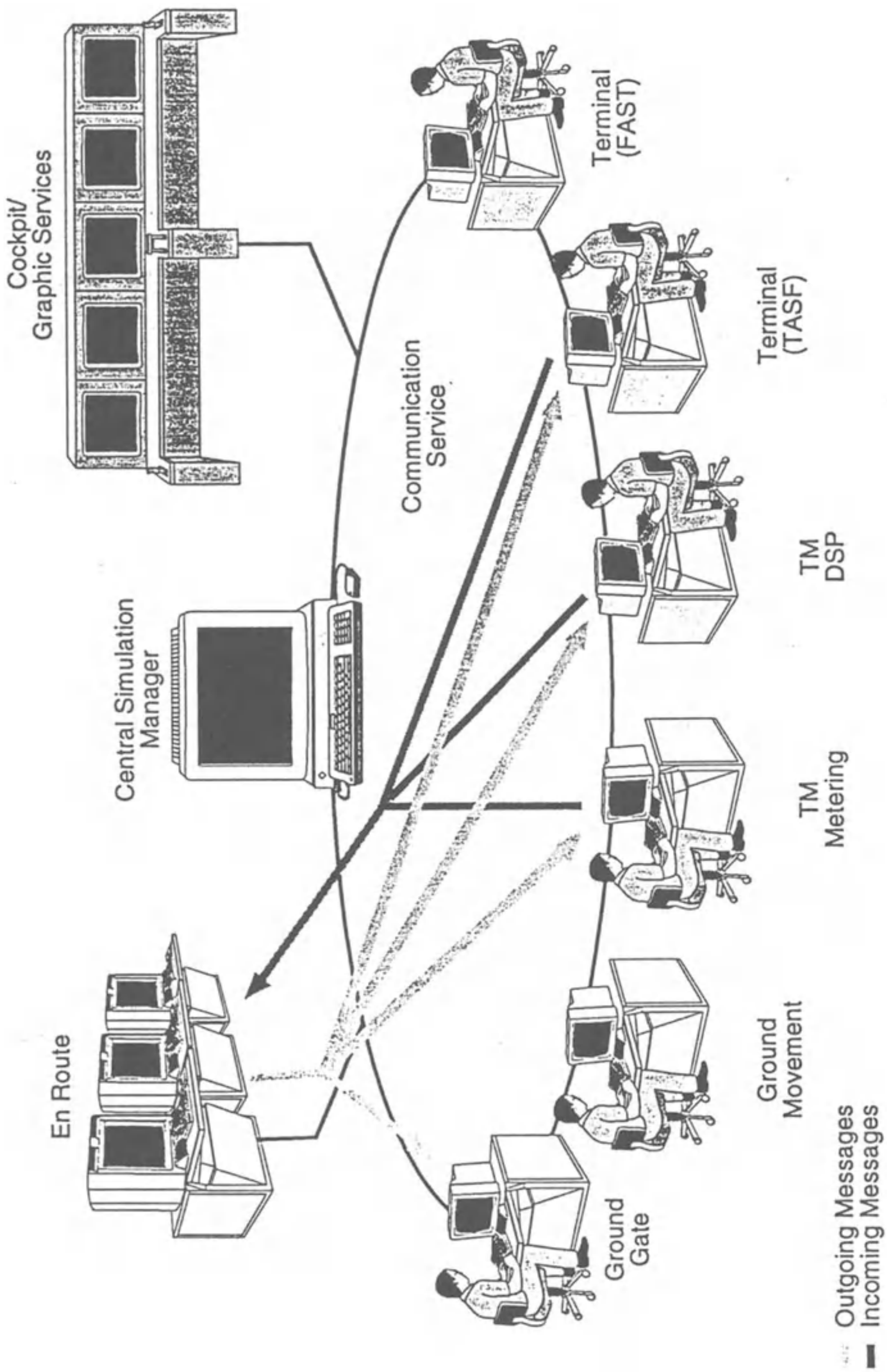


Figure 6. I-Lab Functional Components

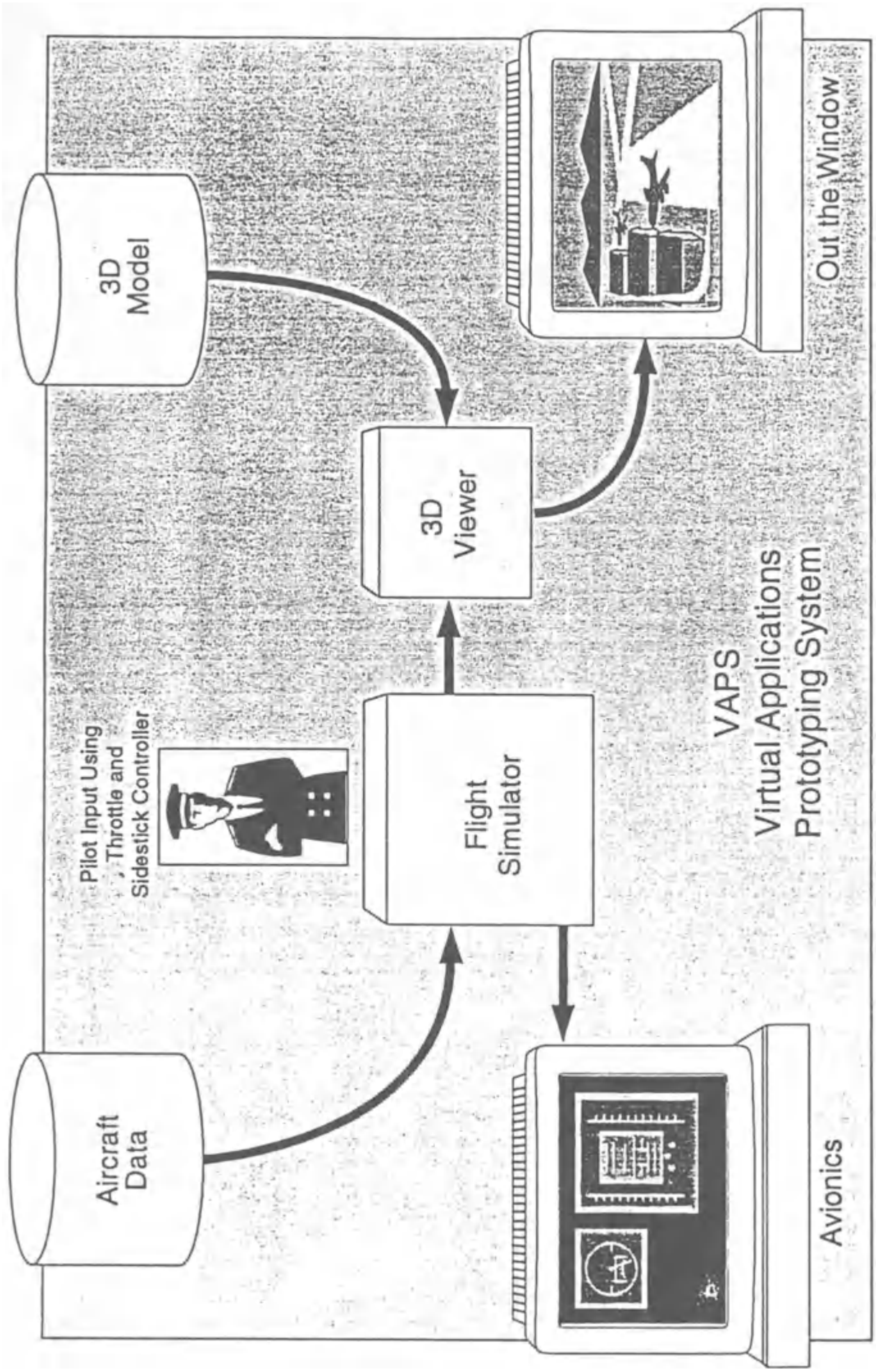


Figure 7. I-Lab Cockpit/Graphic Services

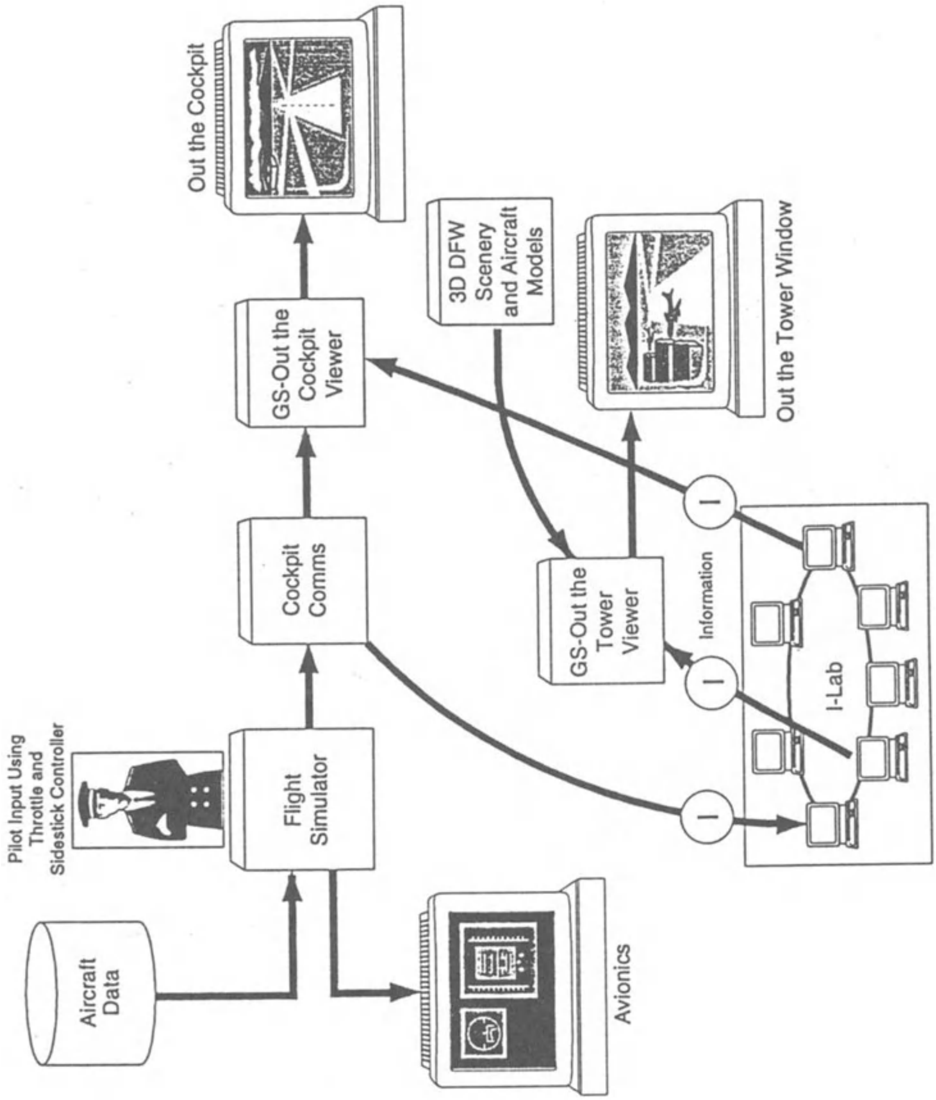


Figure 8. I-Lab Cockpit/Graphic Services - Detailed View

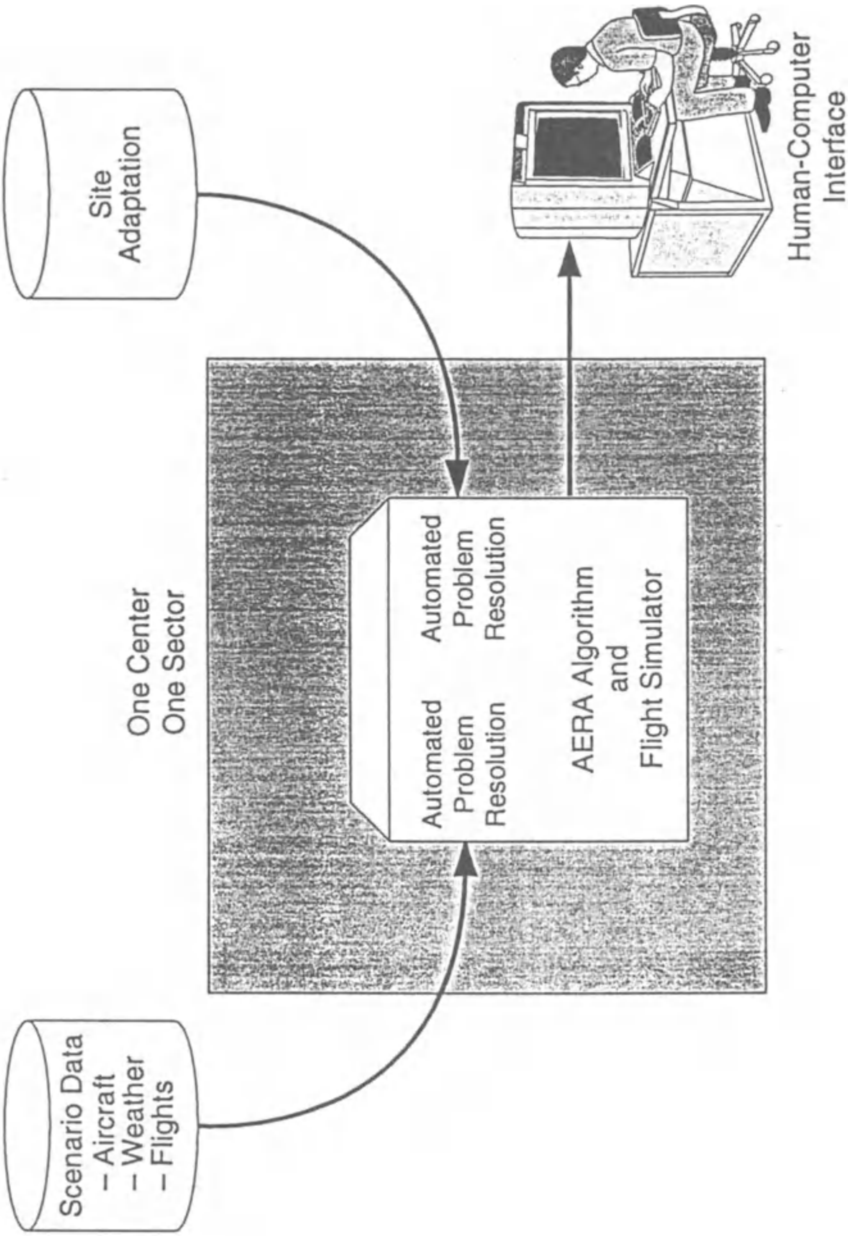


Figure 9. Single Sector AERA Prototype

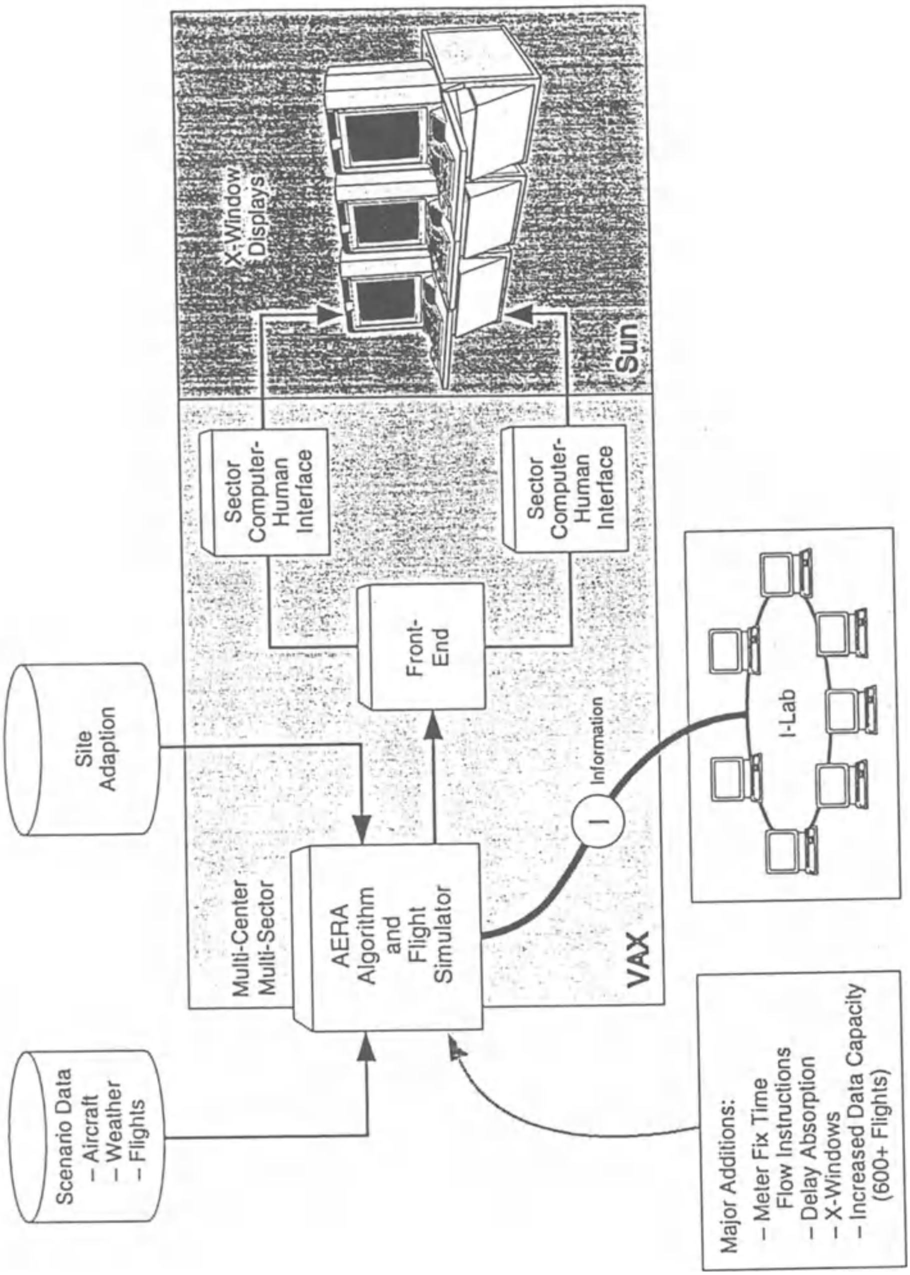


Figure 10. Multi-Sector AERA Prototype

use a "simulated controller" (controller actions are simulated by simply having the computer select the preferred resolution generated by the AERA algorithms).

The Center-Tracon Automation System (CTAS) presented a different problem. Here we had a case where an active R&D effort was periodically issuing new software releases. It was important for the proposed NSC experiments to keep the current version of the CTAS software, so CTAS was incorporated into the I-Lab with minimal changes. This presented an interesting challenge, both from a software viewpoint and from the perspective of close coordination and cooperation between I-Lab and a "vertical" R&D effort.

The I-lab experience to date has taught us a number of valuable lessons:

- Incorporation of legacy code requires design tradeoffs and a thorough understanding of parts of the code and the interfaces. The CTAS experience drove home the point that standards and guidelines for prototypes that are being developed in FAA R&D labs are needed to make incorporation of legacy code in the NSC easier;
- Experimental objectives need to be well understood early in the process so that the laboratory infrastructure can be built with fidelity sufficient for the problem at hand;
- Staffing levels and skills are non-trivial. ATC knowledge and legacy code experience are required in addition to expertise in real time, human in the loop simulation and validation;
- Focus must be placed on the process of developing and using the lab. Because many R&D and customer organizations are involved, internal and external coordination, while often difficult and time consuming, is critical. When experimentation is required to resolve and understand an issue, one must provide enough lead time to ensure that the infrastructure needed can be in place and that the experiment is planned properly and well ahead of time.

5. NSC Infrastructure

A fairly clear picture of an NSC architecture and infrastructure has emerged from the work in I-Lab, from numerous discussions with others involved in related simulation and prototyping activities, and from a projection of the funds that FAA will be able to dedicate to the NSC. The concept has evolved from the National Simulation Laboratory (NSL) proposal by J. Lynn Helms and the FAA R&D Advisory Committee (ref. 2) for a central laboratory (with links to remote labs for distributed experiments), staffed by 150 to 200 people with an annual budget of $20-30 million.

The NSC concept is for a much more distributed system than envisioned by the Helms Committee, and builds more heavily on existing facilities at CAASD (especially the I-Lab) and at the FAA Technical Center in Atlantic City. The capabilities of the I-Lab have already been described. The FAA Technical Center (FAATC) has a number of existing facilities, some of which are already interconnected, that can be used for addressing NSC issues. These include: the NAS System Support Facility, which contains en route, terminal, and Flight Service Station components; an Oceanic Development Facility (currently under development) that will support end to end test and evaluation of the future Oceanic ATC System, including pilot and controller; a reconfigurable cockpit simulator; a data link testbed; a human factors laboratory; and interfaces to operational ATC systems as well as to high fidelity cockpit simulators at NASA Ames and other locations. Interconnection of FAATC capabilities as well as additional infrastructure development at both primary NSC nodes and in other locations will be driven by the critical issues that must be addressed by FAA as it plans and builds its future ATC system.

The NSC will grow incrementally over the next several years. In addition to experimental needs, three factors will shape this growth. First, to allow the NSC to adapt to the needs of different experiments, the design must allow simulation at different levels of abstraction (Figure 11). The NSC must be capable of representing simultaneously

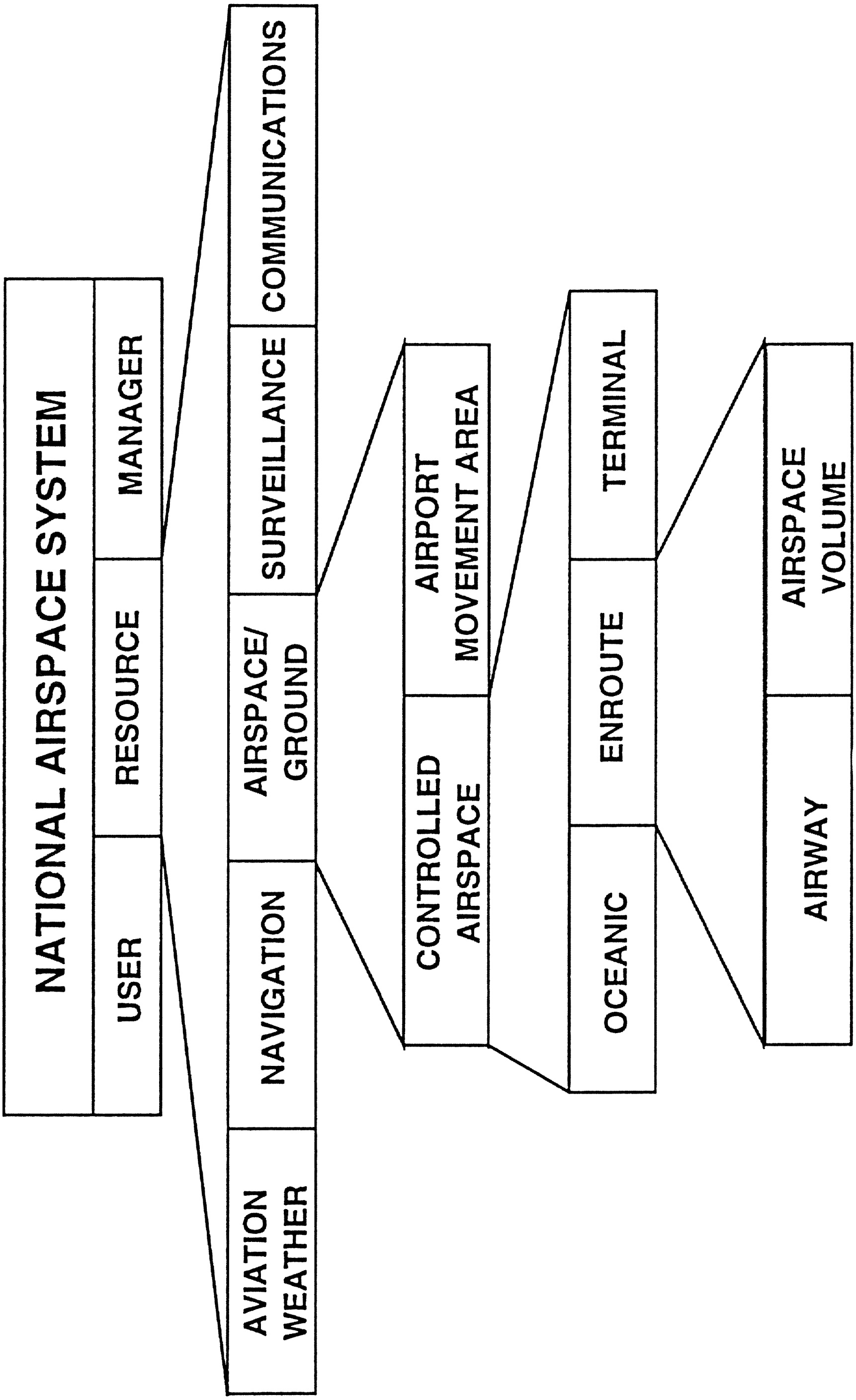


Figure 11. NSC Architecture: Levels of Abstraction

selected portions (i.e. selected areas, functions, objects,etc.) of the future NAS with high fidelity and others with lower fidelity. The architecture must provide a framework for varying and substituting components representing different concepts, design fidelities, and levels of abstraction, depending on the specific problem being addressed. Second, to make the NSC a usable capability, a significant investment in supporting infrastructure is needed in addition to the simulation elements to support experiment preparation, data collection, and the analysis of experimental results. Figure 12 provides a good summary of the functional subsystems required for the NSC. Third, to facilitate connection of remote simulations and to incorporate legacy software, common interface protocols and development standards will be developed for the NSC enhancements and for use by developers of software and hardware in "vertical" ATC R&D labs. The standards will include guidelines for use of the "layered" software model employed for message handling, data management, and graphical display in the I-Lab. Clearly, these architectural considerations are, to a large extent, driven by the desire to modify and expand the NSC quickly and at a relatively low cost as new requirements are identified. Figure 13 summarizes the preparation time expected for different types of NSC changes.

6. First Experiment

The first NSC experiment was completed in the summer of 1992. It examined the interaction between AERA 2 and the Traffic Management Advisor (TMA) portion of CTAS. Specifically, the question explored was how far in advance of an aircraft's expected entry into the terminal airspace TMA should inform AERA 2 that an aircraft must be delayed.

Both controller workload and the risk of unnecessary delay are minimized if the arrival schedule does not have to be revised after meter fix times have been issued. To maintain schedule stability, the developers of TMA proposed freezing an aircraft's scheduled time and

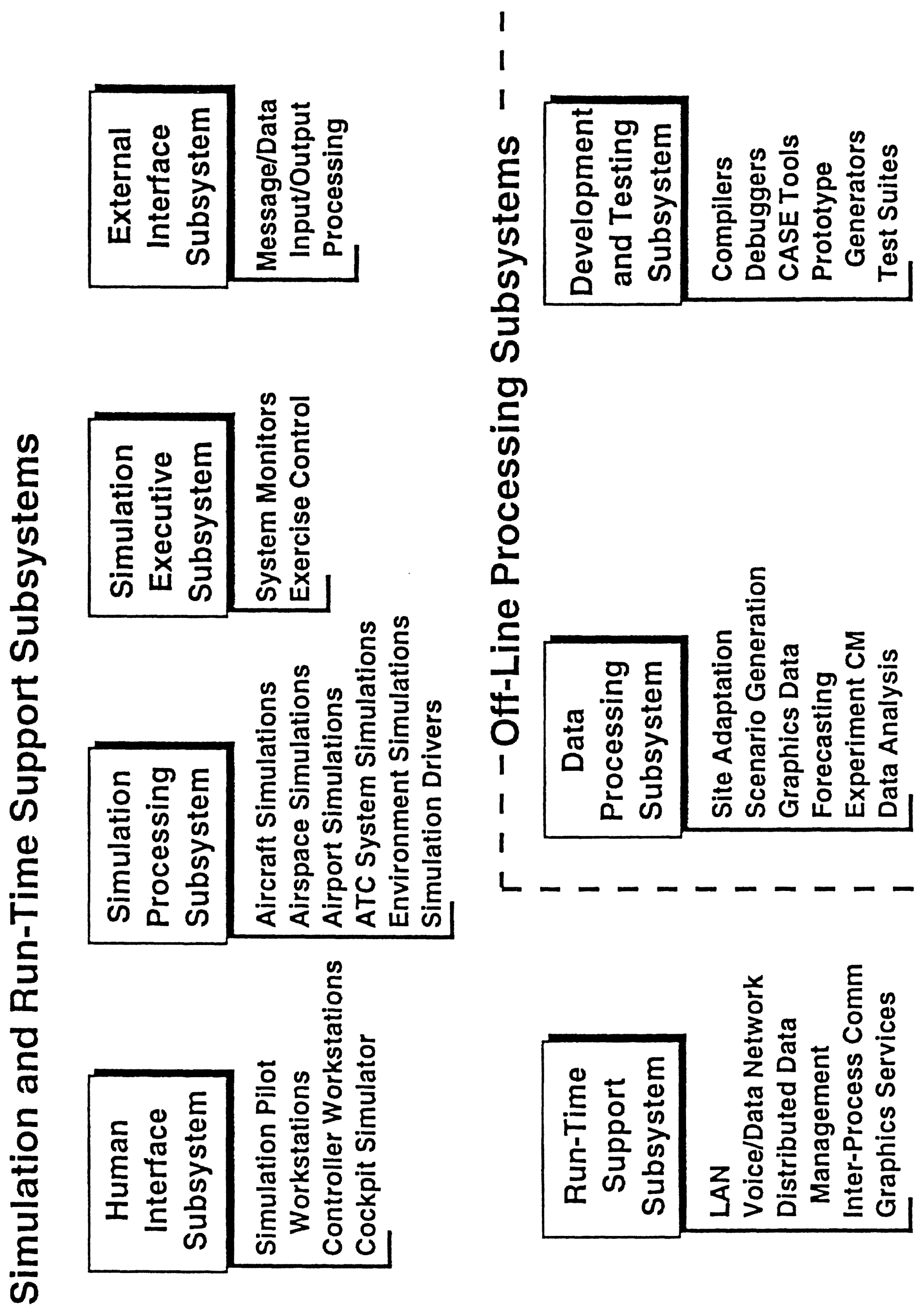


Figure 12. NSC Architecture: Functional Subsystems

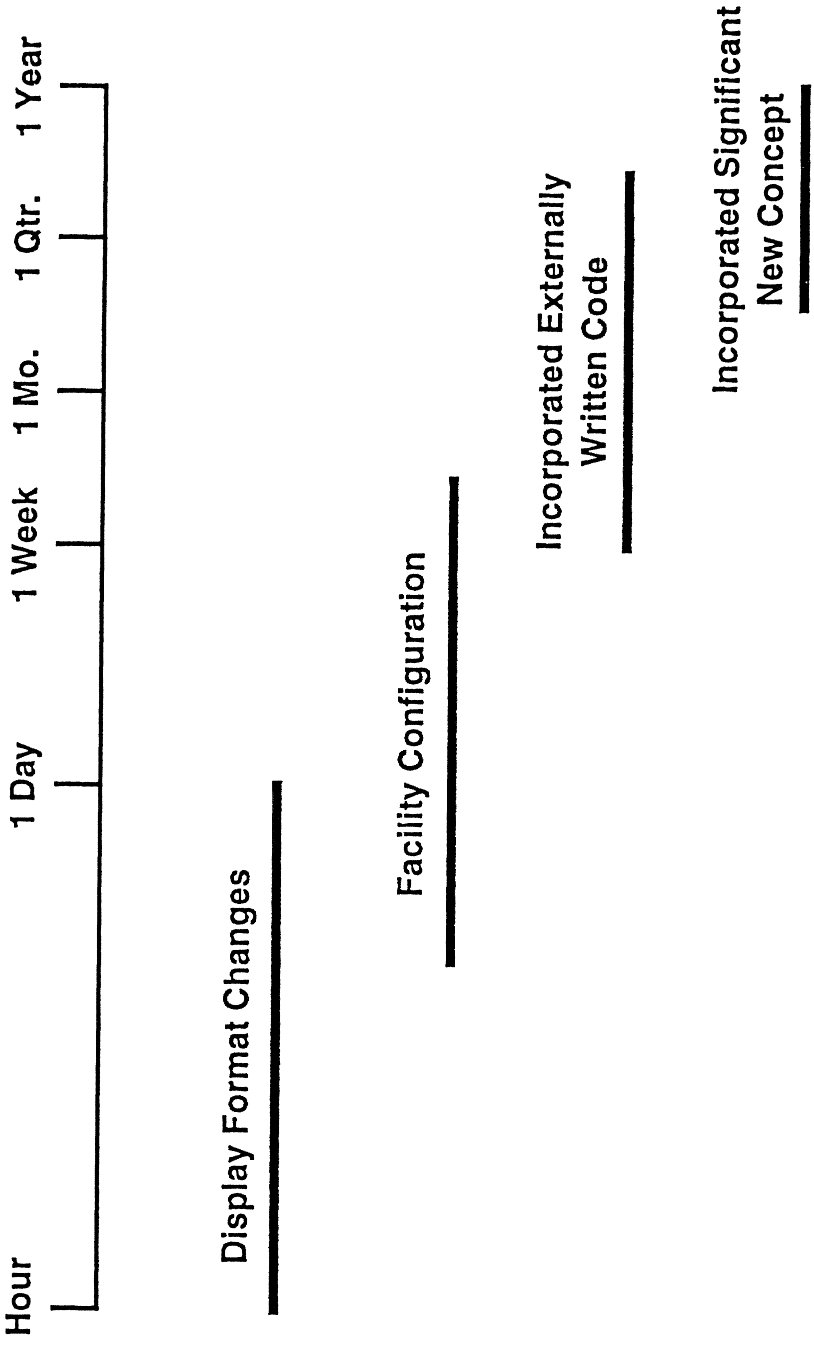


Figure 13. Time Needed to Incorporate Changes into the NSC

making it available to en route controllers 30 minutes before the aircraft would cross the meter fix. Controllers have more choices and may be able to use less disruptive maneuvers if they know the meter fix time well in advance. Thus the designers of AERA proposed that the meter fix time be given to en route controllers 90 minutes before meter fix crossing.

The I-Lab, with its AERA 2, TMA, and terminal ATC simulation, was selected for the experiment. The TMA prototype scheduled aircraft when the aircraft were 30, 60, and 90 minutes from their meter fixes. The effects of those three conditions on a simulated arrival rush into Dallas-Fort Worth International Airport (DFW) were examined. A variety of measures representing system performance, burden on en route and terminal controllers, and burden on pilots were collected.

The data from the experiment suggests that TMA should schedule aircraft fairly close to the terminal area. The 30 minute scenario fared best in terms of average delay and equitable assignment of delay to aircraft, but there were no differences in the other measures. Further work is needed to determine the impact on the large number of short haul flights and the procedure for handling of pop-up in the initial experiment. The experiment also pointed out that proper functioning of TMA with AERA will require careful design and additional experimentation of the interaction of the people operating TMA and AERA.

7. Future NSC Activities

Planned NSC activities fall into two categories - general program activities and experimentation. Program activities (Figure 14) include program planning and documentation, review by a customer oversight group called the Planning and Review Board, laboratory development, and experiment planning. Experimental activities are summarized in Figure 15. Already completed is an experiment that explored the interaction of the CTAS Traffic Management Advisor

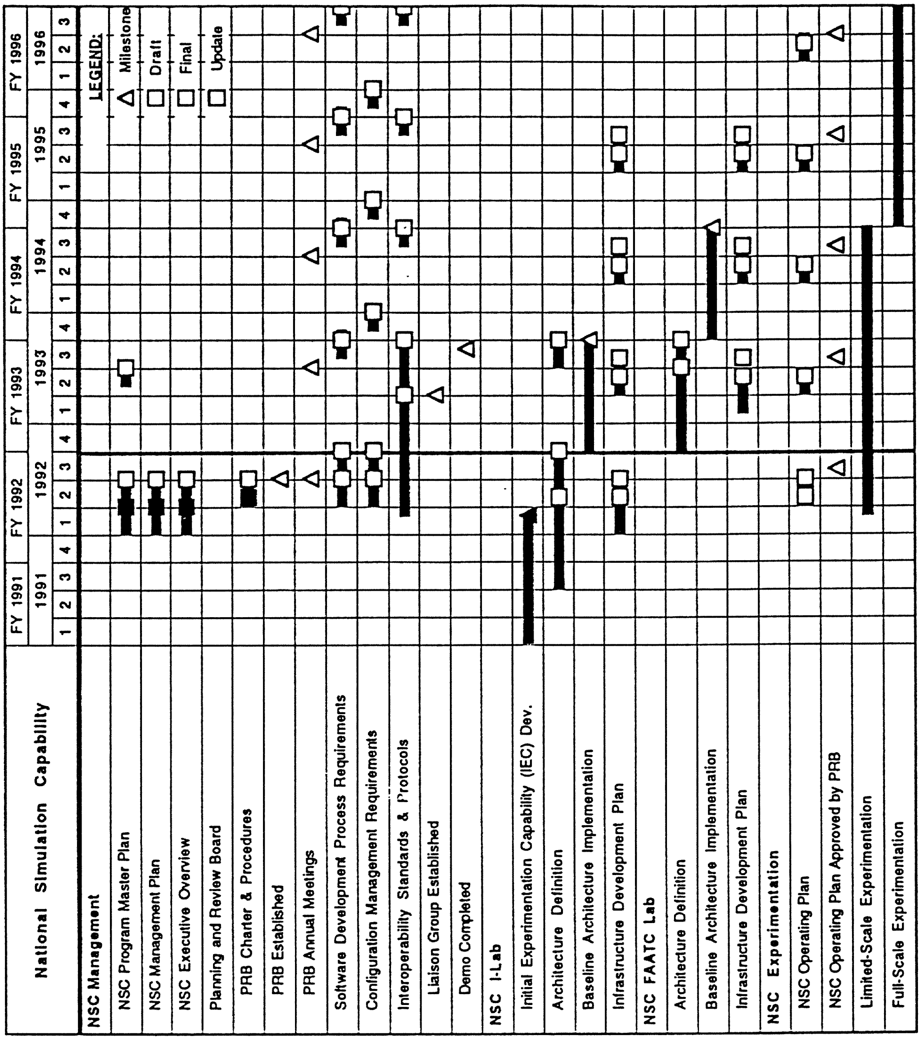


Figure 14. NSC Program Schedule

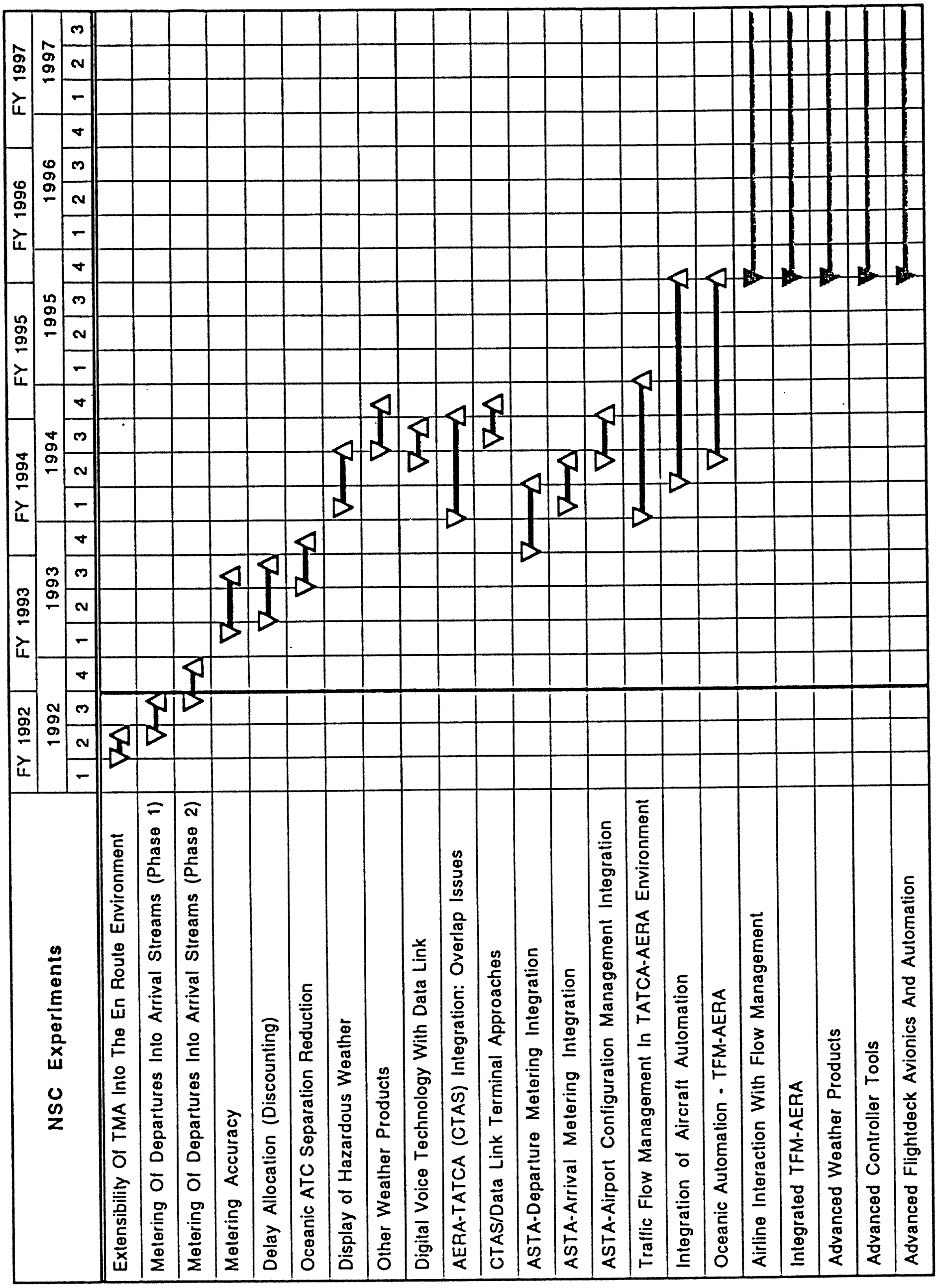


Figure 15. NSC Experimentation Schedule

with AERA 2. In 1993 the NSC will explore metering in the future ATC en route environment, integrated ATC concepts for the CTAS Final Approach Sequencing Tool, and future concepts for Oceanic ATC separation reduction. In 1994 the NSC will begin to look at the system issues involved in future weather product integration, voice data link interactions, AERA - CTAS integration and flow management interaction, and surface traffic automation (ASTA) interaction with future approach and departure concepts, Oceanic automation interaction with future en route and flow management concepts, and finally, ATC interaction with flight management systems.

8. Conclusion

The FAA's National Simulation Capability will add substantially to the FAA's efforts in designing and implementing the 21st century aviation system. The NSC will point the way to the most promising concepts and will reduce the development risks by giving researchers an early look at the "horizontal" interactions of many system elements being developed in vertical" R&D laboratories. Finally, the NSC will provide an international resource where people can try out ideas and work together to learn from one another. As the FAA R,E&D Advisory Committee said in its report:

"The NSL is necessary. It is of major importance to support the design and development of a safe and efficient ATC system for the twenty-first century by:

- facilitating quicker acceptance of new ATC technologies;
- demonstrating, testing, and validating promising concepts in an integrated National Airspace System (NAS) environment;
- achieving more rapid development and deployment;
- providing improved identification of future research needs; and
- providing an enhanced mechanism for gaining early domestic and international acceptance of new procedures and equipment."

References:

[1] *Concepts and Description of the Future Air Traffic Management System for the United States*, U.S. Department of Transportation, Federal Aviation Administration, April 1991.

[2] *FAA National Simulation Laboratory*, A Report to the FAA R&D Advisory Committee, 26 June, 1990.

Further Reading:

FAA National Simulation Laboratory, Executive Overview, AOR-92-1, September 26, 1991. Presents an overview of the NSL stating why the FAA needs the NSL to address system engineering and integration issues.

National Simulation Laboratory (NSL) Requirements Documents (DRAFT), F. X. Maginnis, October 8, 1991. Preliminary draft version of the National Simulation Laboratory Requirements Document.

Configuration and Operational Capabilities of the Integration and Interaction Laboratory's Illustration of Technical Feasibility, Rebecca L. Kim, WP 91W00078, The MITRE Corporation, November 1991. Provides an overview of the first major milestone of MITRE's Integration and Interaction Laboratory (I-Lab) and identifies the functional ATC capabilities present in the software components.

Proceedings of the 1991 Winter Simulation Conference, B.L. Nelson, W.D. Kelton, G.M. Clark, pages 741-747, December 1991, "The National Simulation Laboratory: The Unifying Tool for Air Traffic Control System Development", Randall J. Stevens and Glen F. Roberts. This paper reports on efforts within the Federal Aviation Administration to develop a National Simulation Laboratory for the study of advanced concepts in Air Traffic Control.

Individual Experiment Plan: Timing the Issuance of Meter-Fix-Time Flow Restrictions, January 31, 1992, Plan for conducting the first NSC experiment in the Integration and Interaction Laboratory.

National Simulation Capability (NSC) Executive Overview (DRAFT), March 31, 1992. Presents an overview of the NSC stating why the FAA needs the NSC to address system engineering and integration issues.

National Simulation Capability Program Master Plan (DRAFT), May 1992. The document contains a plan containing program schedules, task descriptions and allocations, and resource requirements for establishing, implementing, and operating the NSC.

National Simulation Capability Management Plan; NSC Program Office,

May 18, 1992. A document intended for programmatic, managerial and technical readers who desire to know more about the organization and processed the NSC will employ to involve and support users.

Glossary of abbreviations and Acronyms

A/C - Aircraft
AERA - Automated En Route ATC
AMASS - Airport Movement Area Safety System
ASTA - Airport Surface Traffic Automation
ATC - Air Traffic Control
ATCSCC - ATC System Command Center
ATCT - Air Traffic Control Tower
CAASD - Center for Advanced Aviation System Development
CASE - Computer Aided Software Engineering
CHI - Computer Human Interface
CM - Configuration Management
COMPAS - Computer Oriented Metering, Planning and Advisory System
CTAS - Center-Tracon Automation System
DFW - Dallas Fort Worth International Airport
DSP - Departure Sequencing Program
DL - Data Link
Eq or Equip - Equipment
FAA - U.S. Federal Aviation Administration
FAST - Final Approach Spacing Tool
FAATC - Federal Aviation Administration Technical Center
GS - Graphic Services
LAN - Local Area Network
MAESTRO - Means to Aid Expedition and Sequencing of Traffic with Research of Optimization
MLS - Microwave Landing System
NAS - U.S. National Airspace System
NASA - U.S. National Aeronautics and Space Administration
Nav/Lndg - Navigation/Landing
NSC - National Simulation Capability
NSL - National Simulation Laboratory
Sim - Simulation
SIMMOD - Simulation Model, a fast-time airport and airspace simulation tool
Surv - Surveillance
Sys - System
TASF - Terminal ATC Simulation Facility
TATCA - Terminal ATC Automation
TCAS - Traffic Alert and Collision Avoidance System
Tfc Mgt - Traffic Management
TFM - Traffic Flow Management
TM - Traffic Management
TMC - Traffic Management Computer
TMS - Traffic Management System
TWR - Tower
Wx - Weather

ATC SIMULATION FACILITY FOR A TOTAL AIR NAVIGATION SYSTEM*

A. Benoît, S. Swierstra, C. Garcia

Engineering Directorate
European Organisation for the Safety of Air Navigation
EUROCONTROL
72, Rue de la Loi, B-1040 Brussels

The *Zone of Convergence or ZOC* concept is a second-generation on-line tool, which provides adequate management of traffic through convergence points *en-route and/or in a Terminal Manoeuvring Area.* It is a large-scale control loop which reacts efficiently to perturbations (man-machine interfaces - ATC system/controller and pilot/aircraft - and meteorogical conditions) as they appear.
The *Simulation facility for a Total Air Navigation System, STANS,* has been designed specifically to meet the requirements associated with the total development cycle of the ZOC on-line ATM tool from the evaluation of the initial concept definition to the implementation in a real ATC system. Its use is described for several applications, furthermore special attention is given to the development and use of the *multi-aircraft flight simulator, ACCESS,* which constitutes the main element of the "air component".

1. Introduction

There is an increasing demand in European air traffic control agencies for an overall *Air Traffic Management (ATM)* function to enable optimum use to be made of available capacity. At the flight planning stage this will be performed on a European scale by the EUROCONTROL Central Flow Management Unit (CFMU) in Brussels, Belgium. Once the aircraft are airborne the management of traffic and control of each individual aircraft should be handed over to an *ON-LINE AIR TRAFFIC MANAGEMENT FUNCTION.* Figure 1 illustrates a possible solution in the form of a network of interconnected local ATM modules which support the executive air traffic controllers in order to achieve an optimum flow of traffic through all centres and

* *This paper was originally prepared for discussion at the EUROCONTROL ATC SIMULATION SEMINAR, held at the Institute for Air Navigation Services, Luxembourg, June 11-13 1991.*

sectors.

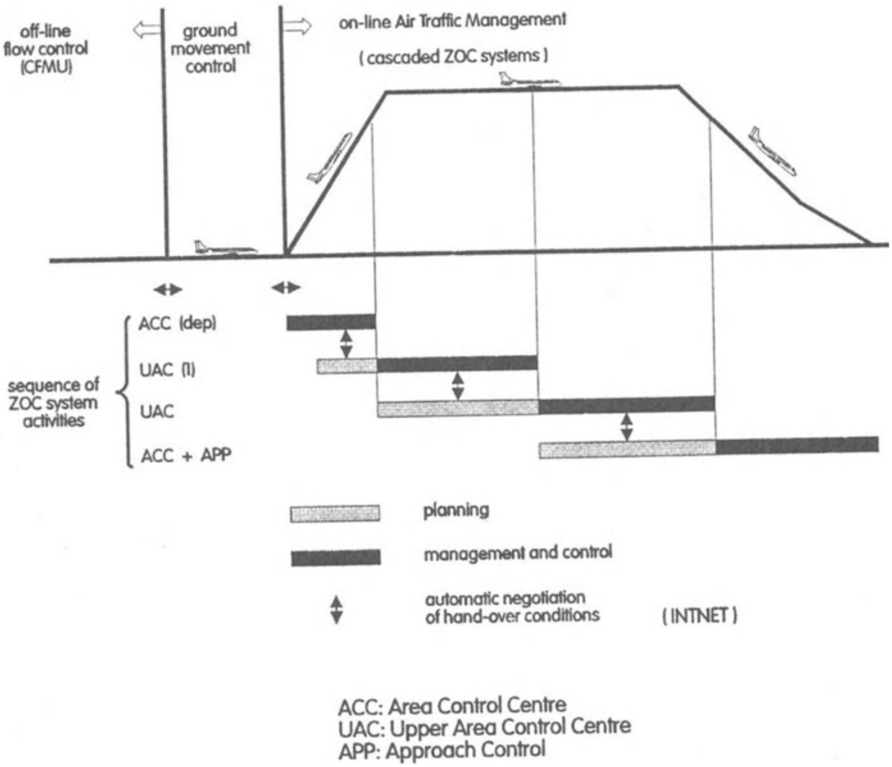


Figure 1: Air Traffic Management Modules

The *Zone of Convergence (ZOC)* (Refs. 1 and 2) concept developed by EUROCONTROL is a second-generation on-line ATM tool, which provides efficient management of traffic through convergence points *en-route and/or in a Terminal Manoeuvring Area (TMA).* Such a system may be considered as a large-scale control loop of which the regulatory quality depends chiefly on the efficiency with which the control functions react to perturbations as they appear. Three major sources of perturbation can be identified: two man-machine interfaces - ATC system/controller and pilot/aircraft - and the meteorological conditions.

Clearly, the test and development platform for an on-line ATM function must allow the independent control of each major perturbation in order not only to optimise the control algorithms but also the ergonomic aspects of the man/machine interface both in the air and on the ground.

The *STANS simulation facility (Simulation facility for a Total Air Navigation System)* has been designed specifically to meet the requirements associated with the total development cycle of the ZOC on-line ATM tool from the evaluation of the initial concept definition to the implementation in a real ATC system.

In this paper we describe how the STANS system is used during the various test scenarios of the ZOC concept and related applications. Special attention is given to the design and use of the *ACCESS multi-aircraft flight simulator* (Refs. 3 and 4), which constitutes the main element of the "air component".

2. Evolution of the Requirements

2.1 The ATC control loop

A schematic structure of the ATC control loop is shown in Figure 2. The diagram shows the breakdown of the overall ATC control loop into individual modules which interface at well defined levels.

The radar subsystem feeds the ATC data-processing system with positional data of a certain accuracy, possibly amended by some aircraft-derived data in the case of those aircraft equipped for example, with a mode-S data link facility. In the subsequent processing, radar data may be correlated with current flight plan information and the overall traffic situation is presented to the ATC controller on his radar scope. The controller will communicate clearances to the individual aircraft through the standard R/T communication channel or use a ground/air data link where available. Subsequently, these directives are applied by the pilots, poss- ibly after negotiation with the controller.

The on-line ATM system receives from the ATC data processing system the flight plan information of all relevant flights and their position tracks as observed by the radar system. Based on this data and

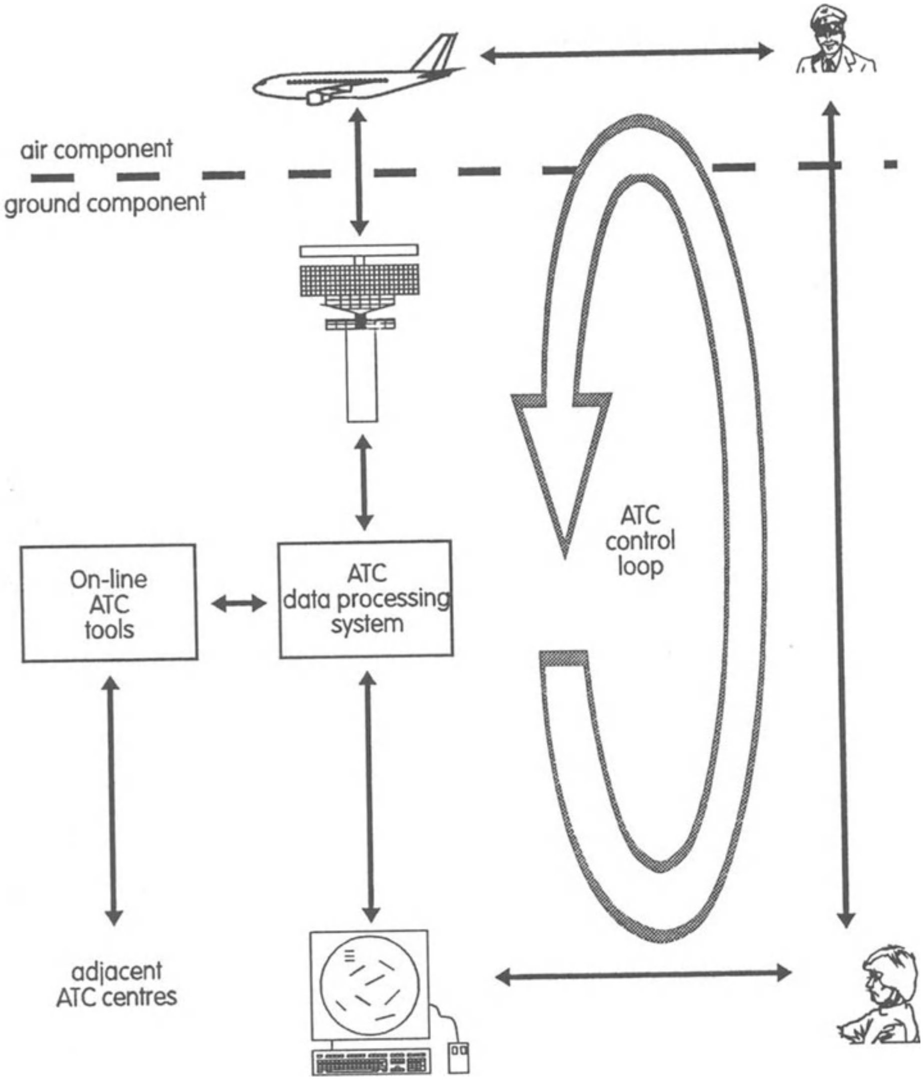


Figure 2: Air Traffic Control Loop

handover conditions negotiated with adjacent ATC centres, the on-line ATM system advises the controller how to optimise the traffic stream. The STANS simulation environment has been developed to support the specific needs required during the development cycle of such ATM systems which are discussed below.

2.2 Organization of simulations

In principle, two categories of simulation runs can be identified. The first one relates to the technical evaluation of the ATM system. Work in this category requires precise knowledge and control of the applied perturbations to assess the system performance and allow tuning. Apart from the standard functions at the controller work position, additional features are required to allow the monitoring of the ATM system operation such as predicted flight paths and system parameters.

During the initial development phases, the *air component* may consist solely of simulated traffic, but as the project matures a mix of simulated traffic and actual aircraft flight simulators manned by line crews is absolutely essential to study the impact of airborne-related perturbations in a controlled environment. Integration of flights with real aircraft is less useful at this stage of the project because of the uncertainty of the actual meteorological conditions, their evolution and their synchronisation with the meteorological conditions that affect the simulated traffic.

The second category of simulations pertains mainly to the operational evolution of the ATM system concept itself. During the early stages work will concentrate on the design of the man-machine interface aspects of the controller work position, but later the operation of the ATM system in a wider context will have to be assessed. For that purpose the impact of its operation in the ATC centre as a whole with several sectors will have to be simulated, perhaps with the realistic emulation of adjacent ATC centres with or without identical ATM systems. After this stage, the introduction of actual air traffic in the simulation scenarios may be considered, gradually leading to the implementation at an actual ATC centre.

2.3 Structure of the simulation system

The scenario for the development of the ATM system as described above clearly calls for a well structured design of the simulation facility. In the *STANS* system, the *ZOC* ATM system is a completely stand-alone module which interfaces with the ATC data-processing system through simple protocol controlled by the ATC system in a master/slave relationship. In a first stage this allowed us to test and demonstrate the basic characteristics of the ZOC ATM system on the STANS simulator in a restricted and controlled environment and, in a subsequent phase, to connect the same ATM module to the *GENESIS large-scale ATC simulation facility* at the EUROCONTROL Experimental Centre (EEC) in Brétigny-sur-Orge, France.

The same interface philosophy is applied to the other interconnections, in particular at the radar data input level. This flexibility allows the air component to be made up of any mix of simulated traffic, actual airline flight simulators manned by line crews and real air traffic as observed by ATC radars.

Nevertheless as the STANS system is in principle a simulation tool, the main part of the air component consists of the *multi-aircraft flight simulator module ACCESS* (Aircraft Control Console for Experiments, Simulations and Studies) (Ref 3).

3. Multi-Aircraft Flight Simulator ACCESS

In an ATC simulation facility the air component comprises all active pilot-aircraft combinations of which most will be simulated. In the STANS facility the simulated traffic is generated by the ACCESS multi-aircraft flight simulator, which is developed as a completely independent package.

In order to meet the vast range of requirements summarised above, the ACCESS flight simulator consists of various more or less independent modules which can be combined in several ways to obtain the required flexibility and consistency. The resulting structure is shown in Figure 3.

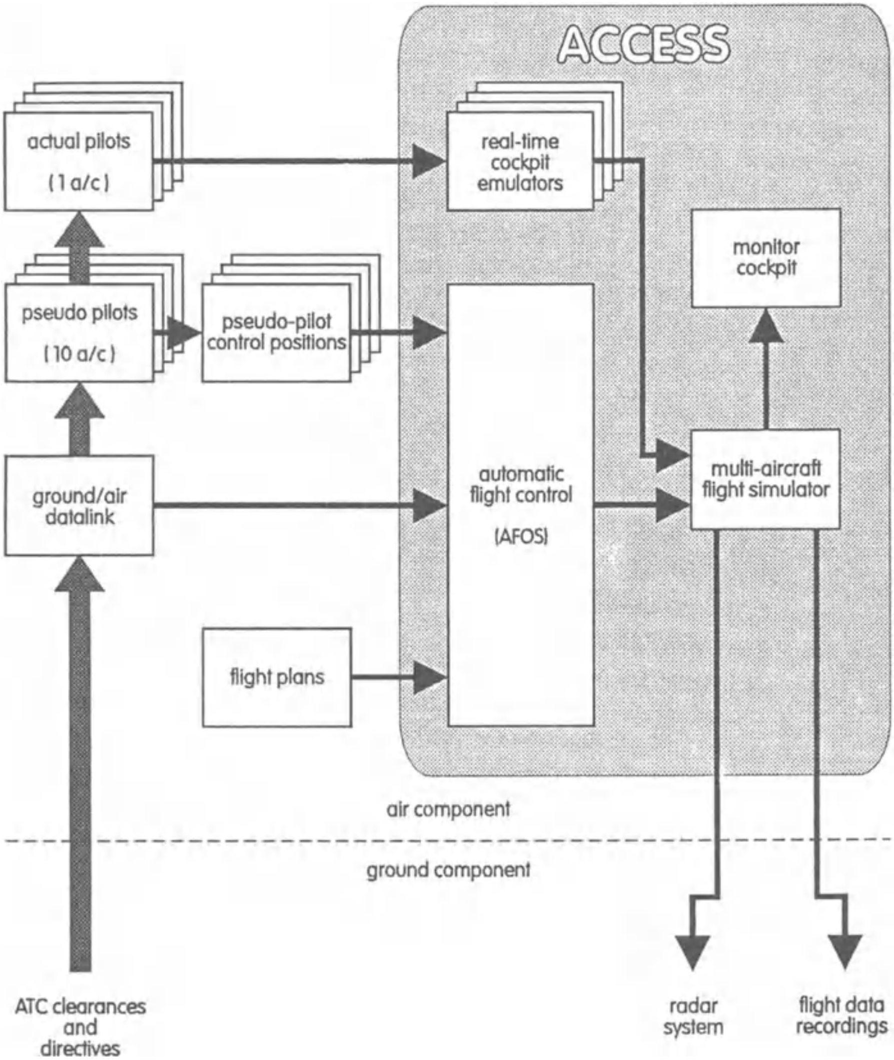


Figure 3: Structure of the STANS Air Component

3.1 Structure of the air component

The multi-aircraft flight simulator module generates data for three customer applications. The first one is the *radar system* of the STANS ground component, which receives data on the exact computed aircraft positions, with the possible addition of certain airborne data in the case of aircraft equipped with an air-ground data link facility. Later, in the radar module of the ground component, the positional data may be modulated with appropriate error functions to emulate a realistic radar performance.

Secondly, the general *flight data* can be *recorded* for cost analysis, estimation of cockpit workload, etc. Thirdly, through a "*monitor cockpit*" module (Ref. 3), it is possible to display in real time the status of the major aircraft instruments and autopilot, flight director and auto-throttle flight mode annunciators (FMA's). The monitor cockpit module provides the means to obtain a realistic picture about what is actually happening in the cockpit of each simulated aircraft.

The interface between the flight simulator and the control modules has been defined at the level of auto-pilot, flight director and auto-throttle commands. In this way the same functionality as that in the real aircraft for normal flight operation is in principle available. In practice it allows the specification of flight scenarios compatible with the actual operation of the aircraft under normal operating conditions, e.g. the use of non-clean aircraft configurations, landing gear, speed brakes, reverse thrust, anti-icing, etc. As Figure 3 indicates, there are *two flight control modules* which address the same interface level. The first allows a *real pilot* to fly *one aircraft* of the traffic sample in an interactive way through the use of a standard computer terminal. The cockpit display is identical to one designed for the monitor cockpit and input commands are accepted through the standard keyboard. The command structure has been defined such that the time required to activate a given aircraft control function via the keyboard is compatible with the time required in the real aircraft. In this mode, the pilot can control the aircraft manually and, in this way, extremely realistic aircraft behaviour is ensured.

The obvious drawback is that this level of operation requires one pilot and one pilot position for each aircraft simulated. To reduce the manpower and machinery required in simulations which involve more than a few aircraft, a facility has been provided to use *pseudo pilot* control positions from which *up to 20 aircraft* can be controlled simultaneously. The typical function of the pseudo pilot operating such a position is to reply to the ATC directives received from the ATC controller via the R/T channel in a standard phraseology and subsequently to enter these directives into the machine using specially designed interface equipment.

Once these directives are entered they are interpreted by the *AFOS (Automatic Flight Operating System)* module of ACCESS, which translates them into a series of pilot actions depending on the type of directive, the actual aircraft state and phase of flight: the AFOS module emulates the strategy followed by an actual pilot in implementing a given directive, but always in a prescribed, predictable way. The number of aircraft that can be handled by a given pseudo pilot depends mainly on the average number of directives which are received i.e. where one pseudo pilot could accommodate 20 a/c flying en route in cruise conditions, this would be reduced to about 3 to 4 aircraft if they are operating in a Terminal Manoeuvring Area (TMA) on radar vectors. However, there is no limitation on the number of pseudo pilot positions which can be operated simultaneously.

With a realistically-operating AFOS module available, only a small step is required to connect an *automatic ground/air data link channel* e.g. one compatible with the mode-S radar system. The controller work position can be equipped with a suitable data link ground terminal input device giving the ATC controller the possibility to directly enter the ATC instructions him/herself. In an ATC simulation exercise, this leads to a considerable reduction of pseudo pilot positions and the associated manpower required.

Going one step further, when the ATM system is a sufficiently advanced one, such as the ZOC system, which is capable of generating reliable ATC advisories for aircraft in all phases of flight, the ATC controller may turn these advisories into directives in an automatic or semi-automatic way, possibly after editing some of the clearances to ensure compatibility with his own strategy. This mode of operation has been shown to be extremely useful as it allows very complex simulations to be performed in an *"automatic" test mode* without or at least with very little human intervention - even in "fast-time" if so required. Moreover, as the pseudo pilot position and data link modules address the same AFOS interface level, it is possible to transmit additional ATC directives to selected aircraft via a pseudo pilot position in addition to those received automatically via the data link channel. In this way the traffic evolution can be disturbed to allow the study of the behaviour of the ATM system under test in a perturbed environment, all with minimal human resources.

3.2 Automatic Flight Operating System (AFOS)

The main function of the AFOS module is to compare current aircraft position and status with the targets described in the flight plan and to perform the necessary manipulation of auto-pilot, flight director, auto-throttle and navigation systems in an identical way to that of an actual pilot when operating the real aircraft. This involves close monitoring of the flight's progress and the necessary actions must be carried out in due time to ensure on-board flight safety (e.g. changes in the configuration of the airframe, the use of speed brakes or engine power during descent, etc). In addition, AFOS interprets and implements the directives received from the ATC controller via the pseudo pilot position or the ground/air data link channel.

A selection of the parameters for consideration is shown in Figure 4. It illustrates the parameters that AFOS considers for the aircraft operation on the basis of a given flight plan and ATC clearances. In the diagram the emphasis is placed on control of the *vertical flight profile.* The horizontal navigation aspects are summarised in the *"a/c heading"* block, which in ACCESS not only includes the *"heading select"* function of the autopilot but also the route following capabilities of *Flight Management Systems (FMS).*

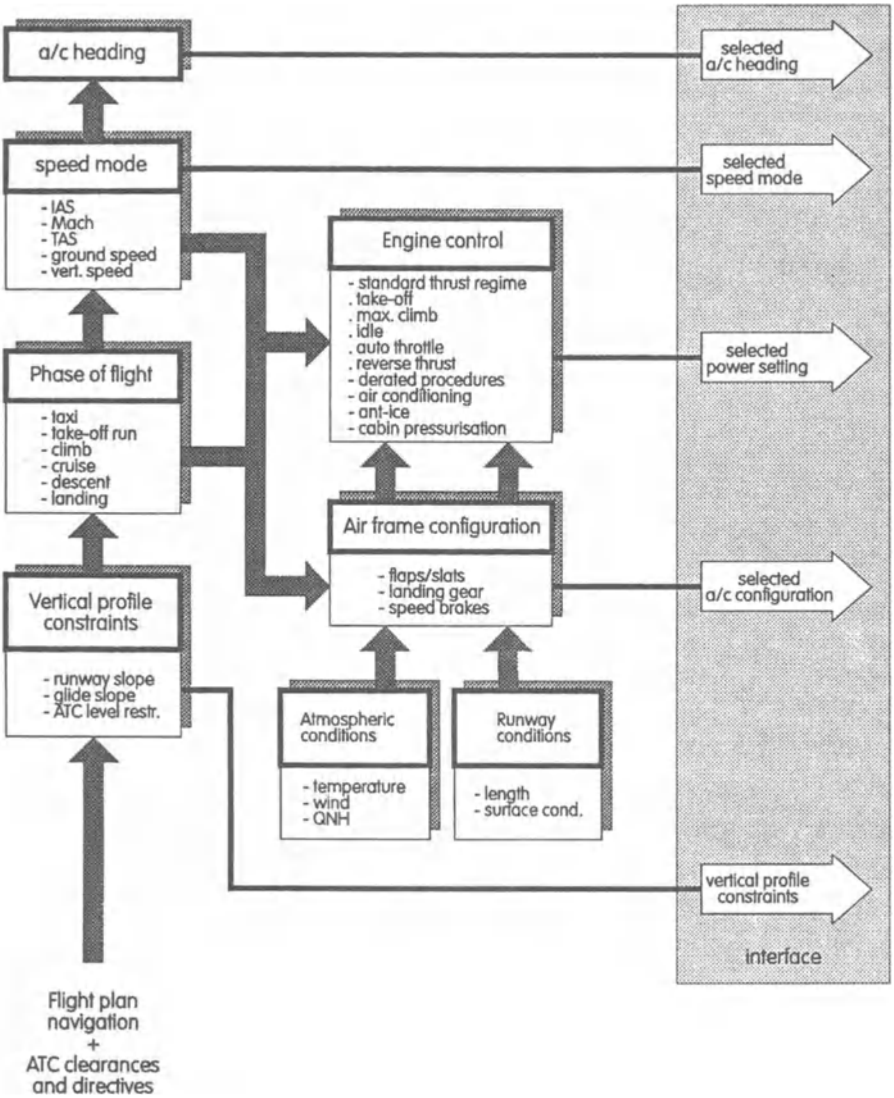


Figure 4: Flight Control (AFOS)

3.3 Flight profile calculations

The ACCESS flight simulation facility offers a choice of *two flight profile calculation modules* which have identical interfaces with respect to both AFOS on the flight control side and the other STANS modules such as the radar system on the output side.

For *normal ATC simulation purposes* the prime requirement is that the sequence of 4-D positions (t,x,y,z) as provided by the radar system is realistic both in terms of vertical and horizontal progress of the flight. They should accurately reflect the physical and operational limitations of the aircraft concerned. In addition, a realistic estimate of the fuel consumed during the simulated flight must be available for flight cost evaluation and all this should be achieved with the lowest computational overhead so as to allow the maximum number of aircraft to be simulated on a given hardware configuration.

For the *evaluation of different ATM strategies* the requirements can sometimes be more severe. These may include the detailed study of the impact of ATC procedures on aircraft operation and performance and cost assessments for the various phases of flight. Typically such simulations will only involve a limited number of aircraft in a well-defined geographical area.

The flight profile calculation technique used for this type of simulation requires a detailed aircraft model as shown in Figure 5. In this approach the engines and airframe are modelled accurately using large *data-tables* as published in the *"Performance Engineers Manual"* for the aircraft concerned. These data-tables represent the only accurate model data validated and consistent for all phases of flight except emergency conditions. The application of the model is straight forward, as the aircraft status and position have only to be delivered to the radar system at sampling rates equivalent to the radar antenna revolution (>5 seconds). This allows a considerable simplification of the equations of motion leading to a point mass model based on two scalar kinetic equations, two scalar dynamic equations and one scalar equation given the fuel mass flow in terms of net thrust (Ref. 4). The actual time step used in the internal calculations depends typically on the phase of flight and in certain cases may be considerably smaller than the radar update rate so as to ensure the stability of the flight control loops. The subsequent integration of the aircraft state results in the desired flight profile.

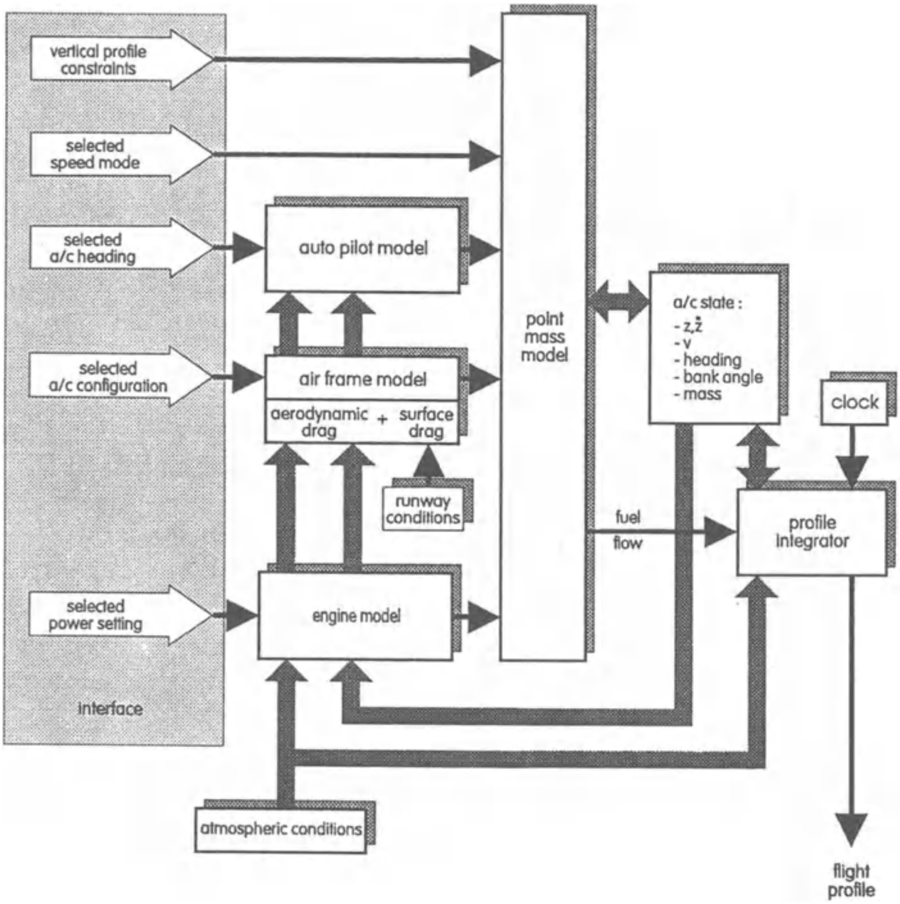


Figure 5: Flight Simulator: Detailed A/C Model

Unfortunately, aircraft performance data for this type of modelling are not always available with the required degree of detail and accuracy (Ref. 5). This is particularly the case for older aircraft but also for the latest ones in cases where manufacturers are reluctant to release the data.

Tools have been developed to provide a partial solution to this problem in the case of aircraft for which at least some reliable data are available. The missing elements can be added through interpolation or scaling from other aircraft for which a complete set of data is available and duly validated for all phases of flight so as to assess the final accuracy obtained.

In *normal ATC simulations* the degree of detail obtained from the level of aircraft performance models above is not required. For these applications it is sufficient if the accuracy of the flight profiles generated is within the normal spread for the aircraft fleet. However, the actual dynamic behaviour of the aircraft must be correctly modelled with the associated fuel flow. As reported in previous work (Ref. 6 and 7), the vertical profile of aircraft flying defined speed regimes can be accurately computed through parametric models. The *EROCOA method* (Equivalent Rate of Change of Altitude) is a technique which accurately describes the climb performance of aircraft by using a set of 8 coefficients. Identical performance for the en-route descent can be obtained from the *PARZOC* approach (PARabolic performance model for application in a Zone Of Convergence). Although initially designed for use in flight profile prediction modules to be applied in on-line ATC systems, these approaches proved also to be very suitable for use as aircraft models in the ACCESS flight simulator.

As illustrated in Figure 6, the technique is simple. The performance term (T-D)/mg which corresponds to the climb gradient, h/v, for a non-accelerated flight can be derived in a straightforward manner from the set of EROCOA or PARZOC coefficients. Through a simplified set of equations of motion it is not only the slope of the flight profile that can be readily computed but also the tangential ac- celeration. Supplementary coefficients to adapt the performance for non-clean airframe configurations, ground movements, etc. complement the aircraft model. The advantages are obvious. The aircraft model based on EROCOA and PARZOC coefficients can be easilyestablished from flight profile data, as observed or published by the manufacturers, which are much more readily available than the detailed performance data contained in the performance engineers' manuals.

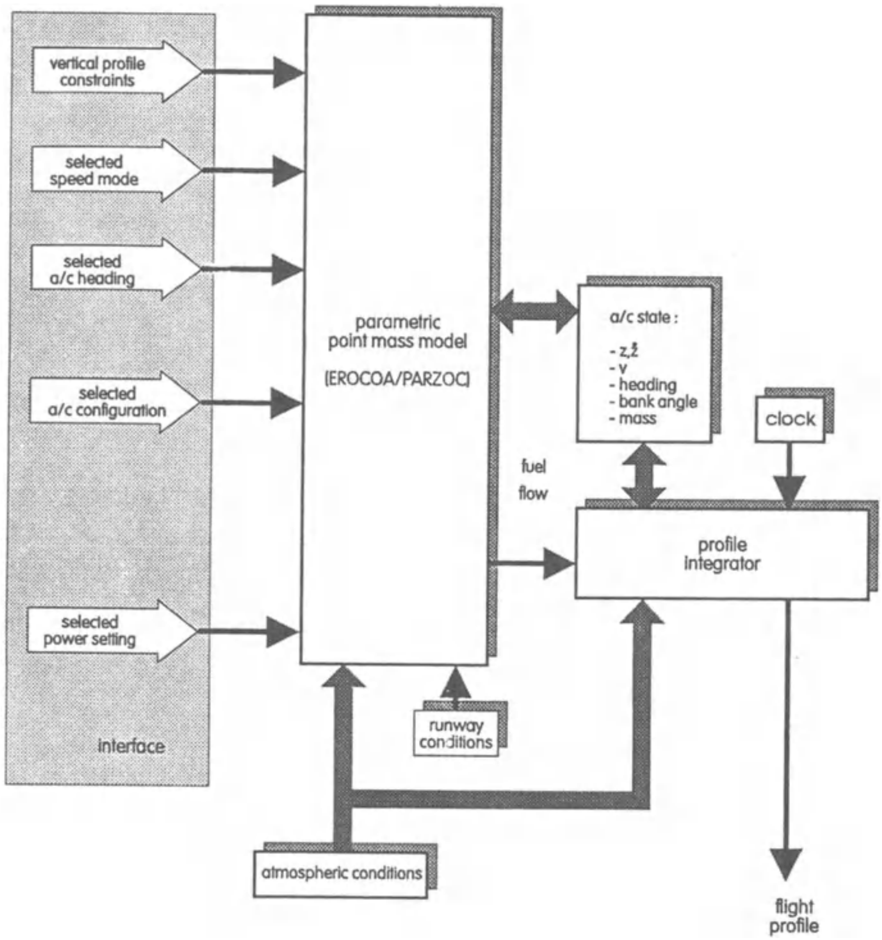


Figure 6: Flight Simulator: Parametric A/C Model

The validation and tuning of the model is straightforward and the computational overhead in respect of its application is considerably lower. Moreover, using several airline flight simulators, field tests of a ground-based 4-D flight path prediction and control module based on this aircraft performance model have confirmed the accuracy level that can be obtained and its applicability to real-life situations (Ref. 8).

In addition during the development phases of the ATM system, it is advantageous to use an identical aircraft model in both the air component which generates the flight profiles and the ground based

ATM trajectory prediction module as this considerably reduces the perturbations which result from the inconsistencies between the two components in the real world. In a later stage the required realism can be introduced by appropriate modulation functions.

3.4 Airborne equipment

It is clear that the degree of sophistication of the on-board equipment has a considerable impact on the navigation of the aircraft. Therefore in the flight plans used by ACCESS the level of airborne instrumentation can be specified. Accordingly the flight path of a small light aircraft in which the pilot navigates on the basis of standard radio aids will show a realistic, random navigation around the route centre line associated with overshooting and short-cutting at turn points. On the other hand, modern aircraft with area-nav facility will stick much more closely to the route centre line.

It can also be specified whether a given aircraft is equipped with an air/ground data link facility e.g. mode-S. If so, as well as the aircraft position and mode-C altitude reports, certain other flight data such as heading, air speed, etc. can also be communicated to the ground-based ATC system. Aircraft heading data in particular will not only improve radar tracker performance but also considerably reduce the reaction time of the advisories produced by the on-line ATM system, particularly in a TMA environment.

This possible mix of airborne instrumentation and equipment levels will allow the design and testing of ATC strategies to be applied in coming years when such a mix will be a reality. Moreover, the associated difference in track-keeping accuracy among the various aircraft simulated results in a very realistic radar picture.

4. Applications

The STANS simulation facility has been developed and used to support the work on Air Traffic Management research in the Engineering Directorate of EUROCONTROL. The typical layout for this work is illustrated in Figure 7. It shows the STANS system "on location" when integrating actual airline flight simulators into the air component. The supplementary traffic is generated by ACCESS controlled through one or more pseudo pilot positions and individual cockpit emulators depending on the manpower available.

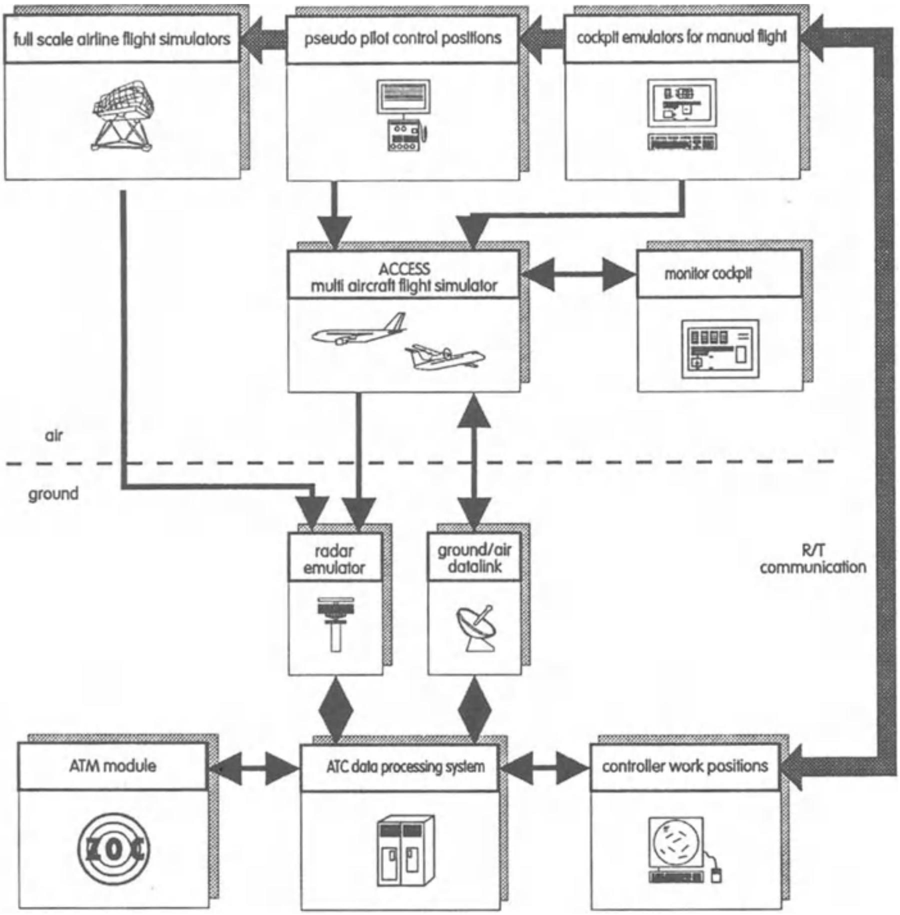


Figure 7: STANS Application Use of Airline Flight Simulators

A similar set-up was used in the large-scale real-time simulation of the ZOC concept at the Eurocontrol Experimental Centre (EEC) at Brétigny, France in 1989. In this exercise the ATC system and controller work positions of STANS were replaced by the EEC standard GENESIS facilities, which provide a more common and familiar user interface to the controllers than the off-the-shelf computer equipment as used in STANS. The air component consisted of eight pseudo pilot positions controlling up to 80 ACCESS aircraft and the Multi Cockpit Simulator facility (MCS) of the EEC flown by an actual pilot.

Another application involves a demonstration of the use of the mode-S data link facility involving the BAC 1-11 research aircraft of the Royal Aerospace Establish- ment (RAE) in Bedford, U.K. and the mode-S radar ground station of the Royal Signals and Radar Establishment (RSRE) at Malvern, UK (see Figure 8). The BAC 1-11 aircraft is flying in a simulated ACC sector over South East Anglia, UK. It is equipped with a transponder capable of supporting the mode-S data link protocol and special hardware which interfaces to the pilot. Among other applications, the demonstration flights have shown the possible use of the data link to communicate ATC directives on the up-link and pilot acknowledgements on the down link. The other traffic in the sector is generated by the ACCESS simulator.

These applications give an indication as to how the modular design of the STANS simulation facility can be adapted to the requirements of specific experiments with only minor modifications.

5. Implementation

The STANS hardware components consist of standard, off-the-shelf computer equipment. The controller work position comprises the 19" colour screen and keyboard of a Data General DS 7540 graphics workstation and the real time cockpit emulators and pseudo pilot control positions are implemented on IBM compatible personal computers.

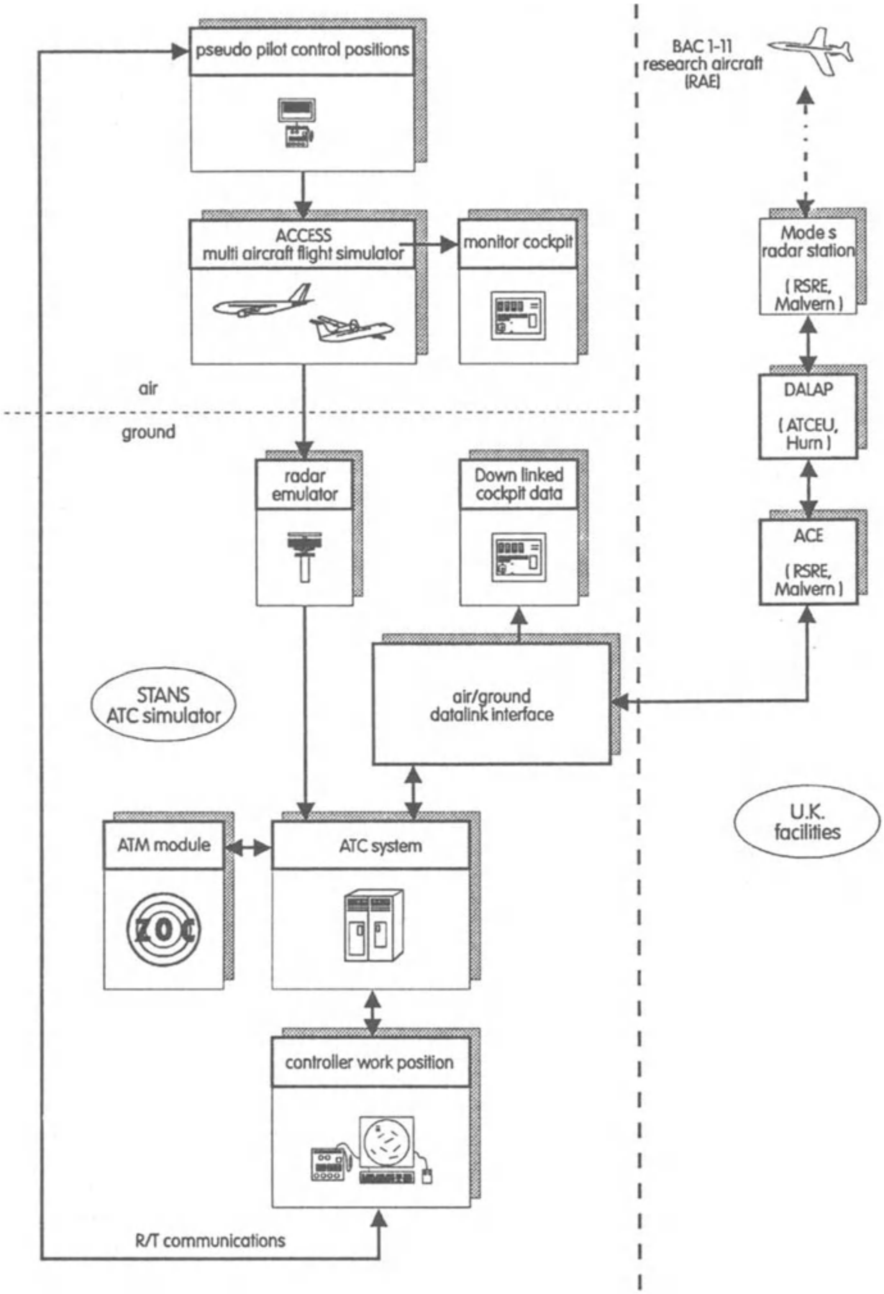


Figure 8: STANS Application. Mode S Data Link Trials

The software of the STANS system is modular and designed along the structure discussed above. The modules are implemented as independent processes which communicate via shared memory and interprocess communication channels (IPC). Most software is written in FORTRAN 77 and based on the AOS/VS operating system of Data General.

In a minimum configuration the total STANS system, i.e. ground and air components, can be implemented on a single DS 7540 workstation (0.9 mips - 4 Mbyte of main memory). In this setup a traffic sample of about 20 aircraft can be run in real time. For larger samples, the individual system modules may be distributed without modification over several mini computers using appropriate network software.

A version of ACCESS has been translated into the "C" programming language and drives the SMART radar data plot simulator, which is a joint development of the firm ORTHOGON and the Dutch Aerospace laboratories, NLR, under contract to the Federal Aviation Administration (FAA) and EUROCONTROL.

6. Conclusions

The *STANS simulation facility* was initially developed to support the *ZOC ATM tool* from the conception phase until on-line application in an actual ATC system. It provides the means to test any ATM module operation in *en-route* as well as *terminal area* environments, using traffic samples which may consist of *any mix of aircraft* and instrumentation level from piston-propelled, VFR equipped, to the most modern jets with sophisticated 4D FMCS systems on board supporting an automatic air/ground digital data link facility. This freedom of choice in the composition of traffic samples together with the configurable characteristics of the air-ground-air communication channels make the STANS facility an *ideal tool for the evaluation of future ATC scenarios* at an early stage of conception.

The software is implemented on *easily transportable hardware* to allow on-site experiments, e.g. the integration of airline flight simulators manned by line crews in order to simulate the entire ATC control loop with the maximum degree of realism in a controlled environment.

The *flexibility* which allows the STANS simulation facility to be adapted to other applications has been demonstrated successfully by the integration of the BAC 1-11 research aircraft of RAE, Bedford in an experiment to show the potential of the mode-S data link in an on-line, real-life environment.

The STANS system is very *modest in required computing resources* and even a minimum configuration constitutes a very cost effective yet powerful research tool. Its modular structure allows an easy adaptation to the requirements associated with large scale ATC simulations as demonstrated during a real-time evaluation of the ZOC concept in the EUROCONTROL Experimental Centre in Brétigny, France.

References

[1] A. Benoît, S. Swierstra, "*Air traffic management and aircraft guidance in and around a main terminal.*", AGARDograph AG-301, Vol II, 1989.

[2] A. Benoît, S. Swierstra, "*The control of inbound flights. - Basic principles -.*", AGARDograph AG-301, Vol II, 1989.

[3] A. Benoît, S. Swierstra, "*A simulation facility for assessing the next generation of 4-D Air Traffic Control procedures.*", Proceedings ICAS 1986.

[4] A. Benoît, S. Swierstra, "*Generation of aircraft trajectories for on-line operation. Methods - Techniques - Tools.*", AGARDograph AG-301, Vol II, 1989.

[5] J-L. Renteux, "*Aircraft performance models for present and future ATC needs.*", EUROCONTROL Doc. 872002, January 1987.

[6] A. Benoît, P. Brown, "*A presentation of EROCOA climb and descent performance data with illustrations of the approximation quality.*",

EUROCONTROL Doc. 732040, December 1973.

[7] A. Benoît, S. Swierstra, "*Simulation of air traffic operation in a Zone of Convergence. Aircraft performance data (PARZOC).*", EUROCONTROL Doc. 812031-2, April 1982.

[8] A. Benoît, S. Swierstra, "Ground-based 4-D guidance of flights in strong winds.", *Journal of Navigation*, Vol. 43, No 2, May 1990.