M-6b-03-6

TOWARDS AUTOMATED CONFLICT RESOLUTION IN AIR TRAFFIC CONTROL [1]

Claire J. Tomlin

Department of Aeronautics and Astronautics
Stanford University, Stanford, CA 94306, USA
tomlin@leland.stanford.edu

Abstract. This paper describes a project currently underway at Stanford University to model and synthesize safe and efficient control schemes for a system of aircraft where each aircraft has some degree of autonomy in planning and tracking its own dynamic trajectory. A key element of this project is an automated conflict resolution scheme (Tomlin *et al.*, 1998*a*), which would be used either on the ground by Air Traffic Control or in the air by the Flight Management System (FMS) of each aircraft. Our conflict resolution scheme is based on sequences of *maneuvers* which are finite sequences of *flight modes* such as heading, altitude and speed changes for each aircraft. These types of maneuvers are routinely used in current Air Traffic Control practice since they are easily understandable by pilots as well as easily implementable by on-board autopilots which regulate the aircraft to heading and speed setpoints. This paper presents the generation and verification of such maneuvers by means of example, describes the mathematical framework and computation engine used, and discusses some of the benefits of such a scheme in a human-centered Air Traffic Control environment. *Copyright © 1999 IFAC*

Keywords. Air traffic control, hybrid systems.

1. INTRODUCTION

Today's Air Traffic Control (ATC) system in the United States is growing at such a rate that will make manual operation of it extremely difficult if not impossible in the near future. Air traffic in the United States alone is expected to grow by 5% annually for the next 15 years (Honeywell, 1996), and rates across the Pacific Rim are expected to increase by more than 15% a year. Even with today's traffic, ground holds and airborne delays in flights due to congestion in the skies have become so common that airlines automatically pad their flight times with built-in delay times. The Federal Aviation Administration (FAA) admits that any significant improvement will require that many of the basic practices of ATC be automated (Perry, 1997). For example, today's airspace has a rigid route structure based on altitude and on ground-based navigational "fixes": current practice of air traffic controllers is to route aircraft along predefined paths connecting fixes, to manage the complexity of route planning for several aircraft at once. The rigid structure puts strict constraints on aircraft trajectories, which could otherwise follow wind-optimal or user preferred routes. Also, while a data link between aircraft and ground is being investigated as a replace-

[1] Research supported by a Stanford University Terman Fellowship, and by NASA (while the author was a graduate student at U. C. Berkeley) under grant NAG 2-1039.

ment for the current voice communication over radio channels between pilot and controller, there is a limit to the amount of information processing that a controller can perform with this data. Studies in (Perry, 1997) indicate that, if there is no change to the structure of ATC, then by the year 2015 there could be a major accident every 7 to 10 days.

In (Tomlin *et al.*, 1997) we propose an architecture for a new air traffic management system along these lines, in which the aircraft's flight management system uses local sensory information from Global Positioning Systems (GPS), Inertial Navigation Systems (INS), and broadcast communication over Automatic Dependent Surveillance-Broadcast (ADS-B) (Radio Technical Commission for Aeronautics, 1997) with other aircraft to resolve local conflicts without requesting clearances from ATC. Key to the success of such a scheme is an automated conflict resolution algorithm, which we propose in (Tomlin *et al.*, 1998*a*). The algorithm has access to the state and intent information of the other aircraft involved in the conflict, through the GPS/INS system linked to the ADS/ADS-B communication link, to information about the aerodynamics and performance characteristics of the other aircraft, and to information about the constraints imposed by the global traffic flow (see Figure 1). Each aircraft is surrounded by two virtual *cylinders*, the *protected zone* and *alert zone*. The radius and height of the protected zone depends on the FAA separation standards (2.5 nautical miles by 1000 feet in Center, 1.5 nautical miles by 1000 feet in TRACON). The size and shape of the alert zone depends on various factors including airspeed, altitude, accuracy of sensing equipment, traffic situation, aircraft performance and average human and system response times. A *conflict* or loss of separation between aircraft occurs when their protected zones overlap. The system of aircraft is defined to be *safe* if the aircraft trajectories are such that their protected zones never overlap. When aircraft enter the alert zone of another aircraft, an alert is issued to ATC as well as to the FMS of each involved aircraft, and depending on the relative configurations (positions, velocities) of the aircraft, a maneuver is generated which resolves the conflict. From a database of flight modes, such as segments of constant heading, of constant bank angle, of constant airspeed, the conflict resolution algorithm synthesizes the *parameters of the maneuver*, such as the proper sequencing of these modes, the numerical values associated to each segment (heading angle, bank angle, airspeed), and the conditions for switching between flight modes. The result is a maneuver, proven to be safe within the limits of the models used, which is a familiar sequence of commands easily executable by the FMSs. The resulting maneuvers may be viewed as

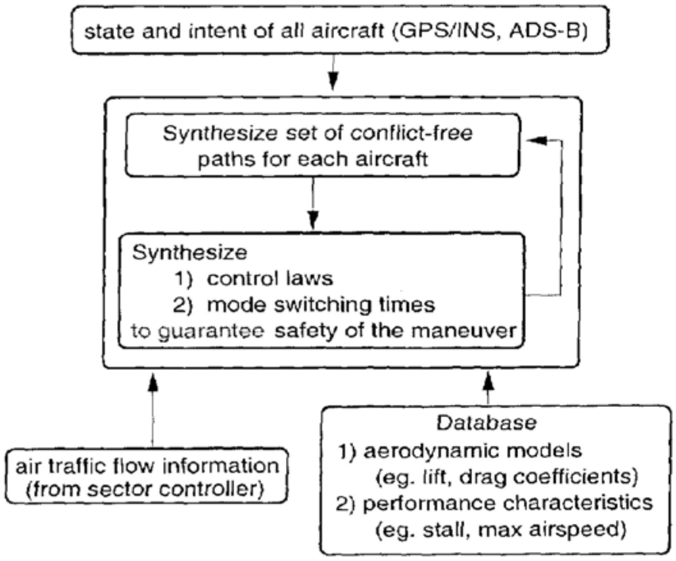

Fig. 1. Conflict Resolution Algorithm.

protocols, or "rules of the road". This type of scheme may be used in the current air traffic control system, as an advisory to air traffic controllers (incorporated in the NASA Ames' Center TRACON Automation System (CTAS) for example); in a *free flight* environment, in which aircraft must coordinate among themselves to resolve conflicts; or in a *partially distributed system* in which free flight could be in effect in certain airspace regions, such as oceanic airspace.

The main thrust of our conflict resolution algorithm is to *verify* that a maneuver successfully resolves the conflict by computing the set of initial conditions for which the maneuver is safe, where safety means that separation is maintained. The flight mode switching occurring in each maneuver is modeled by a finite state automaton with the relative aircraft configuration dynamics residing within each flight mode. A conflict resolution maneuver is therefore modeled by a finite state automaton interacting with a set of control systems, resulting in a *hybrid control system*. The interaction and information exchange of all of the aircraft involved in the maneuver results in a *multi-agent hybrid control system*.

We first present the method by which we generate the database of flight modes to use in the maneuver, we then present our verification methodology for hybrid systems applied to a conflict resolution example, and we conclude with a discussion of the benefits of such an algorithm in a human-centered automated Air Traffic Control environment.

 ISBN: 0 08 043248 4

2. GENERATING CONFLICT RESOLUTION MANEUVERS

Our goal is to generate provably safe conflict resolution maneuvers which are easily understandable by air traffic controllers and pilots, easily implementable in on-board flight management systems, and which may be computing quickly and efficiently for n aircraft. Inspired by results in multiple mobile robot motion planning in which potential field methods are used to construct conflict-free paths for the robots, we developed a simulation (described in (Košecká *et al.*, 1997; Košecká *et al.*, 1998)) in which each aircraft responds to a combination of three fields: an attractive field (to its desired heading), a repulsive field (around each other aircraft), and a vortex field (around each other aircraft to define the direction the aircraft turns when encountering a conflict). We used this to simulate sets of two, three, and four aircraft with various initial conditions: and generated results such as those displayed in Figure 2, which shows two different *overtake* maneuvers, and a *roundabout* maneuver. These resulting maneuvers are intuitively appealing due to their simplicity, yet they are only simulations. The potential field algorithm is not robust to changes in initial conditions, and it is impossible to incorporate complicated dynamic constraints into the algorithm and prove that the aircraft always maintain their required separation.

To provide such a proof, we use the results of the potential field algorithm in the mathematical framework of hybrid system theory. We first abstract the maneuvers into sequences of trajectory segments, or "flight modes", which pilots and autopilots *currently fly*. These are straight lines representing fixed heading and altitude or fixed heading and vertical rate, and arcs of circles representing constant bank angle. Each segment, or mode, has associated to it a set of parameters indicating, for example, length and radius. Figure 3 displays, for example, an abstraction of the roundabout maneuver for two aircraft.

3. SYNTHESIZING CONTROL LAWS, MODE SWITCHES

To verify the safety of the maneuver, and to calculate the parameters for each flight mode and the control laws for each aircraft to ensure safety, we use our hybrid system modeling and controller synthesis method, presented in (Tomlin *et al.*, 1998*b*; Lygeros *et al.*, 1998; Tomlin, 1998).

Consider the maneuver of Figure 3 in which there are three modes of operation: a *cruise* mode in which both aircraft follow a straight path; an *avoid* mode in which both aircraft follow a circular arc path; and a second *cruise* mode in which the aircraft return to the straight path. The protocol of the maneuver is that as soon as the aircraft are within a certain distance of each other, each aircraft turns 90° to its right and follows a half circle. Once the half circle is complete, each aircraft returns to its original heading and continues on its straight path (Figure 3). In each mode, the continuous dynamics may be expressed in terms of the *relative motion* of the two aircraft (described below). In the cruise mode, $\omega_i = 0$ for $i = 1, 2$ and in the avoid mode, $\omega_i = 1$ for $i = 1, 2$. We assume that both aircraft switch modes simultaneously, so that the relative orientation ψ_r is constant. We also assume that the maneuver takes place at a fixed altitude.

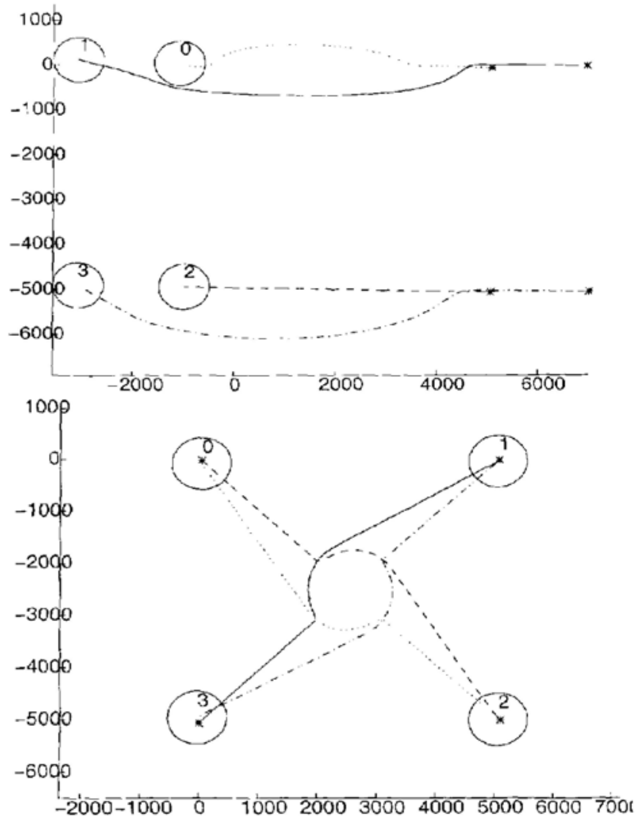

Fig. 2. Generating a database of maneuvers: the upper figure shows two different "overtake" maneuvers, with different repulsive field strengths; the lower figure displays what we have termed the "roundabout" maneuver for four aircraft. Each aircraft is represented by a circle indicating the protected zone in the horizontal plane.

ISBN: 0 08 043248 4

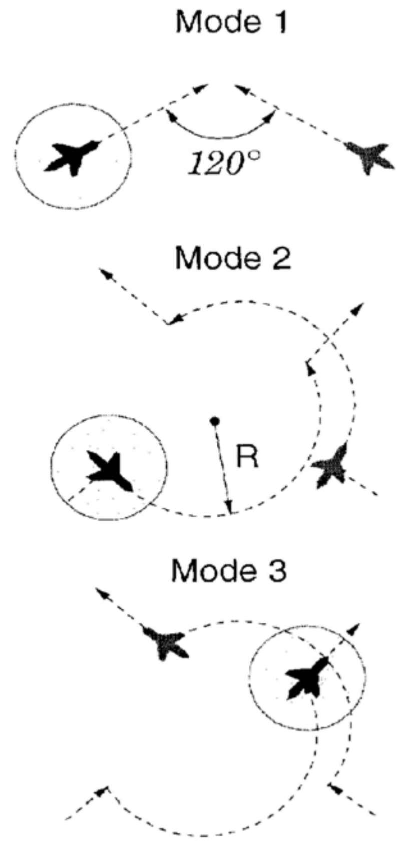

Fig. 3. Two aircraft in three modes of operation: in modes 1 and 3 the aircraft follow a straight course and in mode 2 the aircraft follow a half circle. The initial relative heading (120°) is preserved throughout.

These assumptions simplify our computations and allow us to display the state space in two dimensions, making the results easier to present.

Problem statement: Generate the relative distance between aircraft at which the aircraft may switch safely from mode 1 to mode 2, and the minimum turning radius R in mode 2, to ensure that the 5 nautical mile separation is maintained.

In each mode, the continuous dynamics may be expressed in terms of the *relative motion* of the two aircraft (equivalent to fixing the origin of the relative frame on aircraft 1 and studying the motion of aircraft 2 with respect to aircraft 1):

$$\begin{aligned} \dot{x}_r &= -v_1 + v_2 \cos \psi_r + \omega_1 y_r \\ \dot{y}_r &= v_2 \sin \psi_r - \omega_1 x_r \qquad (1) \\ \dot{\psi}_r &= \omega_2 - \omega_1 \end{aligned}$$

in which $(x_r, y_r, \psi_r) \in \Re^2 \times [-\pi, \pi]$ is the relative position and orientation of aircraft 2 with respect to aircraft 1, and v_i and ω_i are the linear and angular velocities of each aircraft. In the cruise mode $\omega_i = 0$ for $i = 1, 2$ and in the avoid mode $\omega_i = 1$ for $i = 1, 2$. The control input is defined to be the linear velocity of aircraft 1, $u = v_1 \in U$, and the disturbance input as that of aircraft 2, $d = v_2 \in D$, where U and D denote the range of possible linear velocities of each aircraft. Such a situation could arise, for example, in an airborne collision avoidance algorithm in which the flight management system of aircraft 1 wishes to compute the parameters v_1 and α of its avoidance maneuver and can only predict the velocity of aircraft 2 to within some uncertainty.

We model the mode of operation as a discrete state q_i which takes on one of three possible values, $Q = \{q_1, q_2, q_3\}$. The state q_1 corresponds to cruising before the avoid maneuver, q_2 corresponds to the avoid mode and q_3 corresponds to cruising after the avoid maneuver has been completed. There are two discrete actions, which model the transitions between modes. The first (σ_1) corresponds to the initiation of the avoid maneuver and can be controlled by choosing the range at which the aircraft start turning. The second transition (σ_2) corresponds to the completion of the avoid maneuver. This transition is required to take place after the aircraft have completed a half circle: the continuous state space is augmented with a timer $z \in \Re$ to force this transition. Let the continuous state of the system be denoted $x = (x_r, y_r, \psi_r, z)$.

At each transition, both aircraft change heading instantaneously by $\pi/2$ radians. Because the origin of the relative frame is placed on aircraft 1, meaning that aircraft 1 always has a relative position and heading of $(0, 0, 0)^T$ in the relative frame, the transitions rotate the state variables (x_r, y_r) by $\pi/2$ radians. We represent this with the standard rotation matrix $R(\pi/2)$.

Safety is defined in terms of the relative distance between the two aircraft: throughout the maneuver the aircraft must remain at least 5 nautical miles apart. We define the region at which *loss of separation* occurs as a 5-mile-radius cylinder around the origin in the (x_r, y_r, ψ_r, z) space:

$$G = \{q_1, q_2, q_3\} \times \{x \mid x_r^2 + y_r^2 < 5^2\} \qquad (2)$$

The dynamics of the maneuver can be encoded by the automaton of Figure 4.

Using the hybrid system controller synthesis methodology (Tomlin *et al.*, 1998*b*; Lygeros *et al.*, 1998; Tomlin, 1998), we calculate the parameters of the maneuver so

ISBN: 0 08 043248 4

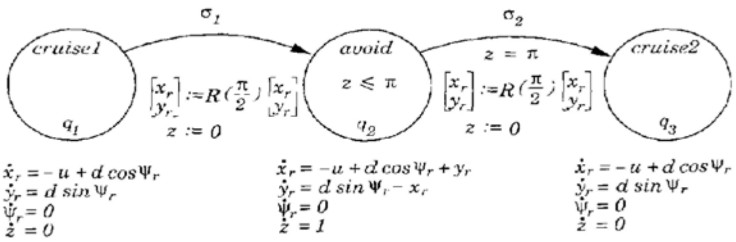

Fig. 4. In q_1 the aircraft follow a straight course, in q_2 the aircraft follow a half circle; in q_3 the aircraft return to a straight course.

that safety is *a priori* guaranteed. In this methodology, we first characterize the subset of the (continuous and discrete) state space which is *unsafe* (in this example, it is the set G), and then calculate the set of all states (denoted W^* in what follows) from which, for all possible continuous and discrete control actions, there are continuous and discrete disturbance actions which drive the system to the unsafe set G. The continuous and discrete control laws are then designed to keep the system out of the set W^*: our algorithm uses a game theoretic methodology to calculate the *best* control action for the *worst* possible disturbance.

The set W^* is computed via a backwards iteration. The first two steps of this iteration are shown in Figures 6 and 7, in which the unsafe set G, shown as a disk (cylinder in q_2) at the origin of the relative frame, is propagated backwards through the continuous and discrete dynamics describing the maneuver. The design of the transition σ_1, indicating when the maneuver should start, is as illustrated in Figure 8(a). σ_1 must be disabled until the relative dynamics in q_1 reach the dashed line as shown, otherwise the aircraft will lose separation with each other either during the maneuver or after the maneuver is complete. At the dashed line, σ_1 is enabled, meaning the transition from q_1 to q_2 may occur at any time. σ_1 remains enabled until the solid line (boundary of W^*), at which point it must be both enabled and forced, otherwise the aircraft lose separation immediately.

Note that there are states (x_r, y_r) which are not rendered safe by the maneuver. Indeed, if the initial state is in the darker shaded region shown in Figure 8(a), then the aircraft are doomed to collide. Figure 8(b) displays the result of increasing the radius of the turn in q_2. Notice that the set W^* (the complement of the shaded region) increases as the turning radius increases. This implies that the maneuver renders a larger subset of the state space safe. Figure 8(b) shows the critical value of the turning radius, for which the maneuver is guaranteed to be safe, provided the conflict is detected early

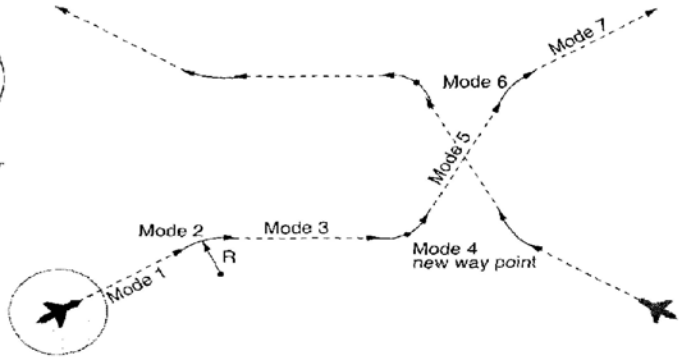

Fig. 5. Two aircraft in seven modes of operation: in modes 1, 3, 5, and 7 the aircraft follow a straight course and in modes 2, 4, and 6 the aircraft follow arcs of circles. Again, the initial relative heading (120°) is preserved throughout.

enough.

We have applied our methodology to more complicated resolution maneuvers, such as the one shown in Figure 5.

4. BENEFITS IN A HUMAN-CENTERED ENVIRONMENT

One of the benefits of our algorithm is that it automatically generates trajectories that *pilots are used to flying*, namely trajectories consisting of straight line segments and arcs of circles. This should also help to maximize the air traffic controllers' degree of comfort with such automated trajectory deviations.

Our conflict resolution algorithm may be programmed into a computer in the ground-based ATC (such as in the NASA Ames' developed Center-TRACON Automation System (CTAS)), or it may be run on board in the aircraft's FMS. One could imagine a short-term solution to automated conflict resolution being the former, and as automation increases, the pilot be given more responsibility for his own trajectory. In either case, having a set of clear, automatically generated protocols which are well understood by both pilot and controller will help to ensure effective controller/pilot coordination as conflict situations arise.

5. CONCLUDING REMARKS

We are currently working on several new directions in this conflict resolution research. One direction is the efficient computation and storage of solutions to hybrid

ISBN: 0 08 043248 4

systems (preliminary work is presented in these Proceedings as (Tomlin *et al.*, 1999)). Another direction is the incorporation of stochastic models into hybrid systems.

6. REFERENCES

Honeywell (1996). Markets Report. Technical Report NASA Contract NAS2-114279. Final Report for AATT Contract.

Košecká, J., C. Tomlin, G. Pappas and S. Sastry (1997). Generation of conflict resolution maneuvers for Air Traffic Management. In: *International Conference on Intelligent Robots and Systems (IROS)*. Grenoble. pp. 1598–1603.

Košecká, J., C. Tomlin, G. Pappas and S. Sastry (1998). 2-1/2D conflict resolution maneuvers for ATMS. In: *Proceedings of the IEEE Conference on Decision and Control*. Tampa, FL. pp. 2650–2655.

Lygeros, J., C. Tomlin and S. Sastry (1998). On controller synthesis for nonlinear hybrid systems. In: *Proceedings of the IEEE Conference on Decision and Control*. Tampa, FL. pp. 2101–2106.

Perry, T. S. (1997). In search of the future of air traffic control. *IEEE Spectrum* **34**(8), 18–35.

Radio Technical Commission for Aeronautics (1997). Minimum aviation system performance standards for Automatic Dependent Surveillance-Broadcast (ADS-B). Technical report. RTCA-186. DRAFT 4.0.

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

Tomlin, C., G. Pappas, J. Košecká, 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.

Tomlin, C., J. Lygeros and S. Sastry (1998*b*). Synthesizing controllers for nonlinear hybrid systems. In: *Hybrid Systems: Computation and Control* (T. Henzinger and S. Sastry, Eds.). pp. 360–373. Lecture Notes in Computer Science 1386. Springer Verlag. New York.

Tomlin, C., J. Lygeros and S. Sastry (1999). Computation of controllers for nonlinear hybrid systems. In: *Proceedings of the International Federation of Automatic Control*. Beijing.

Tomlin, Claire J. (1998). Hybrid Control of Air Traffic Management Systems. PhD thesis. Department of Electrical Engineering, University of California, Berkeley.

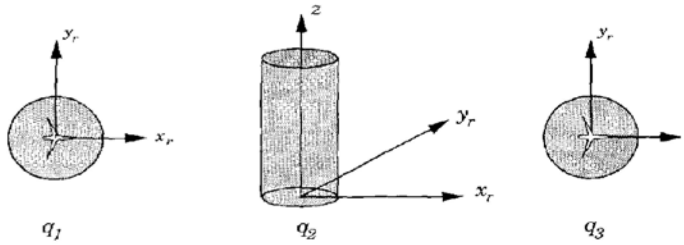

Fig. 6. Calculating the maximal safe operating region for each aircraft, Step 1.

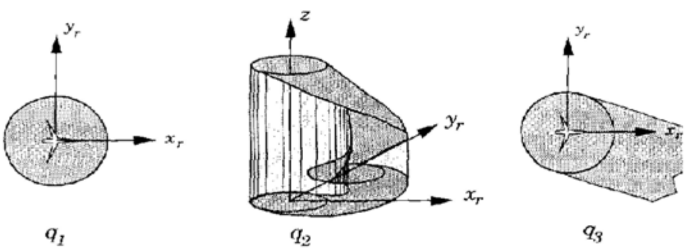

Fig. 7. Calculating the maximal safe operating region for each aircraft, Step 2. The jagged edge in q_3 means that the set extends infinitely.

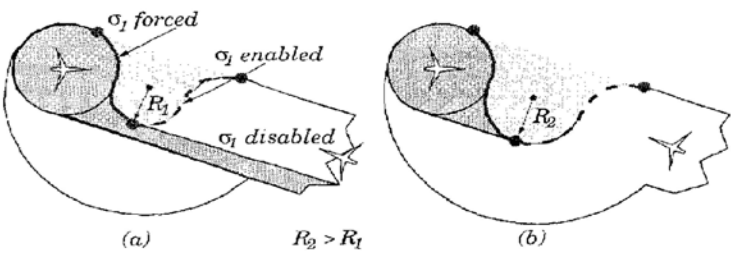

Fig. 8. Determining the Switching Condition σ_1.

ISBN: 0 08 043248 4