Copyright © 1996 IFAC
13th Triennial World Congress, San Francisco, USA

HYBRID CONTROL IN AIR TRAFFIC MANAGEMENT SYSTEMS [1]

C. Tomlin* G. Pappas* J. Lygeros* D. Godbole* S. Sastry* G. Meyer**

* *Department of Electrical Engineering, UC Berkeley, CA 94720.*
{clairet, gpappas, lygeros, godbole, sastry}@eecs.berkeley.edu
** *MS 210-3, NASA Ames Research Center, Moffett Field, CA 94035.*
meyer@cartan.arc.nasa.gov

Abstract. The congestion of aircraft in the airspace around large urban airports strongly suggests the need to increase the efficiency of the current air traffic management system (ATMS). In a new research collaboration which includes UC Berkeley, NASA Ames Research Center, and Honeywell Systems Research Center, an architecture for a new ATMS is proposed. Research is focused on hierarchical, hybrid control systems and their application to air traffic management. The proposed architecture describes an automated, decentralized ATMS: much of the control functionality exists in the flight vehicle management system on board each aircraft rather than at the ground-based air traffic control. Aircraft trajectories are restricted in the airport region; outside this region the architecture allows for so-called "free flight". A simulation and visualization package for hybrid systems in the framework of ATMS is being developed to aid the design process.

Keywords. Air traffic control, Decentralized control, Hierarchical systems, Hybrid systems, Flight control.

1. INTRODUCTION

The main objective of this research is to improve the safety and efficiency of air travel. Breaks in communication between air traffic controllers and pilots are increasing in occurrence; flight delays occur daily. The solutions to these problems have been to update the aging computer systems in ground-based air traffic control (ATC) centers, and to build larger and more modern airports. These solutions are only temporary. A new approach to air traffic management is needed to face the increasing demand for commercial air travel. The research presented here aims at improving air travel conditions by introducing automation to air traffic management.

A concept of considerable current interest in this research is that of *free flight*, or the ability of an aircraft to plan and fly its own optimum trajectory. With the sophisticated global positioning systems, fast real time computers, and graphics displays that are currently being developed and incorporated into the autopilots of new aircraft, these autopilots will soon be able to plan paths which optimize time of flight and fuel consumption and which satisfy the constraints imposed on the aircraft by its environment.

The usefulness of such advanced technology in the aircraft is limited by the current air traffic control practices, many of which are dictated by the need to keep the task of the human controllers simple. For example, aircraft are currently routed along prespecified paths to avoid having to deal with the complications of free flight.

[1] Research supported in part by NASA under grant NAG 2-243, and by the PATH program under grant MOU 135.

In addition, because of heavy workload, air traffic controllers are primarily concerned with maintaining safe spacing between aircraft while guiding the aircraft to their destinations; optimizing travel times and fuel consumption are not prime considerations. The introduction of automation can lead to great savings in terms of travel times, unplanned delays, and fuel consumption, and can possibly increase the number of aircraft handled. An additional benefit will be an increase in the safety of the flights (reduced number of aborted landings, near collisions, etc.). The improvement is likely to be more dramatic in the case of degraded conditions of operation such as aircraft malfunctions, ATC malfunctions (e.g. power failure), shifting winds (that cause changes in approach patterns), bad weather, and switching from manual to instrumented landings. Conditions like these occur regularly in practice and can cause severe degradation in the system performance.

The air traffic management system (ATMS) proposed here will be fully automated, but parts of this work can also be used to produce advisories for ATC and pilots in a semi-automated ATMS. The proposed new architecture for ATMS is inspired by research on the control of *hierarchical hybrid systems*. Because air traffic management requires coordination and control of a large number of semi-autonomous agents (aircraft), the number of control decisions that have to be made and the complexity of the resulting decision process dictate a hierarchical, decentralized solution. Complexity management is achieved in a hierarchy by moving from detailed, decentralized models at the lower levels to abstract, centralized models in the higher. In our architecture, the abstract higher levels will be modeled by discrete event systems and the lower levels by detailed continuous aircraft models and arithmetic control laws. This marriage of discrete event and continuous dynamics results in an overall system which is hybrid. One of the keys in this approach is the use of inter-agent coordination to increase aircraft autonomy and consequently reduce the ATC workload.

A brief overview of current ATC practice is presented in Section 2. In Section 3 the proposed air traffic management system is described. Control issues raised by the proposed design, and a description of the software simulation and visualization package *SmartPlanes* is described in Section 4, and concluding remarks are presented in Section 5.

2. CURRENT ATC PRACTICE

Air traffic control (ATC) in the United States is currently organized by geographical region. The country is divided into 20 Centers, each with its own ATC unit. In addition, around each large urban airport is a TRACON (Terminal Radar Approach Control) region with its own ATC unit. Unplanned delays in average flight times are common, thus the controllers are forced to manually reschedule aircraft landings. There is minimal communication between the different ATC units, and so forecasting delays almost impossible. Figure 1 depicts the horizontal profile of a typical approach pattern inside the TRACON. The aircraft are sequenced manually as they enter the TRACON, and they maintain this sequence along the illustrated route. Where the routes converge, ATC decides which aircraft is allowed to go first and what the ensuing sequence will be. If an aircraft enters the TRACON in an emergency state and must land as quickly as possible, ATC manually reroutes and reschedules the other aircraft so that priority can be given to the troubled aircraft. In the regions outside the airport TRACON, the routing and scheduling tasks are considerably less intensive and the workload is lighter than for TRACON controllers. If ATC predicts a loss of separation between two aircraft, it either slows down one of the aircraft or puts it into a delay loop.

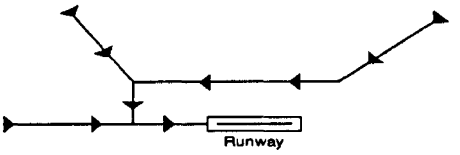


Fig. 1. Typical route pattern for arriving aircraft

In order to improve the efficiency of the current ATC system, NASA Ames is developing an advisory system for ATC. The system is called the Center-TRACON Automation System (CTAS), and is described in (Erzberger *et al.*, 1993) and (Slattery, 1995). CTAS is a program which generates *advisories*, or suggested trajectories, runway assignments, landing sequences, and schedules, which the controller may use in managing air traffic. Field tests of CTAS are now underway at the Denver and Dallas/Fort Worth airports (Green *et al.*, 1995).

3. PROPOSED AIR TRAFFIC MANAGEMENT SYSTEM

3.1 *A Decentralized Decision Making System*

An ATMS which distributes decision making about trajectory planning and conflict resolution to the aircraft has clear advantages over today's system, which is centrally controlled by ATC.

- A distributed system is more fault tolerant. If a single aircraft's computer system fails, most of the

ATMS system is still intact, and the aircraft may be guided by voice to the nearest airport.

- Congestion is increasing monthly in large urban airports. A distributed system is more suited to handling increasing numbers of aircraft than is a centralized system, since a new aircraft and its own computer system may be added easily to the system.
- Free flight, one of the features of a distributed control system, minimizes fuel consumption since each aircraft may optimally plan its own trajectory, which in turn results in reduced delays outside TRACONs. Free flight is also an advantage in avoiding turbulence, since the aircraft would not have to wait for clearance from ATC to avoid rough weather patches.
- A distributed system reduces the ATC workload, allowing ATC to spend more time in resolving safety critical situations.
- In the proposed ATMS model, each aircraft's autopilot contains a detailed model of its own dynamics, and thus the aircraft itself is well equipped to plan its own trajectory in a conflict resolution maneuver.

3.2 *Conflict and Coordination*

In a multi-agent setting there is always the possibility of conflict between the objectives of the agents. In an ATMS, the most dangerous of these is that of conflicting trajectories, a situation that is very likely, especially if one allows aircraft to independently plan and track their trajectories. An automated ATMS should be able to predict such conflicts well in advance and resolve them by altering the plans of one or more of the agents.

Even though quantitatively all path conflicts are the same (two or more aircraft occupy the same space at the same time or, more precisely, get closer than a certain threshold to one another), certain qualitative scenarios can be distinguished. Here we introduce three such examples which we call *Merge*, *Overtake* and *Collision*[2]. The Merge conflict is the situation depicted in Figure 2. Aircraft 1 (solid line) and 2 (dashed line) are coming from different directions or altitudes (or both) but want to follow the same path from point C onwards. Point C is referred to as the *conflict point*. In Overtake (Figure 3) the aircraft are already moving along the same path, with 2 trailing 1 and moving faster. At the conflict point, C, 2 catches up with 1. Finally in the Collision conflict (Figure 4) the aircraft trajectories cross transversally or meet head on.

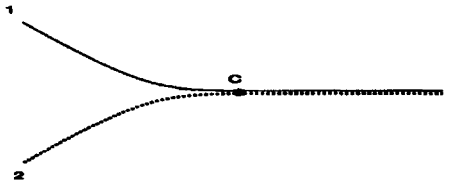


Fig. 2. Merge conflict

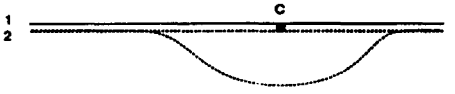


Fig. 3. Overtake conflict

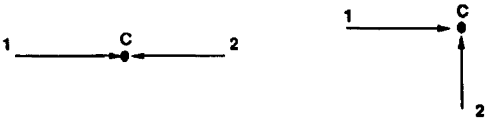


Fig. 4. Collision conflict

Conflict resolution for these examples can be carried out in a number of ways. For Merge, aircraft 1 can decelerate while aircraft 2 accelerates (or vice versa) to create acceptable time separation at the conflict point C. Overtake can be resolved by a transient change in altitude or/and direction of aircraft 2 (dashed line in Figure 3). Finally the Collision conflict can be resolved by any of the techniques used for Merge and Overtake. Which of the possible actions is taken should be dictated by considering factors such as the passenger comfort, fuel efficiency, and resulting delay.

In the ATMS architecture proposed here, the responsibility of trajectory planning and conflict resolution is distributed to the individual agents. Distributed decision making, however, implies the need for inter-agent communication and coordination as the agents that were centrally controlled become semi-autonomous and are forced to cooperate to achieve a common goal. Coordination is needed to guarantee that the agents do not take contradictory actions. To resolve a head on Collision, for example, both aircraft may decide to lose altitude, resulting in a new conflict. Similarly for the Merge conflict both planes may decide to decelerate. Finally, for Overtake conflict, coordination may be needed to guarantee that the overtaken plane does not accelerate while the maneuver is in progress and that it is not going slowly for a special reason (e.g. because it is involved in a Merge with a third aircraft). Coordination is also needed for efficiency and comfort: it may be better for an aircraft that is banking to the right to turn right to resolve a

[2] The names reflect similar situations encountered in highway traffic

collision conflict, rather than turning left or losing altitude.

In the cases discussed here the necessary coordination can be established by simple communication protocols. More complicated protocols may be needed for other maneuvers (e.g. conflicts involving more than two aircraft). The protocols must be designed so that local coordination between a few aircraft has minimal adverse effects on other aircraft in the surrounding region. Once protocols have been designed they need to be formalized and verified. More importantly, the hybrid interaction between the protocols and the continuous dynamics has to be investigated.

3.3 *Proposed Architecture for ATMS*

The functionality of the proposed ATMS architecture is slightly different for the region inside the TRACON (take off and landing) than it is for the region outside the TRACON (en-route airspace). In the region outside the TRACON, the density of aircraft is relatively small, and each aircraft is allowed to completely plan its own trajectory (to be called the free flight region). In this region, conflicts will be resolved by coordination between aircraft, the role of the ground based *Air Traffic Control* (ATC) is simply to provide information about the route, such as upcoming weather conditions. Inside the TRACON, the ATC will determine the approach trajectory for each aircraft. These trajectories will depend on local weather conditions and traffic. A block diagram of the proposed architecture for ATMS inside the TRACON region is presented in Figure 5. The levels of architecture below ATC reside on the individual aircraft and comprise what is known as the aircraft's *Flight Vehicle Management System*, or FVMS. The FVMS consists of four layers, the strategic, tactical, and trajectory planners, and the regulation layer.

3.3.1. *Air Traffic Control*

When an aircraft enters the TRACON region of an airport, ATC passes a sequence of *waypoints* to the strategic planner on board the aircraft. These waypoints are a discretization of a kinematic trajectory, accessed from a database of stored kinematic trajectories, which have been calculated offline for different combinations of:

- aircraft kinematics;
- wind magnitude and direction;
- runway configurations.

These precomputed trajectories have been optimized to provide a minimum-time path for the given aircraft kine-

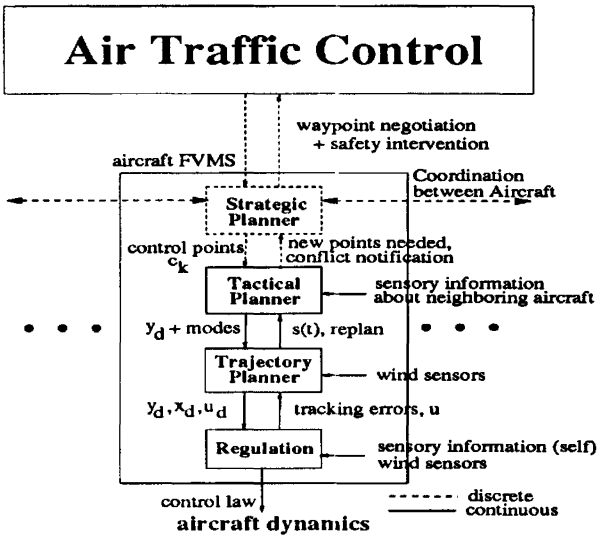


Fig. 5. Proposed ATMS Architecture: Inside TRACON

matics. The waypoints from ATC are time-stamped to provide a suggested schedule of landings, which is designed to meet the announced arrival times and reflects conflict resolution and compromises between airline schedules. Once ATC has calculated these waypoints and passed them to the strategic planner, all of the planning and control tasks are taken over by the FVMS on board the individual aircraft. If the FVMS changes the waypoints then the new set of waypoints is passed to ATC.

Outside the TRACON region, free flight is under effect and the role of ATC is minimal. ATC passes only TRACON exit and entry information to the tactical planner of the aircraft, and then the tactical planner takes over the role of calculating an initial kinematic trajectory for the aircraft.

3.3.2. *Strategic Planner*

The main role of the strategic planner is to resolve conflicts between aircraft. Inside the TRACON, the strategic planner has the additional role of designing a coarse trajectory for the aircraft in the form of a sequence of control points, c_k, which interpolate the waypoints from ATC.

If the tactical planner on board the aircraft predicts that a conflict will occur between its aircraft and other aircraft, it notifies the strategic planner. As discussed in Section 3.2, conflict resolution is achieved by communication between aircraft at the strategic level. The strategic planners of all aircraft involved in the potential conflict determine a sequence of maneuvers which will result in conflict-free trajectories, and then each strategic planner commands its own tactical planner to follow

these maneuvers. These maneuvers and resulting commands are accessed from a database of precomputed solutions to possible conflicts. Inside the TRACON, the commands are passed down to the tactical planner in the form of a modified sequence of control points. ATC is notified if this modified sequence deviates from the original set of waypoints. Outside the TRACON, where the upper levels of the ATMS architecture have looser control over trajectory planning, the commands themselves are passed.

3.3.3. *Tactical Planner*

Inside the TRACON, the tactical planner refines the strategic plan by interpolating the control points with a smooth output trajectory, denoted y_d in Figure 5. Outside the TRACON, it calculates this trajectory from scratch, using only the initial and final conditions of free flight (the TRACON exit and entry information). In addition to the output trajectory, the tactical planner determines the sequence of *flight modes* necessary to execute the kinematic plan. In both regions, the tactical planner is responsible for predicting conflicts.

The tactical planner uses a simple kinematic model of the aircraft for all trajectory calculation. For conflict prediction, it uses information about the positions and velocities of neighboring aircraft (available through radar) and kinematic models to predict their movement. If more information, such as neighboring aircraft type and capabilities, is available through communication, the models can be refined. The assumptions made in extrapolating aircraft trajectories plays a crucial role in conflict prediction. If no a-priori knowledge of the other aircrafts' intentions is known, then it may be assumed that the aircraft will maintain the same velocity over the horizon of prediction. A more conservative approach is to assume that the other aircraft will do their worst to cause conflict. Predicting the trajectories under this assumption involves solving an optimal control problem where the cost function is the spacing between the aircraft in question and its neighbors (that the neighbors seek to minimize). Clearly this approach will predict more conflicts than the constant velocity extrapolation. If it is assumed that the aircraft in question is allowed to move away from its neighbors, conflict prediction becomes a pursuit-evasion dynamical game ((Basar and Olsder, 1995), (Lygeros *et al.*, 1995)), with the neighbors pursuing (doing their best to cause conflict) and the aircraft in question evading (doing its best to avoid it).

When the tactical planner predicts that a conflict will occur, it sends a discrete signal to the strategic planner. After conflict resolution, a new tactical plan needs to be established and new conflicts predicted. Verification is needed to guarantee that this process eventually leads to an acceptable, conflict free trajectory. Because of the relative simplicity of the kinematic models it is predicted that this verification may be carried out using finite state and timed automata techniques.

3.3.4. *Trajectory Planner*

The trajectory planner uses a detailed dynamic model of the aircraft, sensory input which measures the wind's magnitude and direction, and the tactical plan consisting of an output trajectory and sequence of modes, to design a full state and input trajectory for the aircraft. This trajectory, denoted y_d, x_d, and u_d in Figure 5, is given to the regulation layer which directly controls the aircraft. The task of the trajectory planner is complicated by the presence of non-minimum phase dynamics and actuator saturation, which are studied in (Tomlin *et al.*, 1995) and (Pappas *et al.*, 1995).

4. DESIGN ISSUES

4.1 *Controllers with Switches*

ATMS involves switching at different levels. At the higher levels, the tactical planner introduces switching between different operating modes for the aircraft. For every mode segment, the trajectory planner designs a smooth trajectory. While the underlying dynamics of the aircraft are given by smooth functions, the control laws based on full state approximate linearization are different in different modes of operation. Thus the regulation level controller is switched at each instant of mode switching. In addition, the coordinate charts that are used to parameterize trajectories also encounter singularities, and one has to switch to a different chart and correspondingly a different control law. Another kind of switching is necessitated by the disturbances. Because the aircraft engine and control surfaces can produce a limited amount of thrust and moment, the space of trajectories that can be tracked satisfactorily by an aircraft is limited. The effect of small disturbances (as well as passenger comfort violations) may be nullified by reparameterization of the trajectory. In the case of severe disturbances, such as wind shear or the changes in plans of neighboring aircraft, the overall sequencing of modes has to be changed.

4.2 *Specification and Verification*

For systems with high functional and hierarchical complexity, such as ATMS, systematic design and verifica-

tion tools need to be developed. The problem of specification and verification can be reduced to showing that the discrete models used by the higher levels to describe the lower levels (including the plant) are indeed an abstraction of the behavior of those levels. The level of sophistication of current hybrid systems methodologies is severely limited to the analysis of systems with clocks (timed systems) or systems whose dynamic performance can be abstracted by clocks. In ATMS however, an accurate timed abstraction of the continuous behavior will be difficult to obtain because of the complicated nonlinear dynamics of the plant, the multi-agent characteristic of the system, and the on-line replanning of the reference trajectory.

4.3 *Simulation & Visualization Issues*

The complexity of large scale projects renders simulation a valuable tool both in the design of various control laws and coordination protocols as well as in the evaluation of overall system performance. Furthermore, a good simulation package may also be used as a debugging tool in the design process. This requires the development of a simulation package for hybrid systems that will be able to simulate both the low level differential equation models as well as the high level finite state machine models. The complexity of the system also emphasizes the need for efficient computational schemes, such as parallel and distributed computation algorithms. Each simulation run results in a tremendous amount of data that needs to be analyzed and interpreted. Visualization techniques, such as animation, can be used to present the simulation results in a manner which is much easier to analyze by the designer. In this direction, the development of *SmartPlanes*, a simulation and visualization facility for ATMS, has begun. At the current stage, SmartPlanes, shown in Figure 6, is a visualization tool which allows the user to view the trajectory of a single aircraft from various perspectives. In future versions, multiple aircraft will be shown as well as a local radar, and the user will have the ability to configure the airport so as to meet the needs of different cities.

5. CONCLUDING REMARKS

The research presented here represents a novel approach in the investigation of hybrid systems. The point of view is that the research in hybrid systems should be inductive from previously unsolvable problems which require new theory. Modeling and designing controllers for air traffic management is a challenging task demanding more than just standard control procedures: tackling the problems presented in this research will lead to fresh new methods in controlling multi-agent systems with hierarchical, hybrid structures.

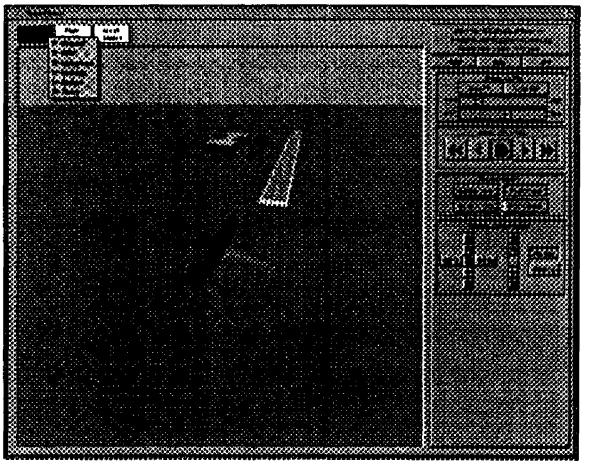


Fig. 6. SmartPlanes Visualization Program

6. REFERENCES

Basar, T. and G. J. Olsder (1995). *Dynamic Noncooperative Game Theory*. second ed.. Academic Press.

Erzberger, Heinz, Thomas J. Davis and Steven Green (1993). Design of center-tracon automation system. In: *Proceedings of the AGARD Guidance and Control Syposium on Machine Intelligence in Air Traffic Management*. pp. 11.1–11.12.

Green, Steven M., Robert A. Vivona and Beverly Sanford (1995). Descent advisor preliminary field test. preprint, to be presented at the AIAA Guidance, Navigation, and Control Conference, Baltimore.

Lygeros, John, Datta Godbole and Shankar Sastry (1995). A game theoretic approach to hybrid system design. Technical report. UCB-ERL Memo M95/77. Electronics Research Laboratory, University of California, Berkeley, CA 94720.

Pappas, George J., John Lygeros and Datta N. Godbole (1995). Stabilization and tracking of feedback linearizable systems under input constraints. In: *IEEE Control and Decision Conference.*

Slattery, Rhonda A. (1995). Terminal area trajectory synthesis for air traffic control automation. In: *American Control Conference.*

Tomlin, Claire, John Lygeros, Luca Benvenuti and Shankar Sastry (1995). Output tracking for a nonminimum phase dynamic CTOL aircraft model. In: *IEEE Control and Decision Conference.*