DECENTRALIZED AIR TRAFFIC MANAGEMENT SYSTEMS: PERFORMANCE AND FAULT TOLERANCE

A. Bicchi * **A. Marigo** * **G. Pappas** ** **M. Pardini** *
G. Parlangeli * **C. Tomlin** ** **S. S. Sastry** **

* *Centro "E. Piaggio, Università di Pisa, Via Diotisalvi 2, 56100 Pisa, Italy*
** *Department of Electrical Engineering and Computer Science, University of California at Berkeley, CA 94720, USA*

Abstract:
In this paper, we investigate the advantages and disadvantages of a proposed decentralized Air Traffic Management (ATM) system, in which aircraft are allowed a larger degree of autonomy in deciding their route ("free flight") instead of following prespecified "sky freeways". A decentralized management scheme is described, which incorporates a hybrid control system switching among different optimal control solutions according to changes in the information structure between agents. The performance and robustness of this decentralization scheme is assessed by means of simulation trials, showing that, whereas optimality of plans is strictly nonincreasing, robustness to system failures is likely to improve.

Keywords: Hybrid systems, Optimal control, Decentralized control.

1. INTRODUCTION

In this paper, we discuss our work towards the design and implementation of a *Flight Management System* (FMS) as part of an *Air Traffic Management* (ATM) system architecture allowing for multi-aircraft coordination maneuvers, which are guaranteed to be safe.

In the proposed ATM architecture of Tomlin *et al.* (1997), we assume that general flow control, delay absorption, and scheduling, are performed by a centralized ATC authority (assisted by automation tools). Each aircraft in the system has an initial flight plan which has been designed by this central authority, and is encoded as a sequence of way points from origin to destination. The planning and control hierarchy on board each aircraft uses this sequence of way points as input, and generates a full state trajectory (including the sequences of autopilot modes) for the aircraft. However, unmodeled "disturbances" in the air traffic system (such as bad weather, wind, mechanical problems with a single aircraft) can force the aircraft to deviate from their original flight paths. In such situations, aircraft should be able to resolve "local" deviations, such as within a sector away from a TRACON, or over oceanic airspace, without requesting clearances from ATC. The proposed system is thus a *partially decentralized system*: we plan to focus our attention on a decision support tool for such an architecture.

A number of issues should be considered when deciding on the appropriate level of centralization. An obvious one is the *optimality* of the resulting design. Even though optimality criteria may be difficult to define for the air traffic problem it seems that, in principle, the higher the level of centralization the closer one can get to the globally optimal solution. However, the complexity of the problem also increases in the process; to implement a centralized design one has to solve a small number of complex problems as opposed to large number of simple ones. As a consequence the implementation of a centralized solution requires a greater effort on the part of the designer to produce control algorithms and greater computational power to execute them. Another issue

[1] Joint work conducted with support of grant NATO CR no. 960750; work at the Univ. of Pisa partially supported through grant ASI ARS-96-170; work at the Univ. California, Berkeley, partially supported by ARO under MURI program DAAH04–96–1–0341, and by NASA under grant NAG 1–1039

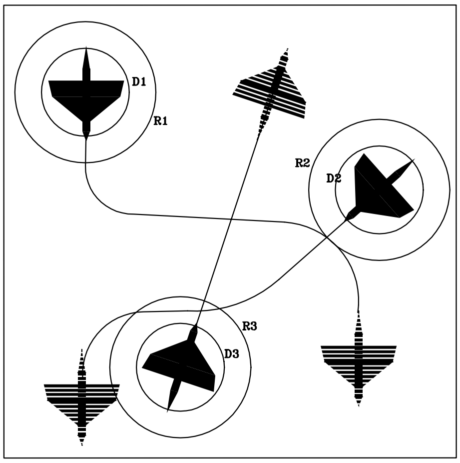

Fig. 1. Three aircraft with initial and final via-points assigned.

that needs to be considered is *robustness*. The greater the responsibility assigned to a central controller the more dramatic are likely to be the consequences if this controller fails. In this respect there seems to be a clear advantage in implementing a decentralized design: if a single aircraft's computer system fails, most of the ATM system is still intact and the affected aircraft may be guided by voice to the nearest airport. Similarly, a distributed system is better suited to handling increasing numbers of aircraft, since each new aircraft can easily be added to the system, its own computer contributing to the overall computational power. Finally, the issue of *flexibility* should also be taken into account. A decentralized system will be more flexible from the point of view of the agents, in this case the pilots and airlines. This may be advantageous for example in avoiding turbulence or taking advantage of favorable winds, as the aircraft will not have to wait for clearance from ATC to change course in response to such transients or local phenomena.

2. DECENTRALIZED ATMS

In order to quantitatively evaluate the effects of decentralization on performance and robustness, we introduce a much simplified, yet significant model of the air traffic control problem. Consider the problem of steering N agents (aircraft) among $2N$ given via–points (see Figure 1). Aircraft are assumed to move on a planar trajectory (in fact, all fly within a given altitude layer), and their cruise speed is constant. The kinematic motion of the i–th aircraft is modeled as

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \cos\theta_i \ u_i \\ \sin\theta_i \ u_i \\ \omega_i \end{bmatrix} \tag{1}$$

where $\xi_i = [x_i, y_i, \theta_i]^T$ is the state vector (comprised of x, y position and heading angle θ), $u_i =$ constant is the cruise speed. Start and goal via-points are $\xi_{i,s}$ and $\xi_{i,g}$, respectively. We assume for simplicity that motions are synchronized, i.e. all aircraft are at the start via–point at the same time, and denote by T_i the time at which the the i–th aircraft reaches its goal. The cost to be minimized in choosing the control inputs (yaw rates) ω_i is the total time–to–goal, or equivalently the total path length (proportional to fuel consumption),

$$\mathbf{J} = \sum_{i=1}^{N} \int_0^{T_i} dt = \sum_{i=1}^{N} T_i \tag{2}$$

Limitations of feasible yaw rates for practical airplanes are incorporated in this model by setting

$$\omega_i \in [-\Omega_i, \Omega_i]. \tag{3}$$

Besides the initial and final state constraints, and the input constraints (3), a constraint of no–collision along solutions must be enforced. Typically, such a constraint is embodied by non intersection of safety discs of radius D_i centered at the aircraft positions, namely by $\frac{N(N-1)}{2}$ inequalities on states of type

$$V_{ij}(\xi_i, \xi_j) = (x_i - x_j)^2 + (y_i - y_j)^2 - D_i^2 \geq 0 \tag{4}$$

It should be noted that, although the Euclidean distance used in the above definition of the constraint does not appear to take into account the fact that the nonholonomic kinematics (1) induce a more complex metric on the space (compare work of Laumond and co-workers, Laumond (1986), Bui *et al.* (1994)), (4) is typical in current ATC systems.

In a centralized ATMS, the above optimal control problem would be solved by ground–based Air Traffic Control, using one of several available numerical techniques. Notice that, even in this simplified setup, inequality constraints on states at interior points may generate difficulties in numerical integration. In our simulations, we adopt the suboptimal strategy introduced in Bicchi *et al.* (1998), and described below in some detail.

In decentralized ATMS schemes, each agent (aircraft) is allowed to take decisions autonomously, based on the information that is available at each time. Several models of decentralized ATC are conceivable, which may differ in the degree of cooperative/competitive behaviour of the agents, and in the information structure. For instance, a noncooperative, zero–sum game of the agents is a conservative (worst–case) approach to guarantee safety of solutions, and as such it has been studied e.g. in Tomlin *et al.* (1997), Kovsecká *et al.* (1997), Tomlin *et al.* (1998). In this paper, we consider a cooperative scheme which falls within the scope of the theory of teams (cf. e.g. Ho and Chiu (1972), Aicardi *et al.* (1987)). In particular, we consider a scheme in which

- The i–th agent has information on the state and goals of all other agents which are at a distance less than an "alert" radius $R_i > D_i$;
- Each agent plans its flight according to an optimal strategy which consists in minimizing the sum of the time–to–goals of all pertinent aircraft.

Let $S_i(\tau)$ denote the set of indices of aircraft within distance R_i from the i–th aircraft at time τ, i.e. aircraft j such that

$$C_{ij}(\xi_i, \xi_j) = (x_i - x_j)^2 + (y_i - y_j)^2 - R_i^2 \leq 0.$$

The goal of the i–th agent at time τ with information S_i is therefore to minimize

$$\mathbf{J}_{i,S_i}(\tau) = \sum_{j \in S_i} \int_{\tau}^{T_j} dt \qquad (5)$$

under the constraints

$$V_{ij}(\xi_i, \xi_j) > 0, \ \forall j \in S_i. \qquad (6)$$

Obviously, when all R_i are large w.r.t. the dimension of the considered flight area, each agent solves the same problem the centralized controller would solve, and the resulting performance would be equal (albeit with N–fold computational redundancy).

When, during execution of flight maneuvers planned based on a certain information structure $I = (S_1, \ldots, S_N)$, an aircraft i with $i \notin S_j$ gets at distance R_j from aircraft j, the information structure is updated, and optimal paths are replanned according to the new cost and constraints for aircraft j. We assume that structure updates and replanning are done in real time by agents.

The system resulting from the above decentralized ATMS scheme is described by a set of continuous variables $\xi_i, \omega_i, i = 1, \ldots, N$, and a set of variables S_i that take values over discrete sets. To each different information structure I_k there corresponds a working mode for the system, i.e. dynamics (1) driven by controls $\hat{\omega}_{i,k}$ which optimize J_{i,S_i} under constraints $V_{ij} > 0, j \in S_i$. The resulting hybrid system is composed of a finite–state machine and of associated continuous–time dynamic systems, transitions among states being triggered by conditions on the continuous variables.

For instance, in the case with $N = 3$, $R_1 = R_2 = R_3$, there are eight possible states (modes of operation), corresponding to different information structures I_k (see Figure 2).

Remark At every state transition, each agent evaluates in real–time the optimal steering control from the current position to the goal for itself as well as for all other aircraft within its alert radius. This implies that shared information consists of present position and goal coordinates. Only the control policy evaluated by an agent for itself is then executed, as the one calculated for others may ignore part of the information available to them (as e.g. it happens in states I_5, I_6, and I_7 in Figure 2). All optimal policies coincide for large R_i's.

3. SHORTEST MULTI–AGENT COLLISION–FREE PATHS

The algorithm introduced in Bicchi *et al.* (1998), tending to optimize problem (2) with constraints (1), (3) and (4), will be considered for use by the planner, and is succinctly described below. The algorithm is first described with respect to

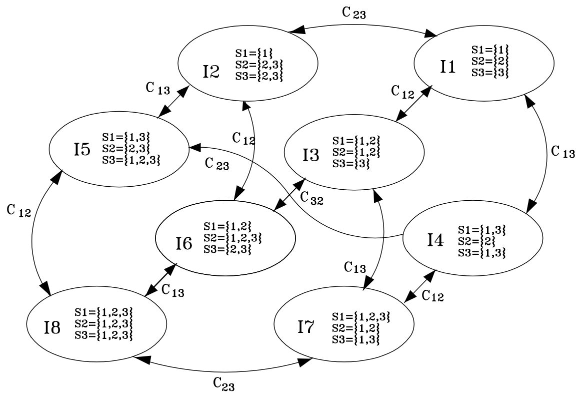

Fig. 2. A decentralized ATMS with three aircraft having equal alert radius. Each node in the graph corresponds to different costs and constraints in the agents' optimal steering problem. Optimizing controllers for such problems cause different continuous time dynamics at each node. Switching between modes is triggered when an airplane enters or exits the alert neighborhood of another ($C_{i,j}$ changes sign).

a centralized implementation, and then adapted to the decentralized setup (5), (6).

The simplified airplane model given by (1), (3) is equivalent to what has become the well–known "Dubins' car" in robot motion planning literature, i.e. a vehicle which only goes forward and has bounded curvature. The solution of the shortest path among two via-points for such system (when a single airplane is considered) has been obtained first by Dubins (Dubins (1957)), who showed that optimal paths are made of concatenations of segments, either circular with minimal radius ("C"–segment), or linear ("S"–segment), and that a shortest path can always be found among 6 candidates of type "CCC" or type "CSC" only. Notice that computation of optimal Dubins' paths is an extremely computationally–efficient procedure. Subsequently, Sussmann and Tang (1991) and Boissonnat *et al.* (1992) reinterpreted this result as an application of Pontryagin's maximum principle. The latter framework is instrumental to developments presented here.

When multiple airplanes are considered, the sum of all lengths of Dubins' paths disregarding (4) is clearly a lower bound to cost (2), and one which is attained if, and only if, the unconstrained Dubins' solutions happen not to collide.

If unconstrained Dubins' solutions collide, then from the theory of optimal control with path constraints (see e.g. Chang (1963), and Bicchi *et al.* (1995) for an application to mobile robot planning) we know that the optimal solution will be comprised of a concatenation of free and constrained arcs, i.e., arcs where $V_{ij} > 0, \forall i, j$ and arcs where $\exists (i,j) : V_{ij} = 0$, respectively. Along constrained arcs, at least two airplanes fly keeping their distance exactly equal to their safety limit.

Along free arcs, however, constraints are not active, hence their Lagrangian multipliers are all zero in the system Hamiltonian, and the problem can be reduced to N decoupled optimization

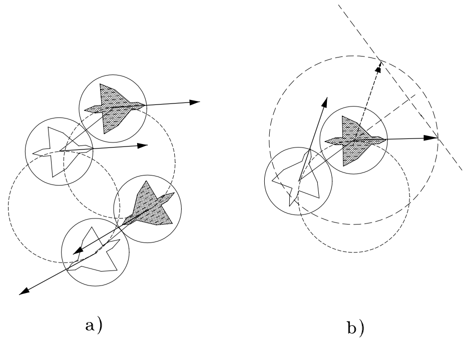

Fig. 3. Possible constrained arcs for two airplanes

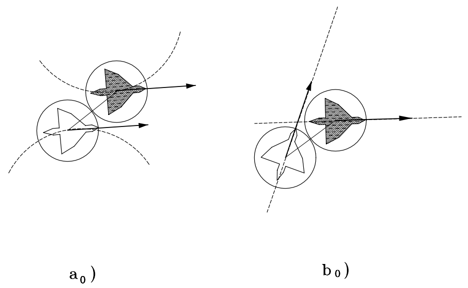

Fig. 4. Zero–length constrained arcs for two airplanes

problem for each airplane, whose solution is again given by paths of Dubins type.

A characterization of constrained arcs can be obtained by differentiating the constraint (4), and implies that only two types of such arcs are possible (type a and b, see Figure 3). Along arcs of type a), the velocities of the two airplanes must be parallel, and the line joining the two airplanes can only translate. Along arcs of type b), the velocities of the aircraft are symmetric with respect to the line joining the aircrafts. Two typical circular trajectories are shown (by dotted circles) in Figure 3 for type a) and b). In both cases, the constrained arc may have zero length, which we will call a_0) and b_0) such as depicted in Figure 4. A constrained arc of type a) or b) must also be a path minimizing solution for the "tandem" system comprised of the two airplanes moving in contact with either parallel or symmetric velocities. In case a), tandem arcs will be again of Dubins type.

Consider now the case of two airplanes flying in shared airspace, such that their unconstrained Dubins' paths collide (see Figure 5). A solution of this problem with a single constrained zero–length arc of either type a_0) or b_0), would be comprised of four Dubins paths $D_{ij}, i = 1, 2; j = 1, 2$, with $D_{i,1}$ joining the initial configuration of agent i with its configuration on the constrained arc, and $D_{i,2}$ joining the latter with the final configuration of the same agent. The set of all such paths for constrained arcs of type a) and b) can be parametrized by a quadruple in $\mathbb{R}^2 \times S^1 \times S^1$ (e.g., position and velocity direction of agent 1, and direction of the line joining the planes). Further, the constraint on the set of paths must be enforced that the constrained arc is hit simultaneously by the two agents, i.e. that $\mathsf{length}(D_{i1}) = \mathsf{length}(D_{j1})$.

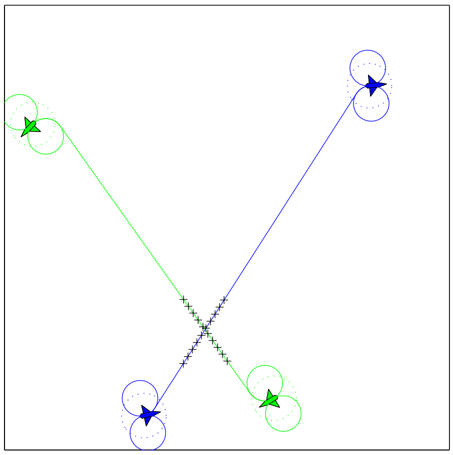

Fig. 5. Configurations of two airplanes for which unconstrained Dubins' solution collide. The dotted circle represents the safety disc; solid circles indicate maximum curvature bounds.

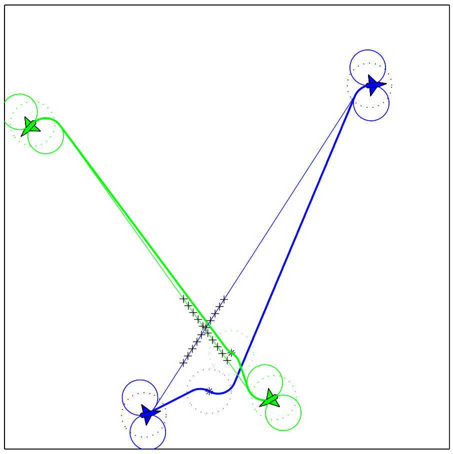

Fig. 6. The centralized OCMP21 solution of problem in 5. The total length is 92.25 (the unconstrained Dubins total length is 88.75). Safety discs contacting at the constrained arc of type b_0) are shown (dotted).

The optimal solution within each case can be obtained by using any of several available numerical constrained optimization routines: computation is sped up considerably by using very efficient algorithms made available for evaluating Dubins' paths (Bui *et al.* (1994)). A solution is guaranteed to exist for both cases. We will refer to the shortest one as to the two–agent, single–constrained zero-length arc, optimal conflict management path (OCMP21, for short). The OCMP21 solution for the example of Figure 5 is reported in Figure 6. Solutions of the two–agent problem with constrained arcs of non–zero length, or with multiple constrained arcs, are also possible in principle. An optimal solution for these cases should be searched in a larger space. However, solutions have to comply with additional requirements on lengths of intermediate free arcs, and seem to be somewhat non–generic. Further theoretical work is currently being devoted to understanding under what conditions non zero–length, and/or multiple constrained arcs may occur in an optimal multi–agent path. In principle, it may even happen that the optimal path is made by concatenating an

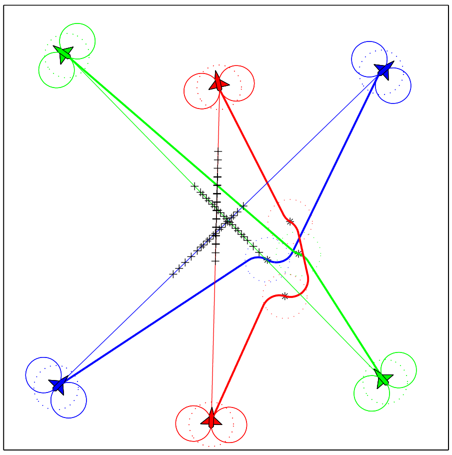

Fig. 7. An OCMP32 solution. The total length is 151.2 (the unconstrained Dubins total length is 140.0). The first constrained arc involves agents 2 and 3 (from left), while the second involves 1 and 2. Boths arcs are of type b_0).

infinite number of free and constrained arcs (cf. the Fuller's phenomenon in optimal control).

In the present version of our planner, we do not search for solutions with multiple and/or non zero–length constrained arcs, trading optimality for efficiency. Another motivation for such limitation is that "acrobatic" flights in tandem configuration do not sem to suit well commercial air traffic.

If three airplanes fly in a shared workspace, their possible conflicts can be managed with the following multilevel policy:

Level 0 Consider the unconstrained Dubins paths of all agents (which may be regarded as single–agent, zero–constrained arc, optimal conflict management paths, or OCMP10). If no collision occurs, the global optimum is achieved, and the algorithm stopped. Otherwise, go to next level;

Level 1 Consider the $\binom{3}{2} = 3$ OCMP21 for each pair of agents. If at least one path is collision free, choose the shortest path and stop. Otherwise, go to next level;

Level 2 Consider the three–agent, double zero–length constrained arc, optimal conflict management problem OCMP32, consisting in searching all 8–dimensional spaces of parameters identifying the first constrained arc of type ξ_1 between agent i and j, and the second constrained arc of type ξ_2 between agent k and ℓ, with ξ_1 and $\xi_2 \in \{a_0, b_0\}$; i, j, k and $\ell \in \{1, 2, 3\}, i \neq j, k \neq \ell$. A solution is guaranteed to exist for all $3^2 \binom{3}{2} \left[\binom{3}{2} - 1\right] = 54$ cases; the shortest solution is OCMP32.

A three airplane conflict management solution at level two (OCMP32) is reported in Figure 7. When the number of airplanes increases, the number of optimization problems to be solved grows combinatorially. However, in practice, it is hardly to be expected that conflicts between more than a few airplanes at a time have to be managed.

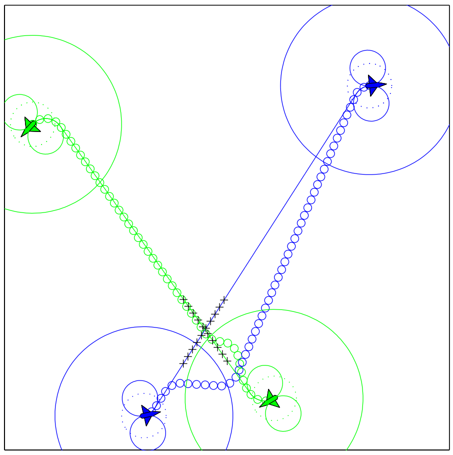

Fig. 8. Decentralized solution of the two–agent conflict management problem (trajectories traced by small circles). Alert discs are drawn in solid lines around the initial and final configurations of agents. The unconstrained Dubins' paths are superimposed for reference.

3.1 *Decentralized implementation*

The algorithm described above can be applied in a decentralized manner by simply having each agent apply its steps taking in consideration only those other agents that are within their alert disc.

The online solutions of the two–agent conflict management problem introduced in Figure 5 and Figure 6, are reported in Figure 8. It can be observed that the two aircraft initially follow their unconstrained Dubins path, until they enter each other's alert zone (this happens roughly at the third step after the start in Figure 8). At this moment, an OCMP21 is obtained by both decentralized planners. Notice that, in this two–agent problem with equal alert radius, the same problem is solved by both, although this does not hold in general. Aircrafts start following their modified paths, which differ from both the unconstarined Dubins paths and the centralized optimal paths of Figure 6. The total length of decentralized solution is 93.85.

4. DECENTRALIZATION: PERFORMANCE AND FAULT TOLERANCE

In order to assess the effects of increasing decentralization in ATMS, we performed a number of simulations whose results are reported below.

In particular, we experimentally compared results obtained by a centralized planner with those achieved by several decentralized planners, with decreasing alert zone radius. The alert zone radius can be regarded as an inverse measure of the degree of centralization for an information structure such as that introduced in the section above.

The first set of simulations concerns performance evaluation. The performance measure, i.e. the total length cumulatively flown by all airplanes, for the problem described in Figure 9 has been calculated for three different values of the safety radius. Results of simulations are reported in

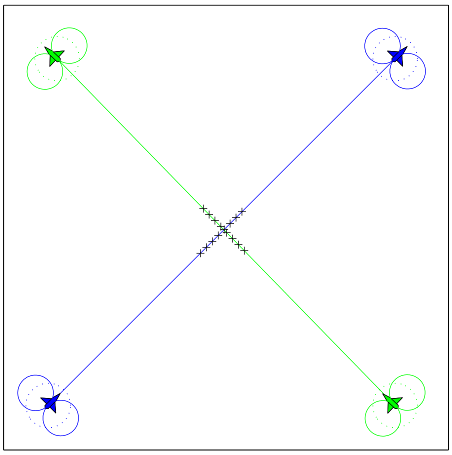

Fig. 9. Air traffic management problem simulated for performance evaluation

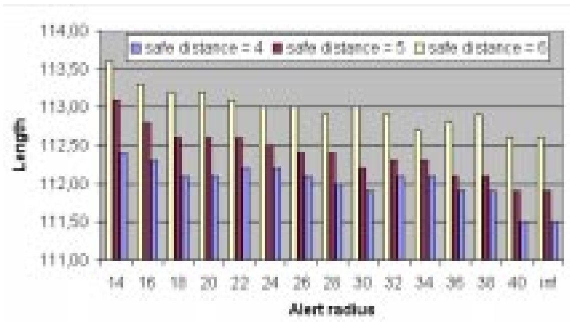

Fig. 10. Total length flown by aircraft at varying the alert zone radius.

Figure 10, and show that the increase of the alert zone radius entails a rather smooth decrease of the total length flown by aircraft. As an effect of the presence of local minima in the numerical optimization process, the length does not decrease monotonically as it should be expected.

To assess fault tolerance, we simulated the same scenario and planning algorithms under degraded control conditions. In particular, some of the controllers were assumed to fail during flight. Controllers affected by failures compute their optimal plans according to the same strategy, but with via–points and information structure randomly perturbed. At the end of such crises, controllers are supposed to access correct data again, and to replan accordingly. The crisis duration is constant for all simulation runs, while they occur at random time. For centralized ATMS, all aircraft are assumed to receive random flight directions from ATC. Under decentralized ATC, other agents are able to maintain their correct operation mode, and replan in real time to try and avoid collisions. As a figure of fault tolerance of ATMS schemes, we consider the number of accidents for 100 crisis situations. Results of simulations, relative to the same initial scenario as in Figure 9, are reported in Figure 11.

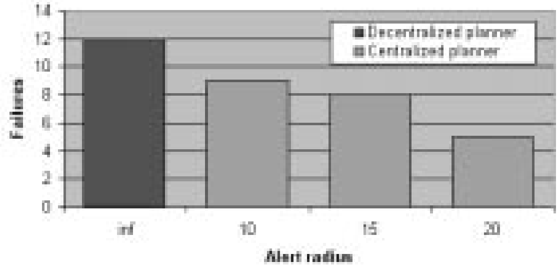

Fig. 11. Fault tolerance to controller failures at varying the alert zone radius. The number of accidents is relative to 100 crisis situations.

References

Aicardi, M., F. Davoli and R. Minciardi (1987). Decentralized optimal control of markov chains with a acommon past information set". *IEEE Trans. on Aut. Contr.* **32**, 1028–1031.

Bicchi, A., A. Marigo, M. Pardini and G. Parlangeli (1998). Shortest collision free paths of multiple Dubins' agents. In preparation.

Bicchi, A., G. Casalino and C. Santilli (1995). Planning shortest bounded–curvature paths for a class of nonholonomic vehicles among obstacles. In: *Proc. Int. Conf. on Robotics and Automation*. IEEE. pp. 1349–1354.

Boissonnat, J.D., A. Cerezo and J. Leblond (1992). Shortest paths of bounded curvature in the plane. In: *Proc. Int. Conf. on Robotics and Automation*. IEEE. pp. 2315–2320.

Bui, X.N., P. Souères, J-D. Boissonnat and J-P. Laumond (1994). Shortest path synthesis for Dubins non–holonomic robots. In: *Proc. Int. Conf. on Robotics and Automation*. IEEE. pp. 2–7.

Chang, S.S.L. (1963). Optimal control in bounded phase space. *Automatica*. 55-67.

Dubins, L. E. (1957). On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangent. *American Journal of Mathematics* **79**, 497–516.

Ho, Y.C. and K.C. Chiu (1972). Team decision theory and information structures in optimal control problems – part 1. *IEEE Trans. on Aut. Contr.* **17**(1), 15–21.

Kovsecká, J., C. Tomlin, G. Pappas and S. Sastry (1997). Generation of conflict resolution maneuvers for air traffic management. In: *Proc. Int. Conf. Intelligent Robots and Systems*. pp. 1598–1603.

Laumond, J. P. (1986). Feasible trajectories for mobile robots with kinematic and environment constraints. In: *Proc. Int. Conf. on Intelligent and Autonomous Systems*. pp. 346–354.

Sussmann, H. J. and G. Tang (1991). Shortest paths for the Reeds–Shepp car: a worked out example of the use of geometric techniques in nonlinear optimal control. Technical Report 91–10. SYCON.

Tomlin, C., G. J. Pappas and S. Sastry (1998). Conflict resolution for air traffic management: A case study in multi-agent hybrid systems. *IEEE Trans. on Aut. Contr.* **43**(4), 509–521.

Tomlin, C., G. Pappas, J. Kosecka, J. Lygeros and S. Sastry (1997). Advanced air traffic automation: a case study in distributed decentralized control. In: *Control Problems in Robotics and Automation* (B. Siciliano and K. Valavanis, Eds.). pp. 261–295. Springer-Verlag.