A98-37082

AIAA-98-4237

USING INTENT INFORMATION IN PROBABILISTIC CONFLICT ANALYSIS

Lee C. Yang* and James K. Kuchar†

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

Abstract

A method to incorporate aircraft flight path intent information into a conflict detection and resolution system is described. The approach uses a set of probability density functions that describe potential trajectory errors such as cross-track, along-track, or course change, in a series of Monte Carlo simulations. The simulations are used to estimate the probability of conflict in traffic encounters. Intent information is readily included in the model, and can consist of a series of waypoints, heading or track holds, target altitudes, or maneuvering limitations from free flight concepts. The method also allows for direct incorporation of the confidence with which an intended path will be followed. An efficient modeling method is used that enables the Monte Carlo simulations to be run in real time, suggesting that such an approach could be used in conflict detection systems. Several example traffic encounters are discussed using conflict probability maps that show regions in which a conflict will likely occur. The potential benefit of access to intent information in a vertical conflict example is also shown.

Introduction

In order to efficiently detect and resolve conflicts between aircraft, estimates of both the current and future states of the traffic environment are required. The current states, obtained through sensors such as radar or via datalinked position information, are used to indicate whether a conflict currently exists and to provide a starting point for projecting future trajectories. The future states of the traffic must be estimated using some form of dynamic model that propagates the states forward in time. Predicted conflicts between aircraft can then be identified in time for resolution actions to be taken to maintain safe and efficient traffic flow.

The dynamic model used to propagate states generally requires some assumptions regarding the future intentions of each aircraft. Often, it is assumed that aircraft will fly on their current headings at their current speeds and altitude rates. Such a model is acceptable if aircraft indeed tend to fly in straight lines. However, if an aircraft will be maneuvering (e.g., changing heading, speed, or altitude), the straight-line projection of its states becomes inaccurate. If additional intent information is available (e.g., that the aircraft will be leveling off at a certain altitude), a more accurate future trajectory may be projected. However, this intent information can be unreliable (e.g., a pilot may level off before or descend through an intended altitude).

An additional source of uncertainty is the accuracy with which an intended path will be flown. For example, even if it is known that an aircraft is traveling toward a particular waypoint, winds and lateral tracking accuracy will affect the flight path of an aircraft, making predictions of its future position increasingly uncertain. Providing additional intent information, such as a required time of arrival at the waypoint, may help reduce some of this uncertainty.

In order to develop effective systems that produce correct detections with few nuisance alerts, it is necessary to consider these uncertainties. Accordingly, several recent research efforts have focused on estimating the probability of conflict.[1-4] These probabilistic methodologies allow for the direct examination of tradeoffs between nuisance alarms and missed detections (alerting too late to avoid a conflict) and can also be used to develop requirements on sensor accuracy. To date, however, these efforts have primarily investigated cases in which intent information is not available; aircraft have been generally assumed to fly along straight paths.

* Graduate Research Assistant, Student Member AIAA
† Assistant Professor, Member AIAA

This paper extends earlier work, and describes a method to compute the probability of conflict when intent information and trajectory uncertainty are both included. In this manner, a conflict detection system can be developed that can tailor the trajectory model to the type of intent information that is available (e.g., that an aircraft is flying to a waypoint or is turning to a specific heading). After outlining the major design issues and presenting the approach, several case study examples are discussed for both horizontal and vertical intent information.

Previous Modeling Efforts

In order to detect conflicts between aircraft, it is necessary to project the future positions of the aircraft over time. To do so, an appropriate trajectory model is required to propagate the aircraft's current states. Approaches to conflict analysis generally rely on one of two propagation methods, termed *nominal* and *worst case*.[1] In the nominal trajectory method, future aircraft positions are assumed to follow a single specific path (usually along a straight line with the current estimated velocity vector). In the worst-case approach, every possible path is considered, limited only by the aircraft's aerodynamic capabilities or other imposed constraints. In between these two methods, however, is a middle ground where the likelihood of various trajectories are weighed by their probability of occurrence. Both the nominal and worst-case approaches provide either a hit (1) or a miss (0) in their evaluation of a conflict. In the probabilistic approach, a weighted value between 0 and 1 is determined, corresponding to the probability of a conflict, P_C. To some degree, both the nominal and worst-case approaches can be considered subsets or special cases of the probabilistic approach.

The nature of the trajectory of an aircraft is inherently uncertain to some degree, and has been studied previously.[2-5] The uncertainty in the future trajectory depends on factors such as wind, autopilot tracking accuracy, or the pilot making course changes. To be able to estimate P_C, it is necessary to obtain an appropriate model of the probabilistic aircraft trajectory.

If the trajectory model is relatively simple, then it may be possible to derive an explicit closed-form solution to P_C. For instance, Paielli and Erzberger developed a viable analytical solution for a pair of aircraft maintaining a straight-ahead course using Gaussian uncertainties in along- and cross-track error.[2] As the number and complexity of the uncertainties are increased in the trajectory model, however, it becomes increasingly difficult to obtain an explicit analytical solution. This is especially true when complex trajectories with heading and speed changes are used.

In previous work, the authors developed a prototype conflict alerting logic based on probability of conflict principles using pre-processed Monte Carlo simulations.[3] Monte Carlo runs were performed on a large set of possible conflict geometries between a pair of aircraft and stored in look-up tables. The tables of P_C were then referenced by a prototype alerting system during real-time, piloted simulator studies at NASA Ames Research Center.[3,6] Although successful, the approach was not able to readily incorporate more complex intent information because the probability values were stored in look-up tables based on a single trajectory model.

Since it is impractical, if not impossible, to predetermine all possible conflict scenarios in advance, especially with multiple aircraft involved in 3-D space, it is desirable to perform the probability computations in real-time. The added flexibility from such an approach is apparent if intent information is to be incorporated directly into the conflict detection algorithm of the alerting logic. Any new information that becomes available (e.g., datalinked information that an aircraft is flying to a waypoint) can be used to update the projected path. As intent changes, so should the trajectory model, which in turn determines P_C. The predicted paths, the conflict probabilities, and ultimately the decision to alert would adjust dynamically to changing conditions and available information on future intent.

To meet this need, a more efficient method of performing the Monte Carlo simulations was developed and is presented in this paper. The approach allows for direct incorporation of intent information (waypoints, headings, speed, or altitude targets) as well as arbitrary (non-Gaussian) probability density functions. Computation of P_C with the model is possible in real time, suggesting that its use in conflict detection systems may be possible.

Before describing the modeling approach and several example case studies, a brief overview of the potential benefit of intent information is appropriate.

Value of Intent Information

Intent information will alter the shape and position of the predicted aircraft trajectory. Knowledge of planned changes in heading, altitude, or speed can be used to better model this trajectory. For example, if a waypoint is programmed into the Flight Management System (FMS), one can expect the aircraft to follow a relatively straight course with some possible errors from wind and speed variations. Without this waypoint information, there is more uncertainty as to where the aircraft will go, requiring a more conservative, broader trajectory model. Restrictions placed on maneuvering by pre-defined rules-of-the-road, Air Traffic Control (ATC) clearances, or a specified required time of arrival could also limit the range of possible trajectories one could expect.

Because the ability to predict the future position of an aircraft is necessary to determine a conflict, the accuracy of the trajectory model essentially affects the performance of any alerting system. Including information with regard to intent into the trajectory model can lead to a better prediction of the hazard situation. However, placing complete reliance on presumed intent information can also lead to false predictions of conflict and false alarms, especially when trying to predict conflicts more than a few minutes in the future.

One example of the difficulties in correctly detecting conflicts in the presence of intent information is a recent case involving the Traffic Alert and Collision Avoidance System (TCAS).[7] Earlier versions of TCAS were prone to generating false warnings in response to certain situations where one aircraft was descending rapidly toward another. If the descending aircraft leveled off above safely the threatened aircraft, a false warning might still be generated because TCAS could not predict the level-off maneuver. This problem was largely solved by modifying the TCAS alerting thresholds. However, an alternate solution could be developed if the target altitude of the descending aircraft were available to TCAS. In such a case, the intent information could be used to inhibit the normal TCAS alert. If the aircraft descends through the target altitude, however, there might not be enough time remaining to avoid a collision.

Although TCAS is a time-critical collision warning system, similar issues involving intent apply to more strategic conflict detection problems as well. In fact, intent information becomes more important as the time horizon increases, because straight-line projections of aircraft position become increasingly inaccurate.

Monte Carlo Simulation Method

In probability conflict analysis, the goal is to determine the likelihood that one or more intruder aircraft will violate the protected zone of a host aircraft of interest, thus determining the level of threat to the host. In this paper, the protected zone was chosen based on current separation standards and defined to be a cylinder 5 nmi in radius and extending 1,000 ft above and below the host aircraft.

To calculate the probability of conflict, P_C, the positions of the involved aircraft must be projected into the future. Fig. 1 shows an example of the predicted position distributions for a single aircraft traveling with a nominal speed of 400 kts. Intent information of a 45° turn at a waypoint 100 nmi ahead was also assumed to be known. At each time shown in the figure, the aircraft is predicted to lie within the corresponding region with probability 0.9999.

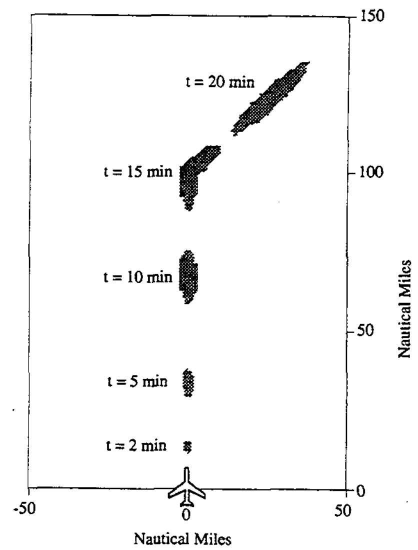


Fig. 1
Example Projected Position Uncertainty

Fig. 1 was generated from point-mass Monte Carlo simulations using a trajectory model that included along-track speed fluctuations (Gaussian, with standard deviation σ = 15 kts) and cross-track variability (Gaussian, σ = 1 nmi). These parameter values are based on prior trajectory modeling efforts.[2] At t = 0 min, the position of the aircraft is known exactly since no sensor errors are included in this example. As shown, the predicted position error grows both along-track and cross-track in time, but generally follows the intended path.

If for some reason there is uncertainty that the aircraft will make the intended turn at the waypoint, an additional confidence probability can be included. In such a case, the position distribution would split into two separate regions: one for the case in which the turn is followed, and one for the case in which the turn is not followed. A situation where this type of modeling might prove especially useful is in vertical conflict analysis where an intruding aircraft may not be entirely trusted to level-off at the expected altitude.

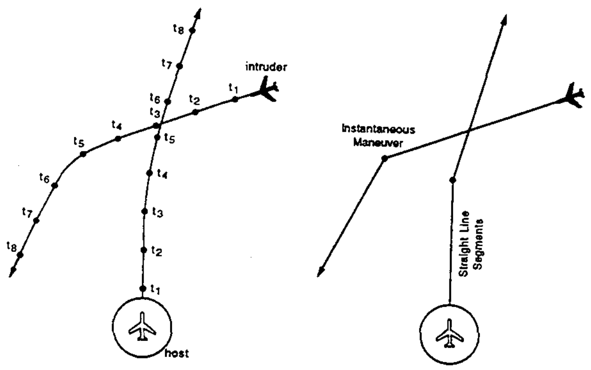
The probability of a conflict is then obtained by extrapolating each aircraft's position in a similar manner. Given the locations, speeds, and headings of the aircraft, each Monte Carlo run consists of stepping through the trajectories over time and determining whether separation minimums of the protected zone are violated. The trajectories vary randomly with each run according to the uncertainty distributions chosen to define the trajectory model. After a certain number, N, of Monte Carlo runs, a count of the number of protected zone intrusions, x, is made. Dividing x by N is then an estimate of P_C.

Trajectory Modeling

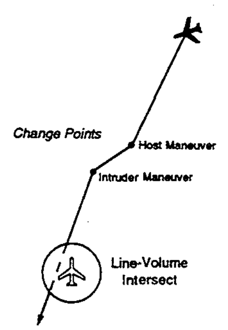
When propagating the aircraft into the future, one possible approach (used in Ref. 3) is to check for a protected zone violation at the end of incremental time steps as depicted in Fig. 2a. For each time step, dt, the position of each aircraft is calculated and horizontal and vertical ranges are checked against minimum separation criteria. This method requires that the time steps be small enough so that intrusions which might occur in-between each end point are not missed. However, reducing dt greatly increases the computational time. This requires either shortening the maximum projection time into the future or waiting longer to obtain a value for P_C.

A more efficient approach (used in the case studies presented in this paper) can be devised if the trajectories are modeled as a series of line segments. Over many runs of the Monte Carlo simulations, the difference in approaches is likely to be indiscriminate in the net result. This simplified approach is represented in Fig. 2b, where change points approximate course changes in the trajectories previously depicted in Fig. 2a. In between change points, the velocity vector of each aircraft is constant. Separate change points are generated at every point at which intended heading, altitude rate, or speed changes occur.

It is also more straightforward to work in the relative frame of the host aircraft, as shown in Fig. 2c. The protected zone is placed around the origin representing the position of the host, and the relative trajectory of the intruder aircraft is propagated. Because of the assumptions made, the trajectories are comprised of straight line segments with each endpoint corresponding to a course or speed change by *either* the host or the intruder aircraft. The task is then to determine if any individual line segment passes through the protected zone around the host aircraft at the origin. Using geometry, an analytic solution can be derived to check for intersections between the equation of the lines and a 3-D volume (the protected zone cylinder). Not only does this method detect conflicts along the entire path rather than only at discrete points, but the computational time is decreased by orders of magnitude compared to the incremental-time approach (Fig. 2a). Also, the method is insensitive to the time scale of the projection; it only depends on the number of course or speed changes that occur between both aircraft.



(a) Incremental Time Steps (b) Straight Line Approximation



(c) Line Intersection in Relative Frame

Fig. 2 Aircraft Trajectory Propagation

A similar approach is used in modeling the vertical plane. Knowledge of vertical speed and/or top or bottom of climb or descent is used to set change points in three dimensions.

In some instances, it may be desirable to model the course transitions more accurately. This might be the case if an intruding aircraft is relatively close and the crucial conflict point is somewhere near the region of the course change. Take for instance the example shown in Fig. 3 where the host aircraft is making a turn toward a target waypoint. A trajectory modeled with an instantaneous turn (dashed line) may be overly simplistic since the actual turn radius can be on the order of 10 nmi or so, depending on the speed and bank angle. This could lead to missing the detection of the conflict with the intruder aircraft shown in the picture. Thus, it is more accurate to include additional line segments to better represent the actual change in heading over time. Fig. 3 shows one additional change point, A, added to better approximate the path of the host aircraft during the heading transition. The turn radius (R) can be estimated from the intended bank angle (ϕ) and speed (v) using

$$R = \frac{v^2}{g|\tan\phi|} \qquad (1)$$

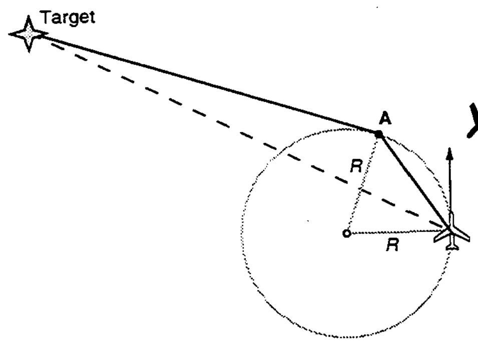


Fig. 3 Heading Change Model

The center of the turn circle can be approximated to be in the direction perpendicular to the current aircraft heading and at a distance R away. Using geometry, the position of point A can then be determined as a tangent line from the turn circle to the target waypoint position. The result is a two-segment path (shown as a solid line in Fig. 3).

If more accuracy is desired, the turn can be further sub-divided into additional straight line segments, though at a cost to computational time.

Trajectory Uncertainties

In each Monte Carlo run, a number of parameters are considered to be random variables that can change with each iteration. The values described below represent one possible model of uncertainties. Alternate probability density functions can be easily incorporated; determination of appropriate values for many of the parameter values is an area for future research.

The baseline trajectory model assumes current position accuracy to be that from combined Global Positioning System (GPS) and inertial navigation system estimates, and is modeled as a normally-distributed random variable with standard deviation of 50 m laterally and 30 m vertically. Along- and cross-track error values are based in part on data from observations of current air traffic by Paielli and Erzberger and were used in previous studies.[2,3] Along-track error is modeled as a 15 kt standard deviation speed uncertainty. Cross-track error grows from its initial standard deviation of 50 m at the aircraft's current position, to a steady-state error with a standard deviation of 1 nmi. This growth is assumed to occur with a lateral deviation error with a standard deviation of 1°. Thus, the steady-state 1 nmi cross-track error is achieved approximately 57 nmi ahead of the aircraft's current position.

Uncertainties in the vertical plane can also be included in the model. Several modeling methods have been examined which are discussed later in this paper.

Implementation

The Monte Carlo simulation model was implemented in C-code on a Silicon Graphics workstation. A modular framework was set up such that relevant parameters can be sent to the Monte Carlo simulation engine, which then outputs the computed probability of conflict. As shown in Fig. 4, inputs include: size of the protected zone, current state information, intent information, and uncertainty information. These inputs may be specified by a researcher interested in specific encounter situations or by data from a real-time flight simulation or flight test. The simulation engine runs in near-real-time, outputting the probability of conflict on the order of 1 sec after inputs are received. This enables conflict detection to be responsive to changes in intent information, such as modifications to a flight plan or changes in autopilot flight mode.

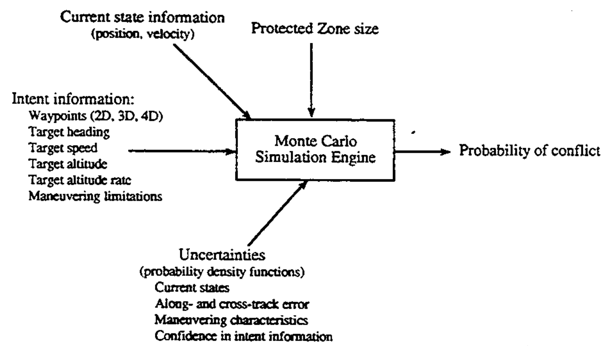


Fig. 4 Monte Carlo Simulation Structure

Current state information includes the position and velocity vector of aircraft in the surveillance region around the host aircraft. Intent information may be specified in a number of forms: multiple 2D (horizontal position), 3D (horizontal and vertical), or 4D (position and time) waypoints; target heading, speed, altitude, and vertical speed; bottom and top of climb and descent; and maneuvering limitations such as maximum permitted turn magnitude and average frequency of maneuvering. Uncertainties in all of these parameters may also be specified, including confidence that an intended path will be followed.

Upon receiving the inputs, the Monte Carlo simulation engine then computes the probability of conflict using the method outlined above. Currently, trajectories are extrapolated using a point-mass model of a jet-transport aircraft; other aircraft types may require customized dynamic models.

Computational Accuracy

The problem posed in calculating P_C is basically that of estimating a value of proportion. The number of conflicts, x, divided by the total number of runs, N, provides an unbiased estimator of P_C with variance σ^2 given by

$$\sigma^2 = \frac{P_C(1-P_C)}{N} \tag{2}$$

For the examples in this paper, $N = 10{,}000$ was used as a compromise between speed and accuracy. Using a Silicon Graphics Indigo Elan 4000 workstation, computational time for 10,000 iterations was on the order of 1 second for a pair of aircraft, providing a 3σ error in P_C of at most 0.015.

In prior simulation studies, it was found that acceptable conflict detection performance was possible by alerting to conflicts which had computed probabilities of conflict between 0.1 and 0.9. Thus, accuracy to ± 0.015 is likely much greater than is required in practice. Accordingly, in a simulation study being performed at NASA Ames in the summer of 1998, the Monte Carlo technique was implemented using $N = 1{,}000$, resulting in a worst-case uncertainty of $3\sigma = 0.05$. This level of accuracy is acceptable for distinguishing a conflict from a non-conflict, and allows for significantly faster computational times.

One additional feature of Monte Carlo simulation is that an estimate of P_C is continuously available. The longer that the simulation runs, the more accurate this estimate becomes. Additionally, a direct estimate of the uncertainty in P_C is also available as discussed above. A different mode of operating, therefore, could be to specify the maximum computational time that is permitted, and to use whatever the estimate of P_C happens to be at the end of that computation time. Alternatively, one could specify maximum error levels in the estimate of P_C, and continue to refine the estimate until that error constraint is satisfied.

Horizontal Conflict Examples

Given the relative speed and heading between aircraft, a conflict probability map can be constructed to display the locations where an aircraft currently must be in order to result in a conflict at some later time. As a simple example, assume two aircraft (host and intruder) are co-altitude and both flying with a velocity of 400 kts in opposite directions. If the intent of each aircraft is known, then potential conflict situations can be predicted. Assume that both aircraft have declared that they will maintain their current speed, heading, and altitude. This might be inferred, for example, through datalink of autopilot mode control settings.

The potential conflict map as obtained through the Monte Carlo simulation is shown in Fig. 5. As a reminder, in this example the intruder aircraft is traveling in the opposite direction as the host. The chart is shown relative to the host aircraft (located at the

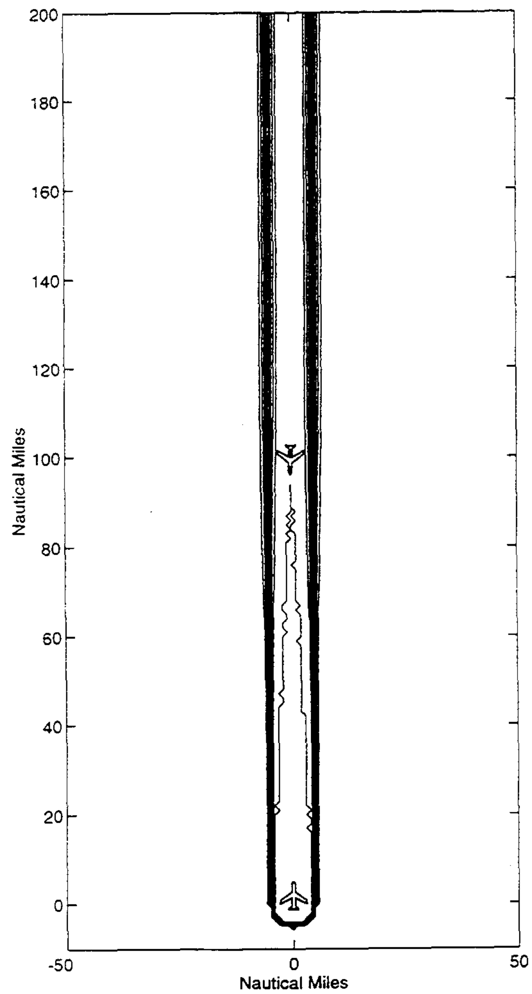


Fig. 5
Case 1: Intruder and Host Maintain Course

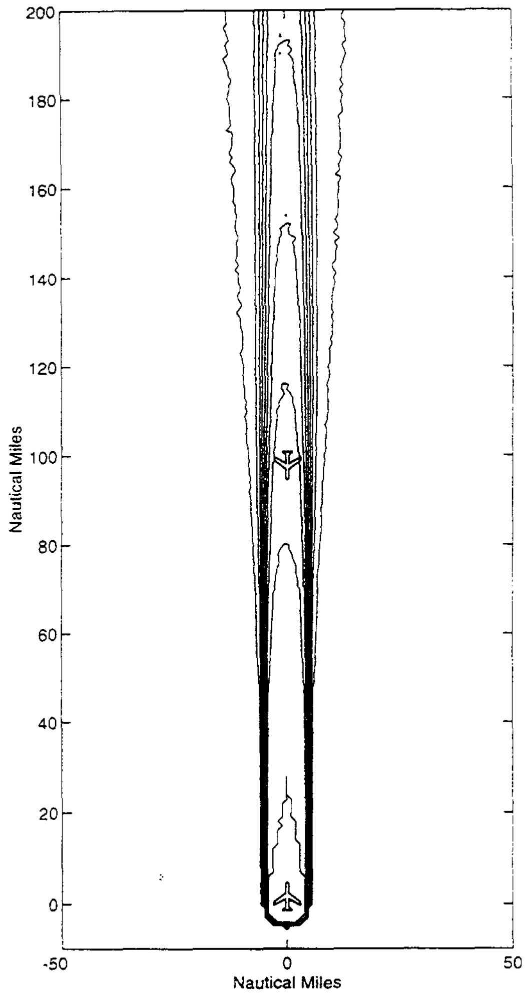


Fig. 6
Case 2: Potential for Intruder Course Change

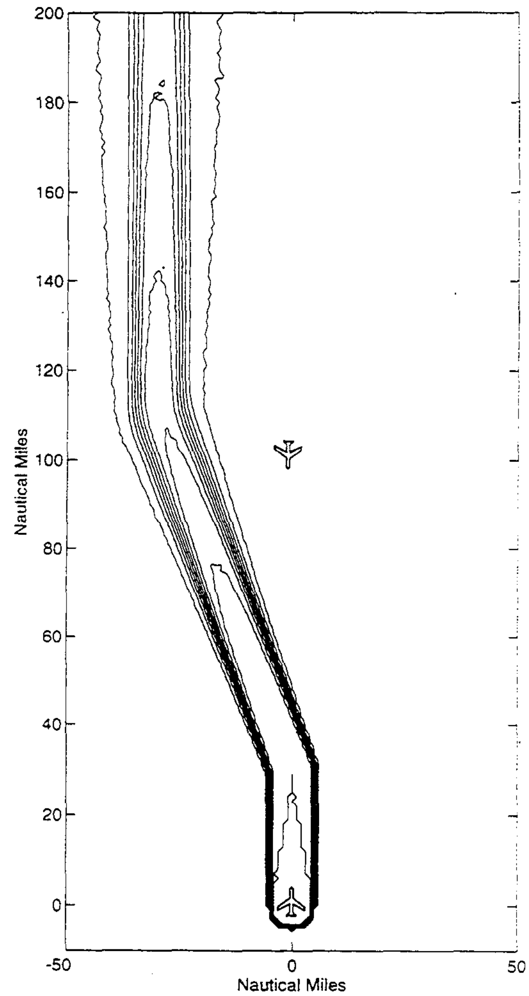


Fig. 7
Case 3: Host Aircraft Following Waypoints

origin (0, 0) with its track pointing up). The top of the chart is 200 nmi ahead of the host aircraft and represents a 15 minute time frame. Contours of constant conflict probability are shown, starting at 1.0 around the host aircraft and decreasing in increments of 0.1. For example, the intruder aircraft shown in the figure 100 nmi ahead of the host aircraft will produce a conflict with a probability of nearly 1.0. Variability and coarseness of the contours is a result of the accuracy of the Monte Carlo simulations. In this case, because the trajectory uncertainties are small, the corridor where aircraft must be located to generate conflicts is relatively narrow. Although the example shown is for a specific relative geometry and speed, similar maps can be generated for any situation.

A more interesting case to observe is when aircraft may change course at some time within the foreseeable future. In many cases, the intentions of each aircraft are not known for certain, but information regarding rules-of-the-road, past experience, or flight restrictions can be helpful in establishing the likelihood of various trajectories. In Fig. 6, the intruder aircraft is still headed in the opposite direction as the host, but now no explicit intention to maintain a straight course is assumed. For this particular case, the likelihood that the intruder would make a heading change is modeled as a Poisson distribution with an average of rate of 4 turns/hr. Also, the hypothetical flight rules in the airspace are assumed to require aircraft to restrict heading changes to less than 20° within a 15 minute period. Thus, potential changes in heading were modeled with an uniform distribution between ±20°. The resultant conflict map is shown in Fig. 6, again using contour spacing of probabilities of 0.1. Note that the probability of conflict decreases more rapidly as one moves farther from the host aircraft due to the increased uncertainty in the intruder's actions. The same intruder aircraft 100 nmi ahead of the host will now cause a conflict with a probability of approximately 0.83

because there is some chance that the intruder will perform a turn.

In the next example, shown in Fig. 7, additional intent information regarding knowledge of waypoints is added. For this case, the intent is supplied by the host aircraft in terms of 3 future waypoint locations in which the host will shift its flight path laterally. Again, the conflict map is shown with contour spacing of 0.1. Here, the intruder aircraft 100 nmi ahead of the host will not create a conflict as long as the intended path is followed.

Comparing Figures 6 and 7 provides some insight into the potential benefit of intent information. Consider for example the flight path shown in Fig. 7. If the host's waypoint information was not used in conflict detection, the situation would likely be modeled as shown in Fig. 6, resulting in a conflict alert. Such a conflict would be unnecessary, however, because as Fig. 7 shows, there is no conflict with the intruder aircraft.

Conflict maps can also be utilized in the examination of avoidance maneuver options for conflict resolution. Fig. 8 shows an example 30° right turn avoidance maneuver made by the host aircraft in response to a conflict alert in the example from Fig. 6. An additional uncertainty was included in this case to represent variability in pilot response time latency in initiating the turn maneuver. The response time was modeled as a Gamma distribution with an average response time of 1 min, and skewed such that there is a 95% probability that the avoidance maneuver will begin within 2 min of the alert.

Comparison of Fig. 6 with Fig. 8 shows the effect that the avoidance maneuver has on the probability of a conflict. Similar analysis can be performed to determine what other avoidance options (e.g., heading, speed, or altitude changes) could be used for the resolution. For multiple aircraft in the airspace, the maneuver could be checked to see if it induces further conflict with currently non-conflicting aircraft.

Vertical Conflict Examples

To more fully illustrate the utility of the Monte Carlo simulation approach, several additional examples are discussed in which a conflict exists in the vertical plane. Rather than depict conflict maps, however, the output of the Monte Carlo simulations is discussed here in terms of the expected unnecessary alert and missed detection rates.

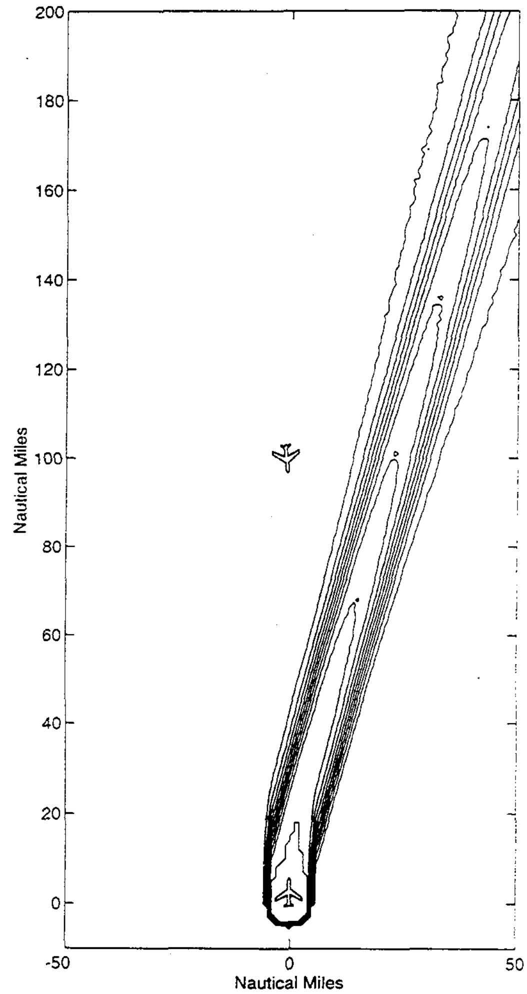


Fig. 8
Case 4: Host Aircraft Turns 30°

System Operating Characteristics

First, a brief description is required of the analysis method that is used. The approach is based on System Operating Characteristic (SOC) curves, which facilitate the visualization of the tradeoffs between unnecessary alerts and missed detections.[8] In any conflict detection decision, there is some probability that the conflict alert is unnecessary. Additionally, there is some probability that the conflict alert is successful in satisfying some performance constraint (e.g., preventing violation of the protected zone). As one varies the time at which a conflict alert is generated, these probabilities trade off against one another as described by an SOC curve.

In order to determine if an alert is successful, it is necessary to consider what resolution actions occur when the alert is generated. Some conflict resolution maneuver must be assumed so that it can be determined whether a conflict is ultimately averted by the alert.

Thus, an SOC curve is specific both to the encounter geometry and to the type of resolution action that is prescribed.

Fig. 9 shows an example SOC curve. As shown, if the conflict decision is made while aircraft are far apart (upper right corner of the plot), the probability of successful alert is high, but because action is taken so early, the probability of unnecessary alert is also high. As the conflict alert decision is delayed, the probability of successful and unnecessary alert both decrease as shown by the curve. If alerts are delayed too long (extreme lower-left corner of the plot) the alerts will not be successful, and there will be no unnecessary alerts as well.

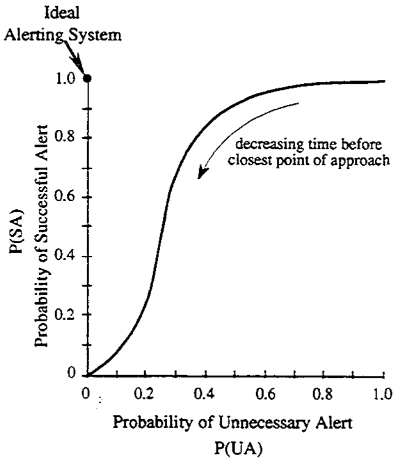


Fig. 9
System Operating Characteristic Curve

The ideal operating point for a system would be at the upper-left corner of the plot, where alerts are entirely successful while simultaneously having no unnecessary alerts. Due to uncertainties in the conflict dynamics, however, the SOC curve generally lies somewhere below this optimal point. The closer a system is able to operate near this optimal point, the more effective the system will be in terms of providing acceptable safety and minimizing nuisance alerts. Because the SOC curve is a function of the resolution action, different resolution options can be compared by examining differences in their SOC curves.[3]

Conflict Examples

The following examples show how the SOC approach can be utilized to show the effects of intent information on conflict analysis. In these examples, the lateral situation is the same as was shown in Fig. 5, and a conflict is defined as a loss of minimum separation of 5 nmi in the horizontal plane and 1,000 ft in the vertical plane. The vertical situation involves an intruder aircraft that is currently above the host aircraft and is descending directly toward it at 1,000 ft/min.

Two cases are considered. In the *nominal* case, it is not known whether the intruder will level off at some point or continue its descent. The vertical profile of the intruder is modeled such that it is equally likely that the intruder will level off at any altitude in a range above and below the host aircraft. Thus, a conflict may exist (the intruder continues to descend into the host) or a conflict may not exist (the intruder levels off safely above the host).

In the *intent* case, datalinked information from the intruder indicates that it will be continuing its descent at 1,000 ft/min through the host aircraft's altitude. For simplicity, it is assumed here that the aircraft maintains this descent rate perfectly. In both cases, however, the along- and cross-track uncertainties are the same as were introduced in Fig. 5.

SOC curves are plotted in Fig. 10 for both cases. The assumed resolution maneuver to a conflict involves a 5 sec delay when a conflict alert occurs, followed by a 1,000 ft/min climb. Other resolution maneuvers may be considered as well, but are not discussed here for brevity. In Fig. 10, the intent case SOC curve is shown by the solid line along the y-axis; the nominal case curve is shown by the dashed line. Operating points for each case are shown in terms of the time at which the conflict alert occurs, in increments of 10 sec relative to the time of Closest Point of Approach (CPA).

The SOC curve in Fig. 10 shows that an essentially ideal alerting decision could be made in the intent case

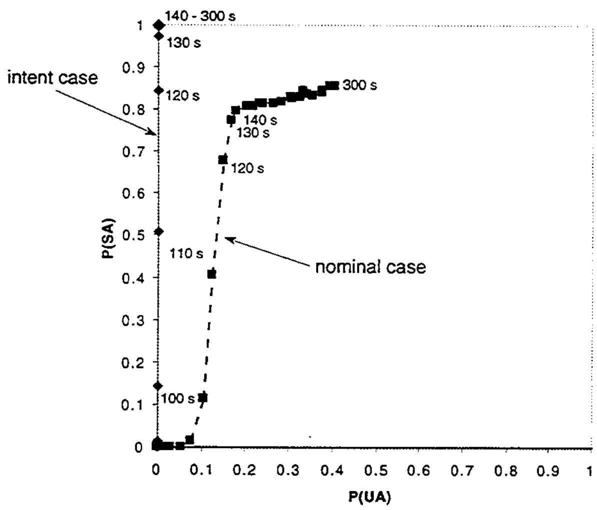


Fig. 10 SOC Curves For Conflict Examples
(conflict alert times shown are seconds before CPA)

(assuming that the intended path was indeed followed). By alerting any time prior to 140 sec before CPA, the host aircraft could avoid a protected zone violation with approximately 100% confidence. Simultaneously, because the uncertainties in this case are relatively limited, the probability of unnecessary alert is approximately 0. Alerting with less than 140 sec to CPA reduces the probability of successful alert as shown.

In the nominal case, the intruder could level off at an altitude above the host aircraft, and a climbing resolution maneuver may actually induce a conflict that would not otherwise have occurred. This also implies that such a resolution maneuver is unnecessary. As a result, the performance of conflict detection in the nominal case is lower than in the intent case, as shown by the dashed lines in Fig. 10. Notice also that the nominal SOC curve shows that successful alert probability cannot be increased beyond approximately 0.8 without greatly increasing unnecessary alert probability.

This case study shows one example of how intent information may significantly improve the quality of a conflict detection problem. Although the cases here are simplified, the overall approach can be applied to more complex situations in order to evaluate the potential benefit of having access to intent information. One additional issue that must be considered is the confidence with which the intended path will be followed. Such confidence can be included in the Monte Carlo simulations, and the resulting performance effect will be depicted in SOC curves.

Conclusions

The ability to rapidly and accurately estimate the probability of conflict will be an important factor in the acceptability of future conflict probe tools. The approach presented here relies on Monte Carlo simulation using a series of straight-line trajectories, and has been shown to allow accurate computation of probabilities in less than one second. Thus, real-time Monte Carlo based conflict probes are becoming feasible, and would allow fairly complex conflict scenarios to be examined.

The flexibility of the Monte Carlo approach to include intent information such as heading, speed, and altitude changes, lends it to be an attractive choice. Such an approach directly includes the intent information in the trajectory models, allowing dynamic updates of conflict probability in response to modifications in flight plan or autopilot mode. Additionally, the Monte Carlo simulation engine can be used as an evaluation tool to investigate issues such as appropriate protected zone size, requirements on sensor accuracy, or determination of appropriate resolution actions. However, a significant obstacle remains in that the probability distributions that are used must be relatively well-known. Otherwise, modeling errors may lead to inaccurate predictions of the probability of conflict.

Acknowledgment

This research was supported by the NASA Ames Research Center under Grant NAG-2-1111.

References

[1] Kuchar, J. K. and L. C. Yang, "Survey of Conflict Detection and Resolution Modeling Methods", AIAA-97-3732, AIAA Guidance, Navigation, and Control Conference, New Orleans, LA, August 11-13, 1997.

[2] Paielli, R. A. and Erzberger, H., "Conflict Probability Estimation for Free Flight." *Journal of Guidance, Control, and Dynamics*. Vol. 20. No. 3. pp. 588-596. 1997.

[3] Yang, L. C., and J. K. Kuchar, "Prototype Conflict Alerting Logic for Free Flight", AIAA Journal of Guidance, Control, and Dynamics, Vol. 20, No. 4, July-August, 1997.

[4] Krozel, J., and M. Peters, "Conflict Detection and Resolution for Free Flight", Accepted for publication in *Air Traffic Control Quarterly*, 1997.

[5] Kuchar, J. K. and L. C. Yang, "Incorporation of Uncertain Intent Information in Conflict Detection and Resolution", 36th IEEE Conference on Decision and Control, San Diego, CA, December 10-12, 1997.

[6] Cashion, P., Mackintosh, M.A., McGann, A., and S. Lozito, "A Study of Commercial Flight Crew Self-Separation", Digital Avionics Systems Conference Proceedings, Irvine, CA, October, 1997.

[7] Mellone, V. J. and Frank, S. M. "Behavioral Impact of TCAS II on the National Air Traffic Control System". *Seventh International Symposium on Aviation Psychology*. The Ohio State University. April 27, 1993.

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