Prototype conflict alerting system for free flight

Lee C. Yang
MIT, Cambridge, MA

James K. Kuchar
MIT, Cambridge, MA

AIAA, Aerospace Sciences Meeting & Exhibit, 35th, Reno, NV, Jan. 6-9, 1997

This paper discusses the development of a prototype alerting system for a conceptual free flight environment. The concept assumes that datalink between aircraft is available and that conflicts are primarily resolved on the flight deck. Four alert stages are generated depending on the likelihood of a conflict. If the conflict is not resolved by the flight crews, Air Traffic Control is notified to take over separation authority. The alerting logic is based on probabilistic analysis through modeling of aircraft sensor and trajectory uncertainties. Monte Carlo simulations were used over a range of encounter situations to determine conflict probability. The four alert stages were then defined based on probability of conflict and on the number of avoidance maneuvers available to the flight crew. Preliminary results from numerical evaluations and from a piloted simulator study at NASA Ames Research Center are summarized. (Author)

AIAA-97-0220-

PROTOTYPE CONFLICT ALERTING SYSTEM FOR FREE FLIGHT

Lee C. Yang[*] and James K. Kuchar[†]

Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

This paper discusses the development of a prototype alerting system for a conceptual Free Flight environment. The concept assumes that datalink between aircraft is available and that conflicts are primarily resolved on the flight deck. Four alert stages are generated depending on the likelihood of a conflict. If the conflict is not resolved by the flight crews, Air Traffic Control is notified to take over separation authority. The alerting logic is based on probabilistic analysis through modeling of aircraft sensor and trajectory uncertainties. Monte Carlo simulations were used over a range of encounter situations to determine conflict probability. The four alert stages were then defined based on probability of conflict and on the number of avoidance maneuvers available to the flight crew. Preliminary results from numerical evaluations and from a piloted simulator study at NASA Ames Research Center are summarized.

Introduction

Future air traffic management concepts such as Free Flight have been proposed to provide a means by which traffic flow efficiency can be increased.[1] Under Free Flight, current methods of traffic separation through the use of a rigid airway structure and in-trail spacing would be relaxed. Consequently, aircraft would have more flexibility to follow arbitrary routes in response to changing conditions. To compensate for the loss of airway structure, automated conflict detection and resolution tools would be required to aid pilots and/or ground controllers in ensuring traffic separation.

Because flow efficiency is a driver for Free Flight, it is desirable that conflicts be resolved using minor course, speed, or altitude changes well before emergency avoidance maneuvers are needed. It is also desirable, given the large number of aircraft in the air, that conflict alerts are only generated when necessary. However, the large amount of uncertainty in the Free Flight environment makes it difficult to determine how likely a projected conflict is to occur. The result is a tradeoff between alerting early to provide a large safety margin (and also producing unnecessary alerts) vs. alerting late to reduce unnecessary alerts (but requiring more aggressive avoidance maneuvers).

Traditionally, alerting systems have been designed through an iterative, evolutionary process.[2-4] After defining alerting thresholds, the performance of the system (in terms of the protection it provides and the unnecessary alert rate) is typically evaluated through simulations of traffic encounters. If collisions or excessive unnecessary alerts occur, the alerting thresholds are modified to improve performance; thus, the performance tradeoffs are generally examined *post hoc*.

The tradeoff between safety and unnecessary alerts is well known in signal detection problems and alerting systems.[5,6] One recent approach to view the tradeoff is the System Operating Characteristic (SOC) curve.[7] The SOC curve explicitly shows the expected safety level and unnecessary alert rate as a function of the alert threshold setting. The shape of the SOC curve depends on sensor accuracy, uncertainties in the future flight paths of the aircraft, and human performance. Thus, changes in sensors or avoidance strategies can be evaluated by examining their impact on the shape of the SOC curve.

This paper presents a novel approach to alerting system design in which the performance tradeoffs are directly addressed in order to select alerting thresholds. SOC curves are used to aid in threshold placement, reducing the need for iterative modifications to improve performance. A prototype alerting system was developed using Monte Carlo simulations to assess the probability of a conflict over a range of Free Flight traffic encounters. The logic was then exercised in a set of piloted Free Flight simulation studies at the NASA Ames Research Center in the Fall of 1996. This study examined enroute conflicts and acted as a testbed for the alerting logic presented here.

[*] Graduate Research Assistant
[†] Assistant Professor, Member AIAA

Methodology

The prototype system was developed based on the concept that initial responsibility for traffic separation is shifted to the pilot. The alerting system must, therefore, provide ample warning time so that strategic maneuvers can be examined and coordination between flight crews can be carried out.

To simplify its development, the alerting system described here was designed for one-on-one conflicts during enroute flight. A more complete, operational system would have to be additionally evaluated for its ability to resolve conflicts between more than two aircraft. In this paper, the aircraft with the alerting system is referred to as the *host aircraft*; the other aircraft involved in the conflict is termed the *intruder*. A conflict is defined as a situation in which the intruder enters a Protected Zone around the host aircraft. Based on current separation standards, the Protected Zone was defined to be a cylinder 5 nmi in radius and extending 1,000 ft above and below the host aircraft.

A multi-staged threshold approach was used to provide a series of alerts to indicate trends in conflict hazard. The multi-stage approach allowed the means of implementing the alert to be tailored to the level of threat. Low-probability threats resulted in relatively passive alerts such as changing the color of a traffic symbol. High-probability, urgent threats produced aural warnings to actively inform the pilots of the conflict.

Figure 1 shows a schematic diagram of the multi-stage approach. Three stages (marked **1**, **2**, and **3** in Fig. 1) produced changes in traffic display symbology in the cockpit of the host aircraft. As implemented, the outermost threshold provided a strategic indication of potential threat more than 10 minutes into the future. In the NASA 747-400 simulator, a hollow traffic symbol on the map display changed color when the first threshold was exceeded, and the flight crew could begin to coordinate resolution with the other aircraft. If the encounter continued, an additional stage informed the flight crew of the heightening conflict by filling in the traffic symbol. At the third stage (**3**), an aural "Alert Zone Transgression" message was provided to the flight crew, indicating that they should take action to resolve the conflict. At this point, there was still ample time to coordinate resolution with other aircraft. If the conflict continued without resolution, an Air Traffic Controller (ATC) took over authority for conflict resolution at the Authority Transition (**AT**) zone. The current Traffic Alert and Collision Avoidance System (TCAS) logic was not modified and was kept in the simulation as an independent, final warning system.
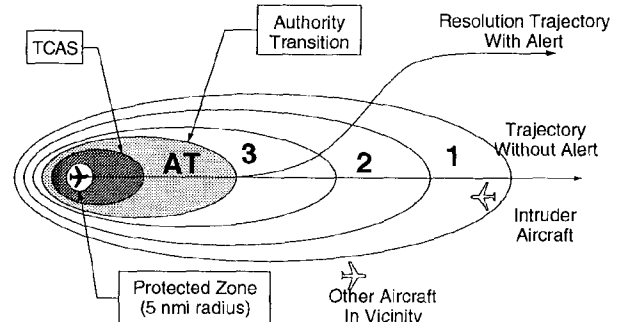

Fig. 1   Multi-Stage Alerting Concept

On the host aircraft, the alerting system operates by obtaining datalinked state information from the intruder. This information includes both the current state of the intruder and also an estimate of the future trajectory of the intruder. Because there are errors in these estimates, the methodology for developing the alerting system is based on a probabilistic analysis of the conflict.

Aircraft Trajectory Model

To determine the probability of conflict, a baseline model of aircraft trajectories was developed. Figure 2 shows a pictorial representation of an aircraft in a Free Flight environment. The modeled parameters include uncertainty in the current position estimate, future along- and cross-track position variability, and the potential for and magnitude of course changes. Given these parameters, the aircraft's future trajectory is represented probabilistically as the dashed region shown in Fig. 2.
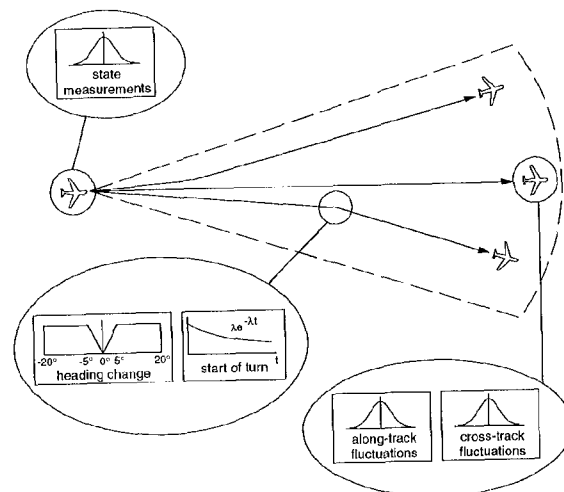

Fig. 2 Probabilistic Trajectory Model

Figure 3 summarizes the uncertainty parameters used in the baseline trajectory model. Uncertainty in current position results from the accuracy of combined Global
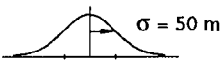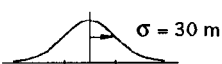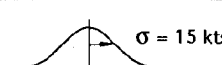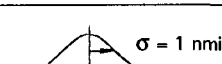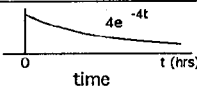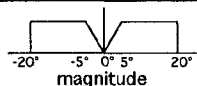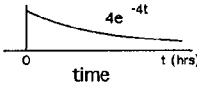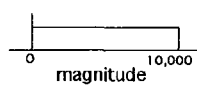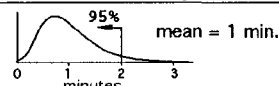
Uncertainty Parameter	Modeled Distribution	
Host Aircraft & Intruder Aircraft	Lateral Position Error	Gaussian $\sigma = 50$ m
	Vertical Position Error	Gaussian $\sigma = 30$ m
	Speed Fluctuation (Along-Track Variability)	Gaussian $\sigma = 15$ kts
	Cross-Track Variability	Gaussian $\sigma = 1$ nmi
Intruder Aircraft Only	Heading Change	$4e^{-4t}$
	Altitude Change	$4e^{-4t}$
Host Aircraft Only	Avoidance Response Latency	Gamma, mean = 1 min.

Fig. 3 Trajectory Model Parameters

Positioning System (GPS) and Inertial Navigation System (INS) estimates and is modeled as a normally-distributed random variable with standard deviation of 50 m laterally and 30 m vertically. Course drift in the future trajectory is modeled as a 15 kt standard deviation speed fluctuation (along-track error) and a 1 nmi standard deviation cross-track error. These tracking error values are based on data obtained empirically from traffic by Paielli and Erzberger.[8]

The host aircraft is assumed to fly a straight trajectory except for the along- and cross-track variations described above. The intruder model includes the additional possibility that the intruder will make a heading change. The likelihood of a heading change is governed by an exponential distribution with a mean rate (λ) of 4 turns per hour. This follows from the assumption that heading changes occur in a Poisson manner. When a heading change is made, its magnitude is modeled probabilistically as well. The intruder is equally likely to make heading changes left or right between 5° and 20°, and is less likely to make turns of less than 5° (see Fig. 3).

Altitude changes are also modeled for the intruder as an exponential distribution with a mean likelihood of 4 occurrences per hour. When a change in altitude occurs, the intruder is equally likely to climb or descend to any altitude within 10,000 ft of its current altitude.

In order to select between alternative conflict resolution options, it is important to evaluate the reduction in conflict probability that can be achieved if the host aircraft maneuvers. Accordingly, a model of host aircraft resolution maneuvers was also developed. The flight crew response latency to a conflict alert is modeled as a probabilistic Gamma distribution with a mean of 1 minute and a variance such that there is a 95% probability that the response occurs within 2 minutes. The relatively long latency is intentionally designed to allow time for coordination with other aircraft and/or ATC. Thus, avoidance maneuvers are assumed to have a large time buffer built in. Once initiated, avoidance maneuvers could include turns, altitude changes, or speed changes.

It must be noted that the values of the parameters used in the trajectory model are estimates at this point and are not expected to be completely representative of Free Flight. Because Free Flight does not currently exist, it is difficult to predict the probabilistic nature of aircraft trajectories. However, such a prediction is necessary in order to estimate the likelihood of conflicts. Even if the values of the parameters are unknown, the impact of changes in the parameters can be evaluated to determine their relative importance. This in turn will help focus future efforts on improving trajectory estimation.

Conflict Analysis

The probability of a conflict, P(C), is defined as the probability that the intruder will enter the host aircraft's Protected Zone given that no alert is issued and that the host aircraft maintains its current course and speed. To calculate P(C), the positions of the two aircraft must be projected into the future to determine the likelihood of a Protected Zone violation. However, an explicit analytical solution incorporating the uncertainty variables listed in Table 1 cannot easily be formulated. Instead, Monte Carlo simulations are used.

Given the locations, speeds, and headings of the host and intruder aircraft, the probability of a conflict can be estimated through Monte Carlo simulation. Each Monte Carlo run consists of stepping through the trajectories of both aircraft over time and determining if a conflict occurs. The trajectories vary randomly with each run according to the distributions from Fig. 3. For instance, in one run the intruder might make a 14° course change 1 minute into the flight; in another run, the intruder may follow a straight-line path for 30 minutes. After a certain number of Monte Carlo runs, a count of the number of Protected Zone intrusions was made. Dividing the number of intrusions by the total number of Monte Carlo runs is then an estimator of P(C).

P(C) was determined through the separate analyses of the horizontal- and vertical-plane situations. A conflict occurs when there are both horizontal and vertical separation violations:

$$P(C) = P(C_{horizontal})\, P(C_{vertical}) \qquad (1)$$

In the horizontal plane, the Monte Carlo simulations were performed over a range of state estimates for the intruder and for several host aircraft avoidance maneuvers. The result of each set of Monte Carlo runs is a plot of the probability of a horizontal conflict for the specific situation (intruder position, heading, speed) and host aircraft trajectory (straight ahead or maneuvering). Initial intruder positions were varied over a grid of dimensions 200 nmi on a side, in 1 nmi increments. Two intruder velocities were used along with nine intruder heading angles. Nine different host aircraft trajectories were examined for each intruder situation. The nine host aircraft trajectories included: straight ahead; left and right turns of 10° and 20°; and 10 and 50 kt speed increases and decreases. The resulting set of probabilities was then stored in a series of look-up tables indexed by position, heading, and speed.

Figure 4 shows a contour plot of the likelihood of horizontal conflict for a specific encounter situation in which the host aircraft is flying at a heading of 360° and

an intruder is currently estimated to be flying at a heading of 330°. The plot shows actual data based on 10,000 Monte Carlo simulations spaced every 1 nmi. In Fig. 4, the host aircraft is in white at the lower left. The plot shows the conflict probabilities for an intruder aircraft in the surrounding airspace relative to the host aircraft. For example, an intruder in the position shown in the figure will cause a horizontal conflict in the future with probability 0.45. If the intruder were farther North or East of the host aircraft, this probability would decrease. As the intruder nears the host aircraft, the probability of a conflict will increase if the intruder remains on a collision course. If the intruder changes heading or speed (or if the host aircraft performs an avoidance maneuver), a different contour plot would represent the probability of a conflict.
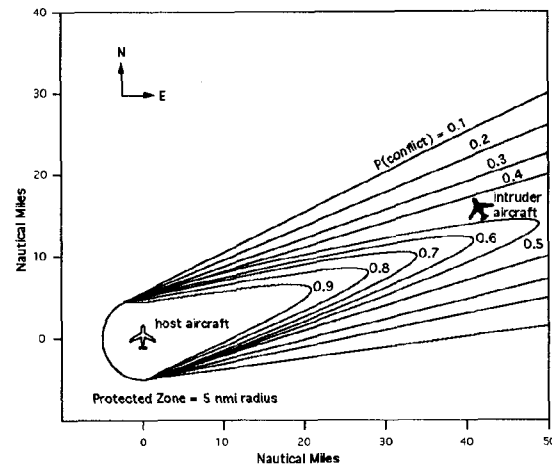
(intruder: 400 kt, 330° heading; host aircraft: 400 kt, 360° heading)

Fig. 4 Example Horizontal Conflict Probability Contours

The vertical conflict probability was obtained by determining the likelihood that the intruder aircraft would follow a vertical path that intersected the Protected Zone. This was performed through an analytical solution of the vertical probability parameters. Different potential vertical maneuvers of the host aircraft were also evaluated by incorporating the host aircraft's vertical speed into the vertical model.

Alerting Design Tradeoffs

The size of the alert zone affects the performance of the alerting system. If the alert zone is too large, an excessive number of unnecessary alerts will be generated. If the zone is too small, there may not be enough space or time in which to maneuver to avoid a conflict. This tradeoff can be examined using two

4

parameters: the probability of Successful Alert (SA) and the probability of Unnecessary Alert (UA).

When an alert is issued, it is defined to be Successful if the Protected Zone is not violated. Thus, the probability of Successful Alert, P(SA), is the probability of a conflict when an avoidance maneuver is performed. P(SA) is therefore a function of time and the specific avoidance maneuver that is performed by the host aircraft:

$$P(SA) = 1 - P(C \mid \text{avoidance maneuver}) \qquad (2)$$

An alert is classified as Unnecessary if the alert was not required to avoid a Protected Zone violation. The probability of Unnecessary Alert, P(UA), is the probability that a conflict would not have occurred had the host aircraft continued on its current course:

$$P(UA) = 1 - P(C \mid \text{no avoidance maneuver}) \qquad (3)$$

To maximize system performance, it is desirable to maximize P(SA) and minimize P(UA). These goals cannot generally be met simultaneously and a tradeoff must be managed.

This tradeoff can be visualized using a System Operating Characteristic (SOC) curve.[7] An example is shown in Figure 5. An SOC curve is a plot of P(SA) for a given avoidance maneuver vs. P(UA). Each point on the SOC curve represents an alerting threshold setting. For example, in Fig. 5, threshold 1 corresponds to a large alert zone: alerts are generated early, resulting in a large value for P(SA) but also a high rate of UAs. As the alert zone size is reduced, the threshold moves along the SOC curve to points 2 and 3. The result is a reduction in UAs but also a reduction in SAs because less time and space are available to perform the avoidance maneuver.

An ideal system would operate in the upper left corner where P(UA) is zero and P(SA) is one: ideally, all alerts are necessary and successful. In reality, SOC curves do not reach the ideal operating point. Instead, the threshold must be placed along the curve based on the tradeoff between UAs and SAs.

The shape of the SOC curve is a function of sensor accuracy, the type of avoidance maneuver, operator response latency, and maneuvering aggressiveness. As sensor accuracy is increased or as response time is reduced, for example, the SOC curve will move closer to the ideal operating point.
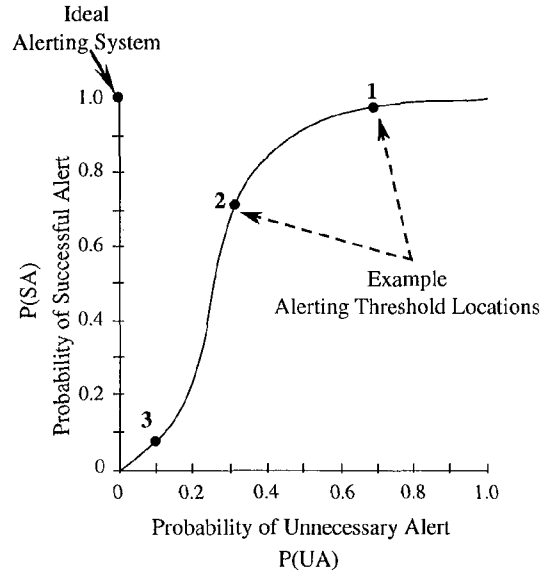

Fig. 5 Example System Operating Characteristic (SOC) curve

An SOC curve that lies along the diagonal from the origin [P(SA) = 0, P(UA) = 0] to the upper right corner [P(SA) = 1, P(UA) = 1] represents a system that is poorly designed. In such a case, an alert is as likely to be Successful as Unnecessary. This means that alerting is just as likely to produce a conflict as not alerting. Thus, the more that the SOC curve moves away from the diagonal, the better the alerting decision.

Because P(SA) depends directly on the choice of avoidance maneuver, a different SOC curve can be constructed for each maneuver option. The most effective avoidance options can then be identified based on the shape of their SOC curves.

Prototype Alerting Logic

As described previously, the prototype system uses four alert stages. The first three stages produce alerts in the cockpit that are intended to aid the flight crew in resolving the conflict before tactical maneuvering is required. At the fourth stage, ATC is notified to issue commands to provide traffic separation. To set the conditions at which these stages are triggered, it is necessary to examine the tradeoffs between P(UA) and P(SA). This requires balancing the likelihood of a conflict against the ability of the host aircraft to avoid a conflict. To do so, five standard conflict resolution maneuvers were considered:

1) Left Heading Change of 30°

2) Right Heading Change of 30°

3) Climb or Descent of 2000 ft/min

5

4) Speed Increase of 50 kts

5) Speed Decrease of 50 kts

These maneuvers serve as benchmarks for estimating the ability of the host aircraft to avoid a conflict. When the intruder is far from the host aircraft, any of these five maneuvers could be used to resolve the conflict. As the intruder nears the host aircraft, some of these maneuvers may no longer provide the required separation between aircraft. The premise behind the alerting logic is that if a sufficient number of these maneuvers are still available to the pilot, the alert can be delayed. When the pilot's options begin to disappear, an alert should be issued.

A maneuver was defined to be *available* to the host aircraft if, by performing the maneuver, the probability of a conflict was reduced to less than 0.05 [i.e., P(SA) > 0.95]. The five maneuver options listed above included the probabilistic response time described earlier (with a mean latency of 1 minute). Thus, when a maneuver was deemed to be not available, safe separation could still be achieved if the pilot reacted more quickly or more aggressively than assumed in the model.

In real time, the logic calculated the number of avoidance maneuvers available, N, to resolve a conflict with the intruder. This was done using the probability contour data stored in look-up tables for each of the five avoidance maneuvers. By comparing N with P(UA), the appropriate alert stage was defined as shown in Table 1.

The leftmost column of Table 1 shows the probability of a conflict if the host aircraft continues along its current trajectory. This assumes that the intruder's trajectory can be represented by the model discussed earlier. The rightmost column shows P(UA), which as discussed earlier is related to P(C) by Equation (3). The other columns indicate the defined alert stages as a function of N. Generally, the more options available to

the pilot, the lower the alert stage.

For example, if P(UA) is 0.35 and there are two avoidance maneuvers available, then the alert stage is 2. If P(UA) drops below 0.3 or if N is reduced to 1, then the alert stage increases to 3. If P(UA) drops below 0.1, then the AT stage is triggered.

Note that because the probability of conflict along different avoidance maneuvers can be estimated, the alerting logic can also be used to determine the magnitude of maneuvering required to resolve a conflict. By interpolating P(SA) between different maneuvering magnitudes, the required action to resolve a conflict with 95% confidence can be determined. For example, if P(SA) for a 10° right turn is 0.93 and P(SA) for a 20° right turn is 0.97, then a 15° turn will result in P(SA) of approximately 0.95. Thus, the probability data can be used both to determine P(SA) when a maneuver is specified, or to determine the magnitude of maneuvering that is required to achieve a specified value of P(SA).

To better understand the underlying design process, the thresholds from Table 1 can be mapped into SOC curves. Figure 6 shows SOC curves for two coaltitude aircraft on a collision course along flight paths at right angles to one another. SOC curves corresponding to each of the five resolution maneuver options are shown in the figure.

When the intruder is far from the host aircraft, the situation maps into the upper right corner of the plot: it is likely that a conflict will not actually occur [P(UA) = 1] and it is likely that any avoidance action would resolve the situation [P(SA) for each of the five avoidance maneuvers is 1]. Data for Fig. 6 were not obtained beyond 200 nmi, so the SOC curves in the figure do not extend all the way to the upper right corner.

As the intruder continues on a collision course, it becomes more clear that a conflict will occur: P(UA)

Table 1   Alert Level Classification

P(C ǀ no maneuver)	Number of Avoidance Maneuvers Available, N				P(UA)
	None	One	Two	Three or More	
0.0 - 0.1	-	-	-	-	0.9 - 1.0
0.1 - 0.2	1	1	-	-	0.8 - 0.9
0.2 - 0.3	1	1	-	-	0.7 - 0.8
0.3 - 0.4	2	1	1	-	0.6 - 0.7
0.4 - 0.5	2	2	1	1	0.5 - 0.6
0.5 - 0.6	3	2	2	1	0.4 - 0.5
0.6 - 0.7	3	3	2	2	0.3 - 0.4
0.7 - 0.8	A T	3	3	2	0.2 - 0.3
0.8 - 0.9	A T	3	3	3	0.1 - 0.2
0.9 - 1.0	A T	A T	A T	A T	0.0 - 0.1

decreases and the situation moves from right to left along the curves. Thus, P(UA) is related to the distance between aircraft and to the time before closest point of approach. As P(UA) decreases, P(SA) also decreases in differing amounts according to the different SOC curves. The effectiveness of a given maneuver depends on how slowly its P(SA) decreases. When a curve's value of P(SA) drops below 0.95, the corresponding avoidance maneuver is no longer available. Thus, as the situation progresses to the left in Figure 6, the different avoidance maneuvers become unavailable in order from speed changes to turns and finally to climb or descent. Thus, the SOC curves show that for this case, vertical maneuvers are the most effective.

The first maneuvers to become unavailable are the speed change maneuvers, at P(UA) of approximately 0.9. This is because large speed changes are generally required to resolve conflicts in the time scales under consideration.

Until P(UA) drops below approximately 0.25, turns and climb/descent avoidance maneuvers will still provide the required separation. At approximately P(UA) = 0.25, however, a 30° left turn maneuver is no longer an option. At approximately P(UA) = 0.2, the 30° right turn is also no longer an option. When P(UA) reaches approximately 0.1, the climb/descend options become unavailable.

At a given value of P(UA), N corresponds to the
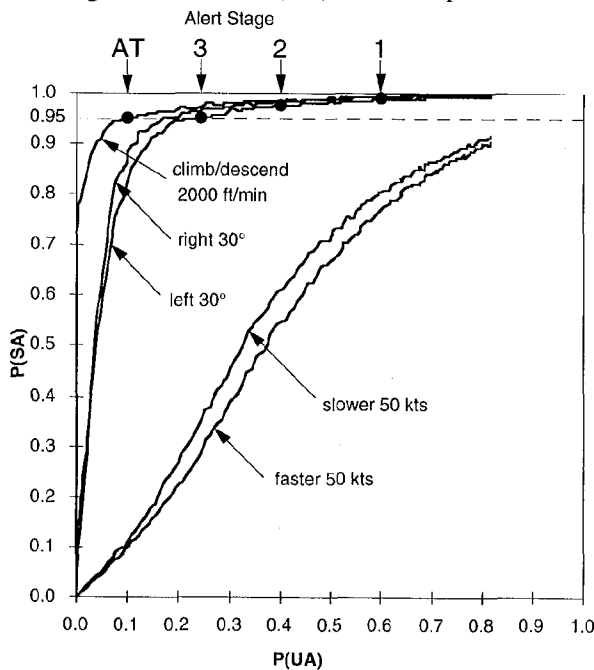
Fig. 6 SOC Curve:
Aircraft on Perpendicular Tracks

number of SOC curves that have values above P(SA) = 0.95. Fig. 6 also shows when the four alert stages are triggered as a function of P(UA). Cross-referencing with Table 1, stage **1** is triggered when N is three or more and P(UA) drops to 0.6. Stage **2** is triggered when P(UA) drops to 0.4 and Stage **3** is triggered when N drops to two. Finally, the **AT** stage is triggered when N drops to zero. Although Figure 6 shows SOC curves for a direct collision between two aircraft on perpendicular flight paths, other geometries produce similar patterns.

The five avoidance maneuvers used here are intended to represent strategic maneuver limits. It should be reiterated that a large response time (mean = 1 min.) is modeled in the avoidance maneuvers (see Fig. 3) and that when N is zero, the host aircraft can still maneuver out of the conflict. A more aggressive, tactical maneuver such as a 45° heading turn or a combined climbing turn may still be available when the five assumed strategic maneuvers are not.

Further examination of the SOC curves show that speed changes make only a limited contribution to the prototype logic. In many cases, a speed change of greater than 50 kts is required for adequate separation with 95% confidence. As can be seen from Fig. 6, the SOC curves for the speed maneuvers deviate only slightly from the diagonal. Thus, it is difficult to provide successful, necessary alerts with speed control alone. Similar difficulties with relying on speed control are mentioned by Krozel, et al. using a much different conflict analysis method based on optimal control theory.[9]
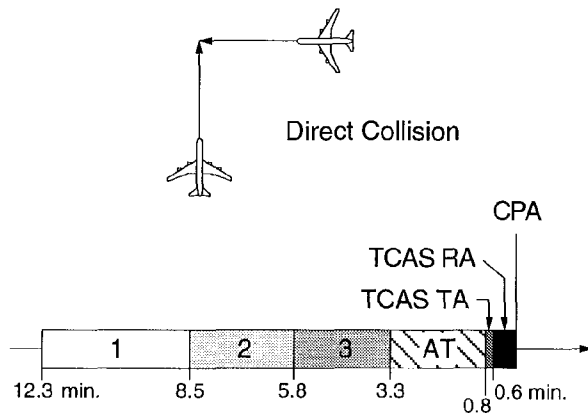
Evaluation

The calculation of the probability of conflict is time consuming due to the large number of required Monte Carlo simulations. Accordingly, the probability contours were stored in look-up tables to be accessed in real time. In operation, the system takes aircraft state data and compares their values against the look-up tables to determine the appropriate alert stage using Table 1. When state values varied between the indices of the look-up tables, the values were linearly interpolated to estimate the probability of conflict.

The alerting logic was evaluated using numerical encounter simulations at MIT and also in a human-in-the-loop simulation study at NASA Ames Research Center. These evaluations were not exhaustive but were used to explore several research issues.

As examples, Figures 7 and 8 show the observed times at which the alert stages were triggered for two different encounter scenarios. Fig. 7 shows the same situation

described by the SOC curves in Fig. 6: two aircraft on a collision course on perpendicular trajectories. Alert stage **1** is triggered 12.3 minutes prior to the time of Closest Point of Approach (CPA). Stages **2** and **3** are triggered at approximately 8.5 and 5.8 minutes to CPA, respectively. ATC is notified to take over authority (at the **AT** stage) at 3.3 minutes to CPA. Finally, TCAS produces a Traffic Advisory (TA) at approximately 45 seconds and a Resolution Advisory (RA) at 35 seconds to CPA.

Fig. 7 Alert Time Line: Direct Collision (90° Crossing Angle)
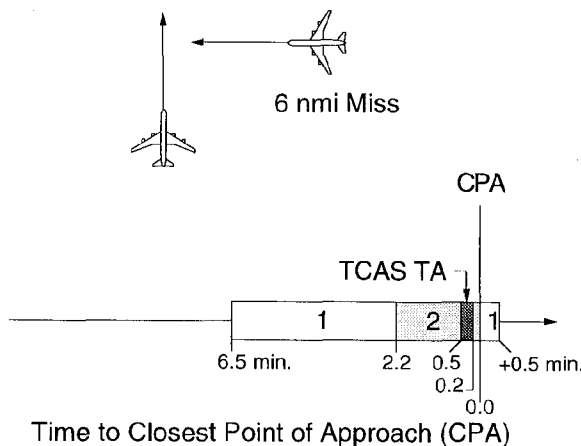
Fig. 8 Alert Time Line: 6 nmi Minimum Separation (90° Crossing Angle)

Figure 8 shows a case in which the two aircraft are not on a direct collision course, but will pass within 6 nmi of one another. Stage **1** is triggered 6.5 minutes before CPA, and stage **2** is triggered 2.2 minutes before CPA. A TCAS TA is also generated at approximately 30

seconds before CPA. When the traffic passes the host aircraft, the alert stages gradually decrease. Thus, the logic increases the alert stage as the potential for a conflict rises, and reduces the alert stage as it becomes less likely that the intruder could turn and cause a conflict.

At the NASA Ames Research Center, the prototype alerting logic was incorporated in a 747-400 simulator as part of a study of pilot decision-making aids for Free Flight. In this study, enroute conflicts were scripted to examine pilot response and to exercise the alerting logic.

In operation, the alerting logic was used to trigger the four stages of alerts discussed previously. Additionally, the probability data were used to determine the magnitude of maneuvering required to resolve conflicts at a specified level of confidence. The pilots in the study were given an interactive tool to explore different maneuvering options. These maneuvers were compared against the probability data to determine if the conflict would be resolved with 95% confidence. The cockpit display then indicated to the pilot whether the proposed maneuver was likely to be successful.

Preliminary results from the NASA study show that the pilots successfully resolved conflicts without ATC guidance in most cases. **AT** alert stages were only observed in scenarios where the intruding aircraft was purposely diverted toward the host aircraft at close proximity. However, a more complete analysis is required to more fully evaluate the alerting logic and to determine the potential impact of airborne conflict resolution on air traffic management.

Conclusion

A prototype conflict alerting system for a Free Flight concept was developed. The system was designed by directly accounting for the probability of a conflict in the alerting logic. A probabilistic model of aircraft trajectories under Free Flight was used in a series of Monte Carlo simulations to obtain probability of conflict contours over a range of traffic encounter situations. Based on these probabilities, alerting thresholds were developed. The alerting system was then implemented in a Free Flight simulation study at NASA Ames Research Center in September, 1996.

The approach taken in developing the alerting logic involved determining the probability of a conflict for several potential avoidance maneuvers. This allowed alerting thresholds to be based on safety rather than an indirect metric such as time to closest point of approach. Additionally, the approach allowed the user to

determine the required magnitude of maneuvering to resolve a conflict with a certain confidence.

A major limitation of the approach used to develop the alerting logic is that a significant amount of preprocessing was required before the alerting thresholds could be defined. Once the thresholds are selected, they only apply to the specific probabilistic model used in the Monte Carlo runs. If a change in the model is required (e.g., to examine the effect of varying sensor accuracy), then the Monte Carlo simulations must be rerun using a new model in order to update the alerting thresholds. A more flexible or real-time means of incorporating the probability of conflict into the alerting thresholds would be valuable in improving the design process.

Look-up tables were chosen to codify the alerting thresholds for this system because of their relative simplicity. However, it may be more effective to use neural networks to map the encounter situation directly to the alerting thresholds without the use of look-up tables. Neural nets can be trained off-line using the pre-calculated data from the Monte Carlo simulations. Once trained, a neural net would represent the alerting thresholds in the real-time system.

Another consideration involves the scope of the conflict. The resolution maneuvers used to develop the alerting logic are based on the immediate problem of avoiding a conflict and do not consider the additional maneuvering required to return to the original flight path. Thus, the logic does not incorporate issues such as increased fuel burn or flight time in the decision to alert. Because the proposed benefits of Free Flight revolve around efficient traffic flow, it will be necessary to incorporate cost-based considerations into the logic in the future. This can be achieved, for example, by weighing avoidance maneuver options by the additional cost or deviation each option would incur.

Finally, centralized traffic management issues have been ignored. Because, as assumed in this Free Flight concept, pilots have initial responsibility for traffic separation, ground controllers could have difficulty when suddenly presented with a conflict that was not resolved by the flight crews. Additional conflict detection and resolution aids must be provided for ground controllers to enable them to return to the traffic management loop and manage traffic once they are alerted to a conflict.

Acknowledgments

References

[1] *Final Report of RTCA Task Force 3: Free Flight Implementation*, RTCA, Washington DC, October 26, 1995.

[2] Williamson, T. and N. A. Spencer, "Development and Operation of the Traffic Alert and Collision Avoidance System (TCAS)", Proceedings of the IEEE, Vol. 77, No. 11, November, 1989.

[3] Miller, C.A., Williamson, T., Walsh, J. A., Nivert, L. J., Anderson, J. L., and A. D. Zeitlin. "Initiatives to improve TCAS-ATC compatibility", Journal of ATC, July-September, 1994.

[4] Radio Technical Committee on Aeronautics (RTCA), *Minimum Performance Specifications for TCAS Airborne Equipment*, Document No. RTCA/DO-185. Washington, D.C., September, 1983.

[5] Barkat, M., *Signal Detection and Estimation*, Artech House, Boston, 1991.

[6] Sheridan, T. B., *Telerobotics, Automation, and Human Supervisory Control*, MIT Press, Cambridge, MA, 1992.

[7] Kuchar, J. K. "Methodology for Alerting-System Performance Evaluation", AIAA Journal of Guidance, Control, and Dynamics, Vol. 19, No. 2, March-April, 1996.

[8] Paielli, R. and H. Erzberger, "Conflict Probability Estimation for Free Flight", Paper accepted for publication in AIAA Journal of Guidance, Control, and Dynamics, 1996.

[9] Krozel, J., Mueller, T., and Hunter, G., "Free Flight Conflict Detection And Resolution Analysis", Paper Presented at the 1996 AIAA Guidance, Navigation, and Control Conference, San Diego, CA, 1996.