AIAA 2002–5011

AIAA 2002–5011

A control theoretic predictive model for sector-based air traffic flow

Alexandre M. Bayen
Stanford University, Stanford, CA 94305

Pascal Grieder
Swiss Federal Institute of Technology, CH-8092 Zürich

Claire J. Tomlin
Stanford University, Stanford, CA 94305

GNC Conference
5–8 August, 2002/Monterey, California

A control theoretic predictive model for sector-based air traffic flow *

Alexandre M. Bayen †
Stanford University, Stanford, CA 94305
Pascal Grieder ‡
Swiss Federal Institute of Technology, CH-8092 Zürich
Claire J. Tomlin §
Stanford University, Stanford, CA 94305

We derive a control theoretic model of sector-based air traffic flow using hybrid automata theory. A subset of this model is used to generate analytic predictions of air traffic jams: we define and derive a *dynamic sector capacity* which we use to predict the time it takes to overload a given portion of airspace. We design and use a validated air traffic flow simulator, to assess the accuracy of our predictions. Then, we use our simulator to show that flow scheduling and conflict resolution may be decorrelated under assumptions on aircraft density conditions.

Keywords: **scheduling, network control, distributed control, hybrid automata theory.**

Introduction

The National Airspace System (*NAS*) is a large scale, layered, nonlinear dynamic system; its control authority is currently organized hierarchically with a single *Air Traffic Control System Command Center (ATCSCC)* (in Herndon, VA) supervising the overall traffic flow. This is supported by 22 (20 in CONUS) *Air Route Traffic Control Centers (ARTCCs)* organized by geographical region. Each Center is subdivided into about 20 sectors, with at least one air traffic controller responsible for each sector. Each sector controller may talk to 25-30 aircraft at a given time (the maximum allowed number of aircraft per sector depends on the sector itself). In general, the controller has access to the aircraft's flight plan and may revise the altitude and provide temporary heading assignments, amend the route, speed, profile, in order to attempt to optimize the flow and to keep aircraft *separated* (5 nautical mile requirement in the Center airspace), as well as to provide weather reports and winds.

Existing NAS modeling tools have functionality which spans the modeling of runway and airport capacity and operations, through airspace operations and conflict resolution,[1,2] to human factors and man-machine integration. See[3,4] for a detailed surveys of NAS modeling and conflict detection and resolution methods. A recent tool, FACET,[5,6] represents the first accurate NAS simulation tool, with the additional capability of "playback" mode using actual traffic flow data. Our goal is to develop a model which complements existing tools, by providing a control theoretic component which models the influence of Air Traffic Control (*ATC*). While the additional logic required to model the actions of the controller does not pose a significant computational problem if the aircraft density in the airspace is low, it becomes an issue as density increases. The long term goal of increasing capacity as well as safety in the NAS cannot be achieved without an in depth analysis of the applied control logic. If such a system were shown to model the current airspace with sufficient accuracy, then a wide array of applications would become feasible, including providing additional support for ATC in predicting delay.

In this paper we present a model, as well as analytic and simulation results, of the aircraft and controller actions within a sector of airspace. We use our model to study the effect of aircraft flow density requirements at sector boundaries (due, for example, to miles-in-trail requirements at airports in subsequent sectors); the model allows prediction capability of how the current system might react to imposed flow conditions. In addition, the model allows the testing of the effectiveness of different controller policies in minimizing

*Research supported by a graduate fellowship of the Délégation Générale pour l'Armement (France), NASA Ames under grant NASA/NCC 2-5422, and by the DARPA Software Enabled Control program, under grant F33615-99-C-3014 (administered by AFRL).

†AIAA student member, Ph.D. Student, Hybrid Systems Laboratory, Aeronautics and Astronautics, Stanford University.

‡Dipl. Ing. ETH, Ph.D. Student, Automatic Control Laboratory, Swiss Federal Institute of Technology.

§AIAA member, Assistant Professor, Aeronautics and Astronautics, Director, Hybrid Systems Laboratory, Stanford University.

delays in posted flight plans.

The paper has four main components. In the first, we present a model for a controlled sector, based on a hybrid system model for each aircraft which encodes simple aircraft dynamics under discrete action of the controller (we have observed that the set of commands used by controllers, while large, is finite, and consists of simple actions such as: "turn to heading of x degrees", "hold current heading", "fly direct to jetway y", "increase speed to z knots"). In the second, we define the concept of sector *dynamic capacity*, which we combine with analysis to predict the time it takes to overload given sectors of airspace, and thus predict delays, assuming controllers use a subset of their available control actions. In the third, we present our simulator: we describe the implementation of our model in C++ with a MATLAB user interface, and, given the discrete and continuous state of the aircraft, the action assigned to the aircraft (computed via a cost function), which takes into account the state of surrounding aircraft and the aircraft flight plan. The structure of the cost function has been designed by us through direct interaction with air traffic controllers. We validate our simulation using Enhanced Traffic Management System (ETMS) data.[6] Finally, we assess how accurate our analytical predictions are with respect to the true system by comparing them against our validated simulator results, and draw conclusions about the effectiveness of different controller policies in flow control problems. We extend our simulations to show how conflict resolution and traffic flow may be *decorrelated* – an important observation that we use in our current work.

The data that we present in this paper pertain to several sectors within the Oakland ARTCC, the geographical data presented is available in the public domain (we used JEPPESEN[7] high altitude enroute charts). Our controller model and cost function has been designed based on our observations, over several hours, of sector controllers for given sectors; we have made justified approximations where appropriate for the study in this paper. Most of the scenarios run in this paper do not represent normal traffic flow: because we are interested in modeling delay propagation of the system under stress, the traffic scenarios modeled represent heavy traffic flow along jetways.

Air traffic flow modeling

Infrastructure, airspace model

The structure of the NAS is complex, for it involves a multitude of interacting agents and technologies: aircraft monitoring, flow management, communication, human-in-the-loop. For the present work, in which we are interested in predicting delay, we extract and model only the features which are important for this purpose. We model a portion of the Oakland ARTCC, which contains five *sectors*. These sectors enclose the Oakland TRACON (Terminal Radar Approach CONtrol), which distributes incoming aircraft to the San Francisco, San Jose and Oakland airports. The TRACON is the final destination of the flows we will be interested in.

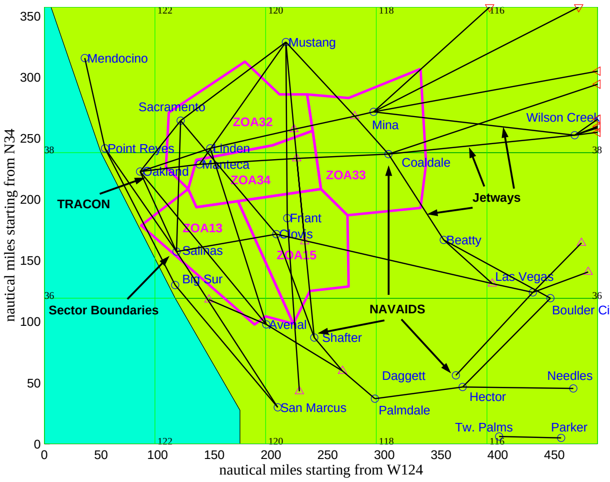

Fig. 1 ATC sectors modeled for this study: 32, 33, 34, 13 and 15 within the Oakland ARTCC. The data modeled comes from FACET[5] as well as JEPPESEN high altitude enroute charts.[7]

We model a sector by a portion of airspace containing aircraft (Figure 1). The interior of this domain is controlled area (in which the local controller can actuate the flow). Within each sector, navigation infrastructure (*jetways, navigation aids, waypoints*) is crucial to help the flow follow certain desired patterns; we therefore model and use it, even if sometimes more than 40% of the aircraft fly off jetway. Our model is multilayer (aircraft are allowed to fly at different altitudes, but do not climb or descend). Altitude changes are not crucial for the effects we want to identify: the type of sector overloads we are interested in mainly results from aircraft acceptance rates at destination airports.

Aircraft behavior

We use a *hybrid model* for each aircraft. A hybrid model describes the evolution of a system by a set of *discrete modes*, each associated with *continuous dynamical systems* and *discrete switches* which enable the system to jump from one mode to another instantaneously. In mathematical terms, we will describe the motion of aircraft i by:

$$\dot{\vec{x}}_i = \frac{d\vec{x}_i}{dt} = \vec{v}_{\text{current heading}} \qquad (1)$$

where $\vec{v}_{\text{current heading}} \in \mathbf{R}^2$ is a constant speed vector held by the aircraft until the next heading or speed change. $\vec{x}_i \in \mathbf{R}^2$ is the planar position of aircraft i. Integration of equation (1) over time produces a (continuous) piecewise affine trajectory. We prefer such a model over a completely continuous dynamical model for two main reasons: *(i)* the time scale of a "change in aircraft behavior" (for example turn or slow down) is on the order of 30 seconds; the time scale of a straight line portion of the flight is usually much longer, sometimes half an hour or more; thus we ignore the dynamics of such maneuvers and focus on their effects only (the set of resulting straight lines); *(ii)* the update rate of ATC monitoring is in general not more than 30 seconds, which makes the details of these maneuvers inaccessible to the ATC. This approximation is widely accepted in the literature.[8–13]

Once this setting is in place, the modeling is relatively straightforward. Monitoring ATC shows that a finite set of maneuvers, which depend on local parameters, is used. Combinations of these maneuvers result in a conflict-free flight environment with the constraints of the air traffic flow met. The following maneuvers are modeled (and consist of changing the right hand side of (1) according to certain rules which we now make explicit). The validity of this model has been confirmed by statistical studies realized at MIT ICAT (Histon and Hansman[14]).

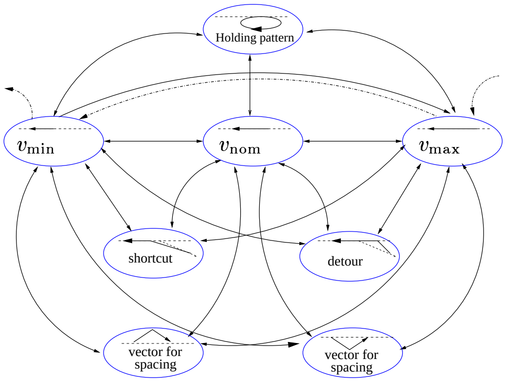

Fig. 2 Hybrid automaton representing the action of one controller on a single aircraft. Each of the eight modes represents one possible state of the aircraft. The arrows joining these states are the mode switches, initiated by the controller. The dash-dotted transitions are used for the analytical solution. The complete set of arrows is used for the simulation.

1. *Speed change:* The ATC decelerates or accelerates the aircraft along its flight plan:

$$\vec{v}_{\text{modified speed}} := \lambda \cdot \vec{v}_{\text{current heading}} \tag{2}$$

   where $\lambda \in \mathbf{R}^+$ defines the magnitude of the velocity change. Our model will allow a finite set of speeds (which means λ has a finite numbers of acceptable values). This encodes the fact that the ATC generally has a finite set of possibilities in the choice of speeds, because the aircraft flies at its optimal speed per altitude and ATC will speed up or slow down the aircraft by not more than 10% of the current value.

2. *Vector-for-spacing (VFS):* This maneuver consists of a deviation of the aircraft from its original flight plan for a short time (part 1 of the maneuver), and a second deviation, bringing it back to its original flight plan (part 2 of the maneuver). This stretches the path (and therefore generates delay). The length of this maneuver depends on the geometry of the sector. Calling R_ψ the rotation matrix by angle ψ, we have:

$$\begin{aligned}\vec{v}_{\text{part 1}} &:= R_\psi \cdot \vec{v}_{\text{current heading}}\\ \vec{v}_{\text{part 2}} &:= R_{-2\psi} \cdot \vec{v}_{\text{part 1}}\end{aligned} \tag{3}$$

3. *Shortcut / Detour:* In certain situations, the ATC will have the aircraft "cut" between two jetways, a maneuver which could either shorten or lengthen the flight plan. The decision to do such a maneuver is often dictated by conflict resolution, but could also be done to shorten the overall flight time if sector occupancy allows it (sometimes called "direct-to" by pilots):

$$\vec{v}_{\text{shortcut}} := R_\psi \cdot \vec{v}_{\text{current heading}} \tag{4}$$

4. *Holding pattern:* In some extreme conditions, holding patterns are used to maintain an aircraft in a flight loop in a given region of space before eventually letting it follow its original flight plan. This is modeled by assigning the aircraft to go to a predefined zone and keeping it there while preventing other aircraft from entering that zone.

Controller action

In practice, each controller is in charge of a sector dependent maximum number of aircraft (on the order of 25-30). The controller constantly cycles through the set of aircraft, accepting aircraft from neighboring sectors, giving directives, and handing them off to the next sectors. In our model, directives translate into switching conditions between modes. The controller is thus represented by an input control which actuates the aircraft, according to the different possible transitions shown in Figure 2.

Theory and analysis of delays generated by merging flows

A good proportion of air traffic jams is generated by restrictions imposed at destination airports, usually themselves driven by weather or airport congestion. These restrictions are imposed as either miles-in-trail or minutes-in-trail, representing the distance (or time) required between each aircraft in an incoming flow to the TRACON. They will be referred to as *metering constraints*. Figure 3 illustrates the topology of the flows incoming to San Francisco (SFO) which are often subject to this type of constraint. These constraints tend to propagate backwards from the airport into the network, and result in miles-in-trail constraints imposed at the entry points of each sector. For example,

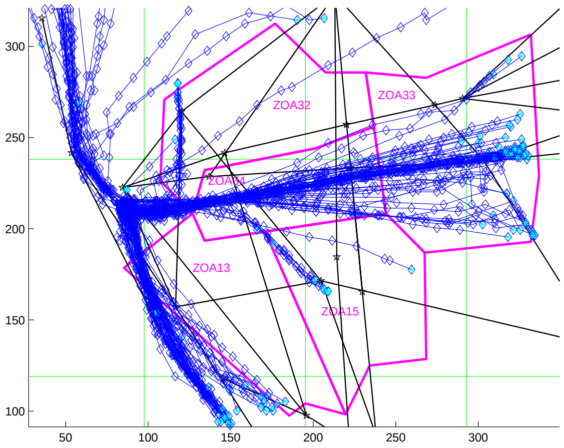

Fig. 3 Overlay of trajectories merging into San Francisco (11 hours of traffic). The data modeled comes from ETMS and FACET.[5]

in the case of Figure 3, typically, the backpropagation of these metering conditions is as follows: TRACON → sector 34 → sector 33 → Salt Lake Center $\cdots$ and similarly for the two other flows.

In the current system, these restrictions are imposed more or less empirically. We would like to understand *(i)* how the traffic jams propagate, *(ii)* what should be the optimal control policy to ensure maximal throughput into the TRACON under these restrictions.

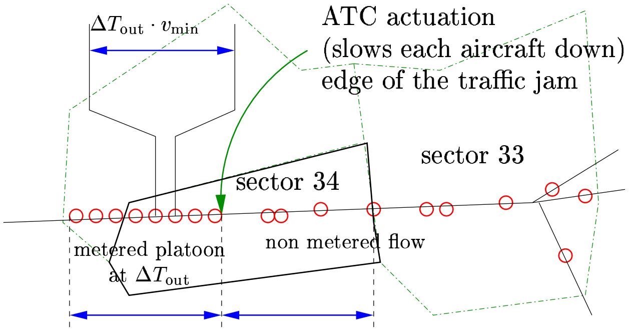
$\Delta T_{\text{out}} \cdot v_{\text{min}}$
ATC actuation
(slows each aircraft down)
edge of the traffic jam
sector 33
sector 34
metered platoon
at ΔT_{out}
non metered flow

Fig. 4 ATC control action on the merging flow. The traffic jam extends from SFO to the edge of the platoon metered by $\Delta T_{\text{out}} \cdot v_{\text{min}}$ (one aircraft every ΔT_{out} minutes). Once the actuation point (edge of the traffic jam) has moved upstream (into sector 33), sector 34 is called saturated.

Sector overload prediction

In Bayen and al.,[15] a simple model of merging flows which predicts the backpropagation of a traffic jam from a destination airport into the network, was introduced. This model can be applied to the merging flows of the type shown in Figure 3. We will use it here to derive the *dynamic capacity* of a sector, i.e. number of aircraft that can be actuated in it (given inflow and outflow conditions), before saturation is reached. We solved the following problem: *Given a metering constraint of ΔT_{out}, compute a controller policy which will force groups of aircraft to* exactly *satisfy the metering constraint at the sector exit point (each aircraft is separated by exactly ΔT_{out}) while maintaining separation.* For this, we introduced the initial position a_i^0 of aircraft i and the location x_{ex} where the metering condition is imposed, $i \in [1, N]$ where N represents the total number of aircraft. We considered the following problem: all aircraft are initially at maximal speed v_{max}, and in order to enforce metering, ATC slows down aircraft i to a minimum speed v_{min} (see Figure 4), at a location x_i^{switch} and time t_i^{switch} which we determined. This scenario is represented in dash-dot in Figure 2. We imposed that each aircraft crosses the metering point x_{ex} at exactly $t_{\text{block}} + (i-1)\Delta T_{\text{out}}$ where t_{block} is the time when the metering condition was imposed. This leads to the following kinematic equations of the aircraft:

$$\begin{array}{ll} x_i(t) = a_i^0 + v_{\text{max}} t & t \in [0, t_i^{\text{switch}}] \\ x_i(t) = b_i + v_{\text{min}} t & t \in [t_i^{\text{switch}}, t_{\text{block}} + (i-1)\Delta T_{\text{out}}] \end{array}$$

The assumption of continuity of $x_i(t)$ enables to solve for b_i, from which we deduce the switching time by: $t_i^{\text{switch}} = (b_i - a_i^0)/(v_{\text{max}} - v_{\text{min}})$. Under the following feasibility conditions,

$$\begin{array}{rl} & a_i^0 \geq x_{\text{ex}} - v_{\text{max}}(t_{\text{block}} - (i-1)\Delta T_{\text{out}}) \\ \text{and} & a_i^0 \leq x_{\text{ex}} - v_{\text{min}}(t_{\text{block}} - (i-1)\Delta T_{\text{out}}) \end{array} \tag{5}$$

we were able to compute analytically the propagation speed of the traffic jam, by solving for the location of the edge of the traffic jam in space and time:

$$\begin{array}{rcl} t_i^{\text{switch}} & = & \frac{x_{\text{ex}} - v_{\text{min}} t_{\text{block}} - (i-1)\Delta L - a_i^0}{v_{\text{max}} - v_{\text{min}}} \\ x_i^{\text{switch}} & = & a_i^0 + \frac{v_{\text{max}}[x_{\text{ex}} - v_{\text{min}} t_{\text{block}} - (i-1)\Delta L - a_i^0]}{v_{\text{max}} - v_{\text{min}}} \end{array} \tag{6}$$

where $\Delta L := v_{\text{min}} \Delta T_{\text{out}}$ is the metered spacing at the outflow of the sector. It follows directly from (6) that the traffic jam will not grow if the two following conditions are met:

$$\begin{array}{rcl} t_i^{\text{switch}} < t_{i+1}^{\text{switch}} & \Leftrightarrow & \Delta L < a_i^0 - a_{i+1}^0 \\ x_{i+1}^{\text{switch}} < x_i^{\text{switch}} & \Leftrightarrow & \left(\frac{v_{\text{min}}}{\Delta L}\right) < \left(\frac{v_{\text{max}}}{a_i^0 - a_{i+1}^0}\right) \end{array} \tag{7}$$

Using this, it is fairly easy to predict the *dynamic capacity* of a sector. We place ourselves in the worst case scenario: we consider an incoming flow of aircraft, each at v_{max}, separated in time by ΔT_{in} chosen to violate the second condition in (7). This will create a traffic jam originating at SFO, which “piles up” and progressively fills sector 34. Calling l the length of sector 34 along the main jetway, we use equations (6) to compute the maximal number of aircraft going through this sector before the edge of the traffic jam moves to sector 33 (*dynamic capacity*), and the time it needs to be reached. We get:

$$N_{\text{limit}} = \frac{l(v_{\text{max}} - v_{\text{min}})}{v_{\text{max}} v_{\text{min}} (\Delta T_{\text{out}} - \Delta T_{\text{in}})} \tag{8}$$

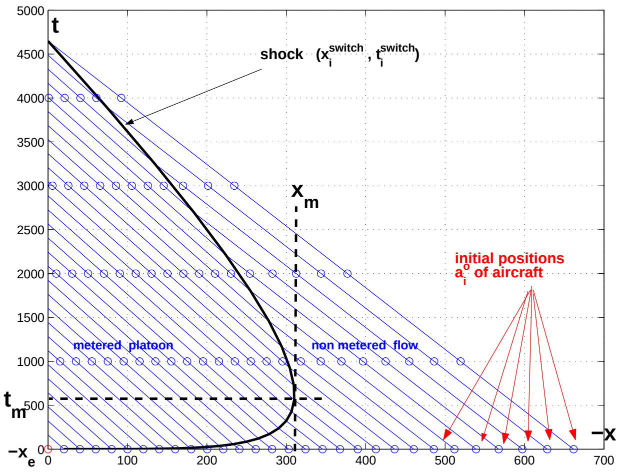

Fig. 5 Switching curve (shock) for a vanishing traffic jam. x denotes the distance to the metering point (SFO). The lines are the trajectories of the aircraft in the (x, t) space. The positions of aircraft are represented every 1000 sec. as dots. Once they have gone through the shock, they are metered (at one aircraft every $v_{\min}\Delta T_{\text{out}}$). The point (x_m, t_m) is the furthest reachable point by this traffic jam.

$$T_{\text{limit}} = \frac{l}{v_{\max}v_{\min}} \frac{v_{\max}\Delta T_{\text{in}} - v_{\min}\Delta T_{\text{out}}}{\Delta T_{\text{out}} - \Delta T_{\text{in}}} \tag{9}$$

Several comments can be made regarding the two previous results: *(i)* As $v_{\min} - v_{\max} \to 0$, $N_{\text{limit}} \to 0$: no aircraft can be handled in the sector additional to the aircraft already there, because no actuation is possible (it is not possible to "make the aircraft lose time" in this sector); *(ii)* As $\Delta T_{\text{out}} - \Delta T_{\text{in}} \to 0$, $N_{\text{limit}} \to \infty$ and $T_{\text{limit}} \to \infty$: if the incoming flow is such that it is almost metered as imposed at the exit of the sector, the number of aircraft required to saturate this airspace becomes large and the time it takes to saturate this sector grows accordingly.

Maximum extent of a traffic jam

The construction of the switching curve or *shock* $(x_i^{\text{switch}}, t_i^{\text{switch}})$ described previously, can be used to compute the maximal space extent of a traffic jam (defined as the portion of jetway affected by a traffic jam). Using (6), one can trace the shock location in the (x, t) plane (Figure 5). The maximum x, called x_m, obtained at t_m gives the worst situation which happens with this distribution of aircraft over time. In the case of Figure 5, we see that the traffic jam does not propagate more than 300 nm upstream from the sink, x_{ex}. Therefore no metering conditions should be applied upstream from that point. In the current system, such information is not available to the ATC, thus leading to extra buffers taken by the controllers, which in turn leads to non optimal operating conditions as well as backpropagation of "virtual overload", in fact a set of conservative precautions.

Air traffic flow simulations

The models of the previous section rely on a mathematical analysis of metering, based on geometry. In order to understand how realistic these models are, we need to validate them against the real system. Since the real system is not available as a testbed, we need an abstraction of it which reproduces its behavior adequately. For this, we have created a simulator which mimics true ATC behavior and against which we validate our predictions. It is itself validated against the real system (this will be the focus of the next section). This simulator is based on empirical studies that we realized at the Oakland ARTCC, its core is based on observed behavior. Figure 2 (all transitions enabled) summarizes the behavior model observed at the ARTCC. The switching logic behind the transitions is the object of this section, and is implemented in the form of a cost function, described below.

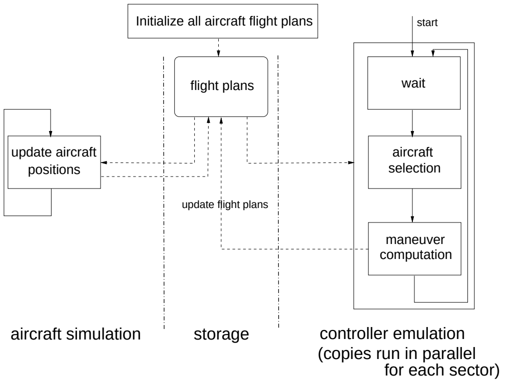

Fig. 6 Program flow of the simulator.

Overall Program Flow

The overall program flow of the simulator is shown in Figure 6. The input to the code is a set of aircraft filed flight plans (Figure 6, middle column), that can either be user generated or taken from ETMS data (as in FACET). As in the true system, these flight plans are not conflict free and usually do not satisfy metering conditions imposed on the network. Once the program is initialized, aircraft motion simulation follows these flight plans (Figure 6, left column). As time is advanced, conflict as well as metering constraints are dealt with on a sector by sector basis (with sector-wide look ahead, Figure 6, right column), according to the full automaton shown in Figure 2. The flight plans are updated accordingly.

Key Data Structures

Aircraft dynamic equations (1) produce a set of segments; the knowledge of the points connecting the segments and of the aircraft velocity is thus enough to completely define an aircraft trajectory. This trajectory is thus implemented as a *linked list* of *points* $[x, y, z]$, with a prescribed velocity between the points. The linked list is modified by the simulated controller in the program. The output for each aircraft is the updated linked list. The sectors are implemented as sets of accessible linked lists. They also contain additional data such as metering conditions (number of aircraft through a given boundary per time unit).

Controller Emulation

ATC behavior is modeled by three levels of priority: *Priority 1: No loss of separation.* The prevalent requirement for ATC is to ensure that any aircraft pair is always separated by more than 5 nautical miles. *Priority 2: Metering conditions.* The controller needs to ensure that the outflow from his sector is an acceptable inflow for the next sector (or TRACON). Metering conditions can be of various nature: admittance rate or separation at downstream junctions. *Priority 3: Best possible throughput.* ATC will try if possible to give direct routes to aircraft in order to minimize their flight time

These priorities are put in mathematical form with help of a cost function J, which we now define.

$$J = \mathrm{cost}_{\mathrm{LOS}} + \mathrm{cost}_{\mathrm{BC\ breach}} + \mathrm{cost}_{\mathrm{delay}} + \mathrm{cost}_{\mathrm{aircraft\ actuation}} + \mathrm{cost}_{\mathrm{maneuver}} + \mathrm{cost}_{\mathrm{min\ dist}}$$

Each term of the cost is a weighted function:

$$J = \sum_{i=1}^{N_{\mathrm{aircraft}}} \frac{n^i_{\mathrm{LOS}} \cdot w_{\mathrm{LOS}}}{\Delta T^i_{\mathrm{LOS}}} + \sum_{i=2}^{N_{\mathrm{aircraft}}} (T^i_{\mathrm{breach}})^2 \cdot w_{\mathrm{breach}} + \sum_{i=1}^{N_{\mathrm{aircraft}}} J^i_{\mathrm{maneuver}} + \sum_{i=1}^{N_{\mathrm{aircraft}}} (TOA^i_{\mathrm{pred}} - TOA^i_{\mathrm{real}}) \cdot w_{\mathrm{delay}} + N_{\mathrm{moved}} \cdot w_{\mathrm{single\ move}} + \sum_{i=1}^{N_{\mathrm{max}}} f(d^i_{\mathrm{min}}) \cdot w_{\mathrm{dist}}$$

1. *Loss of separation (LOS) cost:* n^i_{LOS} is the number of losses of separation involving aircraft i in the current sector with its current flight plan. $\Delta T^i_{\mathrm{LOS}}$ is the time until the first loss of separation for aircraft i.
2. *Boundary breach cost:* T^i_{breach} is the time by which an aircraft violates the ΔT time separation constraint from its predecessor (set to zero if the two aircraft are separated by more than ΔT).
3. *Delay cost:* $TOA^i_{\mathrm{pred}} - TOA^i_{\mathrm{real}}$ accounts for the difference between predicted and actual time of arrival (TOA) at the last waypoint of the flight. Positive delays are penalized; earlier arrivals are favored. TOA^i_{pred} and TOA^i_{real} are computed by integration of the flight plans for each aircraft.
4. *Aircraft actuation cost:* N_{moved} accounts for the number of flight plan modifications chosen in the current solution. Large N_{moved} are penalized (the solution chosen by the ATC is often the simplest).
5. *Maneuver cost:* J^i_{maneuver} accounts for the cost of the maneuver selected for aircraft i. Not all maneuvers are of equal preference and therefore have different costs. It is easier for a controller to prescribe a speed change than a VFS or a shortcut. A holding pattern is the least liked option, for it requires constant monitoring of the aircraft. This reflects in the weight choice:

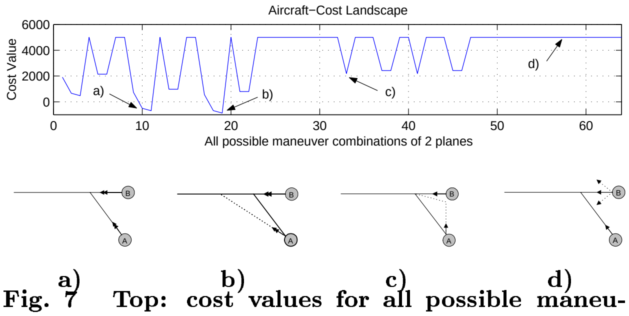

Fig. 7 Top: cost values for all possible maneuver combinations in a two-aircraft intersection scenario, where the eight maneuvers of Figure 2 are enabled (thus generating $8^2 = 64$ possible values of J). Four out of 64 examples are extracted and illustrated (four lower pictures). (a) Both aircraft maintain same speed; (b) Aircraft A takes a shortcut maintaining aircraft B at max speed; (c) A makes a VFS at low speed; (d) A does nothing, B is not able to prevent the loss of separation. In the presented case, the simulated controller would choose solution (b) since the lowest cost is associated with that maneuver.

$J^i_{\mathrm{speed\ change}} < J^i_{\mathrm{shortcut}} \sim J^i_{\mathrm{VFS}} \ll J^i_{\mathrm{holding\ pattern}}$
6. *Minimal distance cost:* $f(d^i_{\mathrm{min}})$ penalizes aircraft distributions in which aircraft are closely spaced (but do not lose separation) against more sparse distributions. Here, $\mathrm{dist}_{\mathrm{max}} = 7$nm.

$$f(d^i_{\mathrm{min}}) = \frac{1}{d^i_{\mathrm{min}}} \cdot w_{\mathrm{dist}} \quad \text{if } d^i_{\mathrm{min}} < \mathrm{dist}_{\mathrm{max}}$$
$$f(d^i_{\mathrm{min}}) = 0 \quad \text{otherwise.}$$

In order to reflect the three levels of priority stated above, the weights shown in the cost function J are: $w_{\mathrm{LOS}} \sim 10^{300} \gg w_{\mathrm{breach}} \sim 10^4 \gg$ other weights ~ 10. Minimizing J will thus first deal with losses of separation, then metering conditions, and finally optimization of the flow. An example of a cost landscape[1]

In order to reduce the computation time, we define as N_{choice} the maximum number of aircraft considered by the simulated controller in each time iteration. Aircraft are selected according to the following rule: aircraft involved in LOS are selected first, then aircraft breaching boundary conditions, and finally remaining aircraft until the selection list has reached N_{choice} aircraft, or until there are no more aircraft to select. In practice, $4 \leq N_{\mathrm{choice}} \leq 8$, where $N_{\mathrm{choice}} = 8$ enables more complicated situations but makes the code run slower. The set of all maneuver combinations for the N_{choice} aircraft is called the maneuver set. At each iteration of the controller emulation loop, an exhaustive recursive search on the maneuver set is run in order to find a set of N_{choice} maneuvers which minimizes J. The computational complexity of finding the optimal J for N_{choice} aircraft subject to n_{maneuver} possible discrete maneuvers is $O((n_{\mathrm{maneuver}})^{N_{\mathrm{choice}}})$. We can reduce this cost to

[1] In Figure 7, the cost J has been truncated at $5 \cdot 10^3$ for readability. Thus a LOS cannot be visually differentiated from a breach in this plot, though it can in our data.

$O((n_{\text{maneuver}}-2)^{N_{\text{choice}}})$: (i) the cost of the current maneuver has already been computed at the previous step and thus does not need to be recomputed; (ii) two maneuvers are mutually exclusive (shortcuts), therefore only one needs to be called. Including the cost of checking for conflicts, the total cost of a time iteration becomes: $O\left(N_{\max}^2 \cdot (n_{\text{maneuver}}-2)^{N_{\text{choice}}}\right)$ where $N_{\max}$ represents the total number of aircraft in the sector. Due to both the discretization of time, and the restriction of the search space to a manageable number of aircraft, our search is not guaranteed to find a theoretically optimal solution. However, we believe that the search does provide a reasonable approximation of the controller's behavior. By adjusting the two key control parameters, the number of selected aircraft N_{choice} and time between controller activation ΔT_{act}, a transparent trade-off between run-time and control quality can be made.

Code validation against ETMS data

The controller logic presented in the previous section is the result of numerous observations we made in the Oakland ARTCC, monitoring the work of air traffic controllers. The fact that one can classify ATC action into a set of favorite directives was experimentally validated for a different airspace (see Hansman and Histon[14]). However, even if the automaton of Figure 2 and the cost function of the previous section implemented in the simulator are consistent with our observations, nothing guarantees that their use will provide the same type of flow as an actual controller would generate. Furthermore, we would like to assess how well our cost function describes the decision making of a human controller. We therefore need to validate our code against recorded aircraft trajectories. We use ETMS data provided by NASA Ames.[2]

Data extraction

We can extract two types of information from the ETMS data: the actual flown aircraft trajectories, and the filed flight plans for each aircraft (eventually updated if modifications are made during the flight). The position is given in latitude / longitude and in terms of navaids, fixes, jetways which we look up in a database.

This study is limited to sector control, and we did not implement Traffic Management Unit (TMU) action. TMU operates at the Center level and makes strategic flow scheduling decisions, which go beyond the range of a single sector controller. We therefore need to validate our simulator at a scale where TMU actuation is already incorporated in the flight plans (typically one or two sectors). Since our interest focuses on sectors 32, 33, 34, 15, and 13, the actual flight plans are truncated, and we keep only points in that sector. The filed flight plans are truncated similarly.

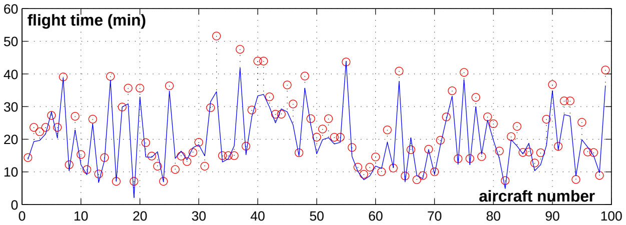

Fig. 8 Flight time comparisons for the first 100 aircraft going through sector 33 in the ETMS data set we used. The dots are the flight times for the ETMS recorded points. The solid curve is the result of our simulations.

Validation

Comparison of flight times. We select the first 100 aircraft of the ETMS data set which are flying above 33000ft for more than 6 minutes. Their recorded trajectories are extracted as sequence of waypoints which are used as initialized flight plans for our simulations. We compare the flight time in the simulation to the real one. The experiment is run for the following set of speeds: $M \in \{0.6, 0.7, 0.8\}$ (M is the Mach Number), which matches what we see in the data for this altitude. In the run, the controller is activated every $\Delta T_{\text{act}} = 10$sec. The results are shown in Figure 8. Two main conclusions can be made: *(i)* The simulator is able to recreate the flow without major modification, and eventually resolves apparent conflicts in the data; these conflicts can be due to inaccuracy of the measurements or transmission (two aircraft separated by less than 5 nm at the same altitude), or due to problems of interpolation when speed changes in time; *(ii)* The time comparison (Figure 8) shows relatively good matching. The flight times provided by the simulator are usually shorter because by default the simulator will always try to maximize the throughput in the sector. The mean deviation is 120 sec. for flight plans with an average duration of 1300 sec (9.2% error).

Verification of conflict resolution. We select aircraft flying through sector 33 in a time frame of 10 hours (a total set of 314 aircraft) and simulate these flights.The filed flight plans are not conflict free. We want to show that the simulator is able to provide a conflict free environment. For this run, the activation time of the controller is $\Delta T_{\text{act}} = 20$sec.[3] The set of speeds allowed is $M \in \{0.55,\ 0.75,\ 0.89\}$ (since we are considering the full range of altitude, we need to consider the full

[2] Data is collected from the entire population of flights with filed flight plans in the NAS. ETMS data is sent from the Volpe National Transportation System Center (VNTSC) to registered participants via the Aircraft Situation Display to Industry (ASDI) electronic file server. A file containing all recorded data is generated. It displays for each aircraft the current flight data (time, position, speed, heading), as well as the filed flight plan (in terms of navaids, jetways, fixes, etc.). The update rate of the measurements is on the order of one minute.

[3] $\Delta T_{\text{act}} = 10$sec. or $\Delta T_{\text{act}} = 20$sec. is on the order of the maximal actuation rate of a controller. We chose $\Delta T_{\text{act}} = 20$sec. in this particular case because of the duration of the computation (10 hours of real time are simulated).

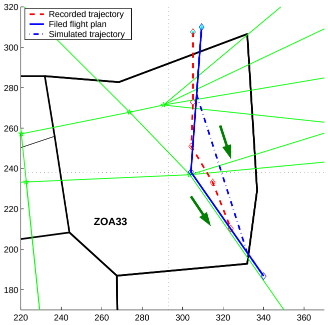

Fig. 9 An example of maneuver caused by conflict resolution, reproduced by the simulator. The recorded data (dashed) exhibits a shortcut from the filed flight plan (solid). The simulated trajectory (dashed-dotted) is a shortcut of the same type.

range of speed). The simulator is able to provide a conflict free environment. During the simulation, it has to actuate 50 different aircraft. The number of resolved conflicts can be assumed to be on the same order, since a single intervention will usually resolve not more than one conflict.

Validation of maneuver assignments. The core of the simulator is the model of the human controller by a decision procedure based on a cost function described previously. The validation so far has shown the correlation of flow patterns generated by our code and these observed in real life. We now would like to assess the validity of our decision procedure. For this, we identify conflict resolution maneuvers. They are typically obtained by identifying deviations from the filed flight plan. For these maneuvers, we generate the following input data. All aircraft are assigned their actual flight plan (recorded trajectories). The aircraft for which the maneuver was identified is assigned its filed flight plan (a set of way points). We thus put the simulator in the same situation as the human controller, where it has to make the decision that was actually taken. For sector 33, we were able to identify 20 distinct maneuvers out of 300 examined flight plans. The simulator reproduced 16 of them.[4] An example is shown in Figure 9.

Predictive results

In this section, we assess how accurate our analytical predictions are with respect to the real system, by comparing them with simulations. Then we use our simulator to show a decorrelation between conflict resolution and metering, used in our current work.

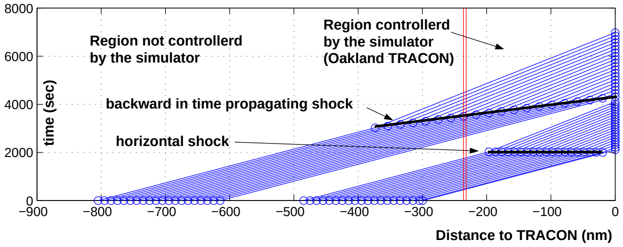

Fig. 10 Shocks generated by two successive platoons. The first shock is steady in time (it only propagates backward in space). It corresponds to a piling up process on a highway where all vehicles slow down at the same time. The second shock propagates backward in space and time (which is much harder to handle in practice, because actuation must be performed upstream first).

Sector overload

We present an example of two backpropagating shocks, solved with an extension of the method explained previously.[15] Two platoons, each at 10 miles-in-trail, are respectively subjected to 15 miles-in-trail and 20 miles-in-trail outflow boundary conditions (the boundary conditions of platoon 2 start after platoon 1 has exited the area, at time $t = 4300$). The speeds are $M \in \{0.59,\ 0.75,\ 0.89\}$. The aircraft flows for this run are shown in Figure 12. The results and interpretations are shown in Figures 10 and 11. Two shocks appear successively. From Figure 10, we can see that within the second platoon, the first twelve aircraft need to be actuated within the Oakland Center, whereas the last eight need to be actuated upstream (Salt Lake Center). Since in general, no knowledge of the required boundary conditions is propagated upstream, we can predict that in the real system, the last eight aircraft would not be actuated until they enter the Oakland Center and that no solution to this metering problem would be found without putting the aircraft on hold. We verify this by simulating this flow: in Figure 11, we see that for the last eight aircraft, the first activation time in the simulator is higher than the predicted (upper plot): the simulated controller is not able to actuate the aircraft on time, because they are not in its airspace. We can verify on the middle plot that these aircraft are breaching the boundary conditions (by about one minute each), which can also be seen in the lower plot: their delays, (i.e. the inverse of the cumulated breaches) become negative. This is an illustration of distributed and decentralized control: the actuation occurs in different sectors, and the only communication between them is through the metering conditions. Obviously, the lack of the centralized actuation (here communication and strategic TMU planning) disables efficient flow scheduling.

[4] Small-scale maneuvers are less likely to be executed correctly by the simulator because the probability of selecting the respective aircraft at exactly the 'right' time is small, which explains the small discrepancy between the results. Also, even if the maneuver is executed correctly by the simulator, the resulting flight plan will look different from the ETMS data, since the simulator is restricted to a single angle of deviation ($\theta = 22.5°$).

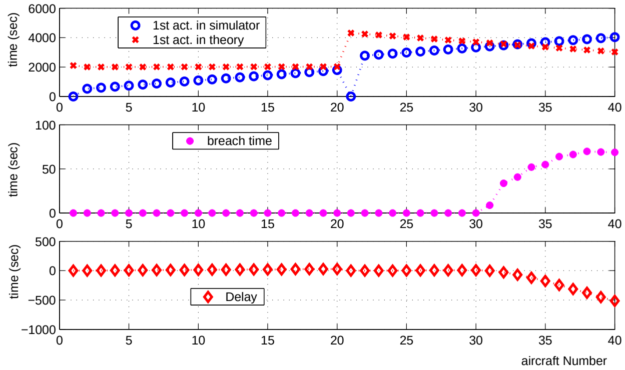

Fig. 11 First actuation times of the aircraft (upper plot), simulated and predicted; breaches in metering conditions (middle plot), simulated; delays (lower plot), simulated, for the case of the two platoons of Figure 10.

Decorrelation of cross traffic in metered flows

A working assumption which is often made in flow scheduling is the decorrelation of metering conditions and cross traffic when traffic density is relatively low (i.e. conflict avoidance actuation does not impact metering actuation). We display this property of the flow by simulating two streams of intersecting (and conflicting) aircraft (see Figure 13). Two platoons of aircraft intersect at a navaid (Clovis, in Sector 15), and we consider pairwise conflicts. One platoon is subject to outflow metering conditions (at the boundary of sector 15 and 34), and we would like to quantify the impact of the other platoon on the travel time of the first platoon through the sector.

We investigate 16 different configurations: we select the same inflow conditions for the two platoons: $\Delta L_{\text{in}} \in \{15, 20, 25, 30\}$ miles-in-trail, and the following outflow conditions for the platoon heading towards sector 34, $\Delta L_{\text{out}} \in \{15, 20, 25, 30\}$ miles-in-trail (see Figure 13). We compare the travel time for each platoon without the presence of the other with the travel time when the two platoons intersect. For each configuration, we do 10 runs where the initial position of the aircraft within each platoon is perturbed by a uniform noise of amplitude 2 nm. This scenario is the worst case possible for this type of cross flow: all aircraft for the two respective platoons conflict pairwise. This results in 160 runs.[5] For each $(\Delta L_{\text{in}}, \Delta L_{\text{out}})$, we compute the three following quantities

1. $\frac{1}{N}\sum_{i=1}^{N}\left(T^{\text{no cross flow, B.C.}}_{\text{aircraft } i} - T^{\text{no cross flow, no B.C.}}_{\text{aircraft } i}\right)$
2. $\frac{1}{N}\sum_{i=1}^{N}\left(T^{\text{cross flow, B.C.}}_{\text{aircraft } i} - T^{\text{no cross flow, no B.C.}}_{\text{aircraft } i}\right)$
3. $\delta T^{\text{cross flow}}_{\text{travel time}} - \delta T^{\text{no cross flow}}_{\text{travel time}}$

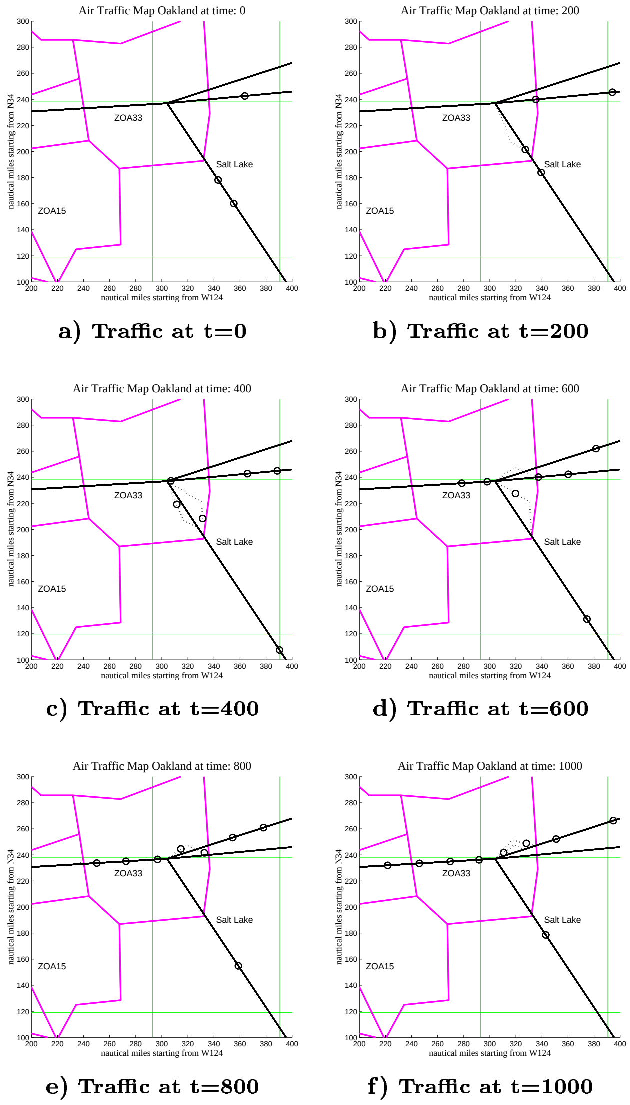
Air Traffic Map Oakland at time: 0
Air Traffic Map Oakland at time: 200
Air Traffic Map Oakland at time: 400
Air Traffic Map Oakland at time: 600
Air Traffic Map Oakland at time: 800
Air Traffic Map Oakland at time: 1000
ZOA33
ZOA15
Salt Lake
nautical miles starting from W124
nautical miles starting from N34
a) Traffic at t=0
b) Traffic at t=200
c) Traffic at t=400
d) Traffic at t=600
e) Traffic at t=800
f) Traffic at t=1000

Fig. 12 Sector 33: traffic flow for the merging traffic simulation of of Figures 10 and 11. The radius around the aircraft is 2.5 nm. The solid lines represent the aircrafts' flight plan. The dotted lines correspond to maneuvers assigned by the simulator. The simulator makes extensive use of the vector-for-spacing to meter the aircraft.

called respectively $\delta T^{\text{no cross flow}}_{\text{travel time}}$, $\delta T^{\text{cross flow}}_{\text{travel time}}$ and $\Delta T_{\substack{\text{due to}\\ \text{cross flow}}}$. Here, N is the total number of aircraft ($N = 20$, we have two platoons of 10 aircraft). In each of the formula above, $T^{\text{no cross flow, B.C.}}_{\text{aircraft } i}$ represents the travel time of aircraft i in absence of the other platoon, and with outflow boundary conditions as shown in Figure 13 (and similarly for the other subscripts). The results for the mean difference in travel time is shown in Figure 14 (averaged over 10 runs for each case). We show the numerical results obtained respectively in the $\Delta L_{\text{in}} = 15$ and $\Delta L_{\text{in}} = 20$ miles-in-trail inflow case,

ΔL_{out}	$\delta T^{\text{no cross flow}}_{\text{travel time}}$	$\delta T^{\text{cross flow}}_{\text{travel time}}$	$\Delta T_{\substack{\text{due to}\\ \text{cross flow}}}$
15nm	0.7185sec.	140.54sec.	139.82sec.
20nm	64.8sec.	166.93sec.	102.13sec.
25nm	90.761sec.	157.08sec.	66.32sec.
30nm	104.65sec.	164.86sec.	60.203sec.

(10)

[5] The settings for these runs are: $M \in \{0.8,\ 0.85,\ 0.89\}$, the vector-for-spacing maneuver was limited to a 5 nm deviation from the original flight plan. These settings were chosen to guarantee short flight times. The interval between controller activation was set to $\Delta T_{\text{act}} = 20sec$.

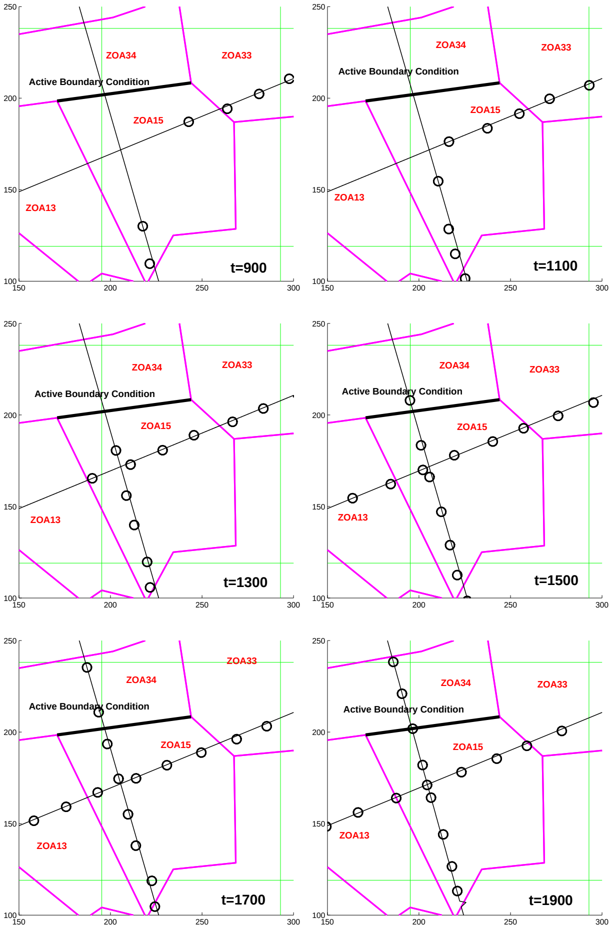

Fig. 13 Two intersecting platoons at the Clovis navaid for the configuration where the aircraft have an inflow of 20 miles-in-trail. Aircraft are pairwise conflicting (aircraft i from platoon 1 conflicts with aircraft i of platoon 2), assuming flight plan is to follow straight line at initial speed. The simulator is thus forced to adjust each of the pairs. Deviations (vector-for-spacing) are barely visible on the plot because of their small amplitude (5nm).

ΔL_{out}	$\delta T^{\text{no cross flow}}_{\text{travel time}}$	$\delta T^{\text{cross flow}}_{\text{travel time}}$	$\Delta T_{\text{due to cross flow}}$
15nm	0sec.	28.155sec.	28.155sec.
20nm	0.704sec.	35.95sec.	35.245sec.
25nm	74.255sec.	101.89sec.	27.636sec.
30nm	99.574sec.	123.89sec.	24.314sec.

(11)

$\Delta T_{\text{due to cross flow}}$ is represented in Figure 14 for the complete set of $(\Delta L_{\text{in}}, \Delta L_{\text{out}})$ investigated here. Even if the peak of $\Delta T_{\text{due to cross flow}}$ happens for $(\Delta L_{\text{in}}, \Delta L_{\text{out}}) = (15, 15)$, the maximum $\delta T^{\text{cross flow}}_{\text{travel time}}$ happens as expected for $(\Delta L_{\text{in}}, \Delta L_{\text{out}}) = (15, 30)$, which is the maximal inflow, minimal outflow, as can be seen in (10). The comparison between (10) and (11) and Figure 14 clearly shows the predominance of conflict resolution over boundary conditions for high density of traffic (see last column in (10) and (11)).

We see that the difference in delay between separate and simultaneous flow is significantly larger if the aircraft are spaced at 15 nm when compared to looser platoons (see Figure 14). While the mean difference per plane is always larger than 60 seconds for the 15 nm-platoons, it is always smaller than 60 seconds for looser spacing. With an average flight time of 660 seconds over all scenarios, 60 seconds corresponds to an average delay of 9% in flight time. The worst case difference (15 nm inflow, 30 nm outflow) is more than 21% of the overall flight time. These numbers are significant, especially when considering the possibility of multiple intersecting platoons.

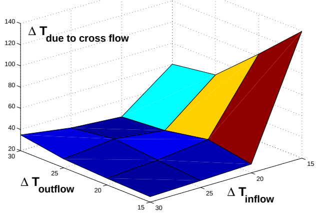

Fig. 14 Plot of $\Delta T_{\text{due to cross flow}}$ averaged over 10 runs.

Note that in Figure 14, intuitively we would expect the peak of difference in delay (cross flow vs. no cross flow) to happen for $(\Delta L_{\text{in}}, \Delta L_{\text{out}}) = (15, 30)$, which is the hardest to achieve situation (maximum inflow, most restricted outflow). In fact this maximum happens at $(\Delta L_{\text{in}}, \Delta L_{\text{out}}) = (15, 15)$, which a priori does not seem to be problematic (conservation of flow). This can be explained by looking at Figures 15 and 16. In absence of cross flow, the delay accumulated because of the boundary conditions is maximal for $(\Delta L_{\text{in}}, \Delta L_{\text{out}}) = (15, 30)$, as expected (Figure 16). When subtracted from the difference in travel time with cross flow (Figure 15), a max appears at $(\Delta L_{\text{in}}, \Delta L_{\text{out}}) = (15, 15)$, because $\delta T^{\text{no cross flow}}_{\text{travel time}}|_{(\Delta L_{\text{in}}, \Delta L_{\text{out}})=(15,15)} = 0$. These results are thus perfectly consistent. The flows of Figure 13 are worst cases, and it is rare to observe such a density in both directions. However, some sectors (for example sector 33) are often subject to less structured but as dense cross flows, and our predictions are relevant for this type of airspace.

Currently the TMU does not take the influence of conflict resolution into account when making decisions, since the impact of ATC on the sector level is considered to be negligible for overall flight times. In the current system this is not always true thus leading to inaccuracies in the predictions of sector occupancy, i.e. how many aircraft will be a sector in $t + 15$ minutes. We thus have shown in a particular case, that the influence of conflict resolution actuation increases with higher traffic density, which therefore necessitates information feedback from the sector to the Center.

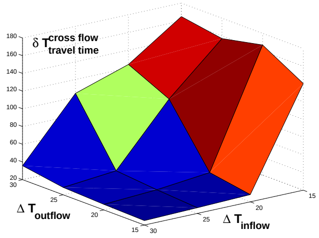

Fig. 15 Plot of $\delta T^{\text{cross flow}}_{\text{travel time}}$ averaged over 10 runs.

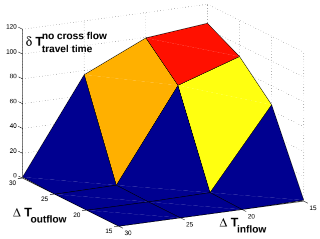

Fig. 16 Plot of $\delta T^{\text{no cross flow}}_{\text{travel time}}$ averaged over 10 runs.

Summary and current work

We have derived a control theoretic model of sector-based traffic flow using hybrid automata theory. We used a subset of this model to generate analytic predictions of the traffic flow (dynamic sector capacity, extend of traffic jams). These predictions were validated against an abstraction of the real system (a simulator using the full model, which we validated against real data). We finally generated flow conditions under which we can decorrelate metering from conflict resolution.

This last result is useful for our current work, because it enables us to ignore conflict resolution for scheduling. We generated a scheduling algorithm to draw the same predictions as the analytic shock construction derived here, in the more realistic case where the agents (aircraft) are now allowed to evolve on a truly two-dimensional environment (a directed acyclic graph, see Figure 17). This formulation relies on a continuous sequencing problem with release times and deadlines (whose discrete version is known to be NP-complete). We pose the continuous sequencing problem as a mixed linear integer program which we can solve with existing tools. It is then embedded in a graph search procedure which solves for the best possible scheduling under workload constraint (i.e. FAA approved sector capacity). We thus hope to perform more realistic predictions of the backpropagation of traffic jams in overloaded sector areas.

References

[1] Plaettner-Hochwarth, J., Zhao, Y., and Robinson, J., "A Modularized Approach For Comprehensive Air Traffic System Simulation," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Denver, CO, August 2000.

[2] Erzberger, H., Davis, T. J., and Green, S., "Design of Center-TRACON Automation System," *Proceedings of the AGARD Guidance and Control Symposium on Machine Intelligence in Air Traffic Management*, Berlin, Germany, 1993, pp. 11.1–11.12.

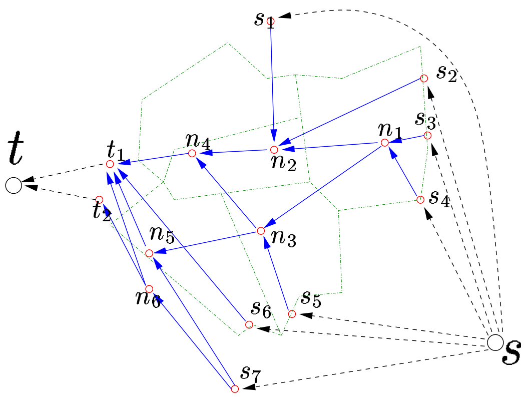

Fig. 17 Graph model used for the current implementation of the delay prediction algorithm.

[3] Odoni, A., "Existing and Required Modeling Capabilities for Evaluating ATM Systems and Concepts," Tech. rep., Massachusetts Institute of Technology, 1996, DRAFT Report for NASA AATT program.

[4] Kuchar, J. and Yang, L., "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 1, No. 4, 2000.

[5] Bilimoria, K., Sridhar, B., Chatterji, G., Seth, K., and S.Graabe, "FACET: Future ATM Concepts Evaluation Tool," *3rd USA/Europe Air Traffic Management R&D Seminar*, Naples, Italy, June 2001.

[6] Bilimoria, K. and Lee, H., "Properties of Air Traffic Conflicts for Free and Structured Routing," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Montreal, August 2001.

[7] Jeppesen, "High Altitude Enroute Charts," Tech. rep., http://www.jeppesen.com, February 2000.

[8] Dugail, D., Mao, Z.-H., , and Feron, E., "Stability of Intersecting Aircraft Flows Under Centralized and Decentralized Conflict Avoidance Rules," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Montreal, August 2001.

[9] Pallottino, L., Bicchi, A., and Feron, E., "Mixed Integer Programming for Aircraft Conflict Resolution," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Montreal, August 2001.

[10] Bilimoria, K., "A Geometric Optimization Approach to Aircraft Conflict Resolution," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Denver,CO, August 2000.

[11] Mao, Z., Feron, E., and Bilimoria, K., "Stability of Intersecting Aircraft Flows Under Decentralized Conflict Avoidance Rules," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Denver,CO, August 2000.

[12] Nilim, A., El-Ghaoui, L., Duong, V., and Hansen, M., "Trajectory-based Air Traffic Management (TB-ATM) under weather uncertainty," *Proceedings of the 4rd USA/Europe Air Traffic Management R&D Seminar*, Santa Fe, NM, December 2001.

[13] Bertsimas, D. and Stock Patterson, S., "The air traffic flow management problem with enroute capacities," *Operations Research*, Vol. 46, 1998, pp. 406–422.

[14] Histon, J. and Hansman, R. J., "The Impact of Structure on Cognitive Complexity in Air Traffic Control," Tech. rep., MIT, June 2002.

[15] Bayen, A., Grieder, P., Sipma, H., Meyer, G., and Tomlin, C. J., "Delay predictive models of the National Airspace System using hybrid control theory," *2002 American Control Conerence*, Anchorage, AK, May 2002.