COMPUTING RISK FOR UNMANNED AIRCRAFT SELF SEPARATION WITH MANEUVERING INTRUDERS

Jason W. Adaska, Numerica Corporation, Loveland, CO

Abstract

Safe integration of Unmanned Aircraft Systems (UAS) into the civil airspace requires the development of a sense and avoid (SAA) capability that enables UAS to remain "well clear" from other airborne traffic. Providing this capability when encountering non-cooperative, maneuvering intruders, such as those operating under Visual Flight Rules (VFR), is particularly challenging due to the inherent uncertainties in predicting the future trajectories of these intruders. Experts have suggested [1] that one way of meeting this challenge is to treat "well clear" as a separation standard that is quantified using the risk (i.e. probability) of Near Mid-Air Collision (NMAC) at some future time, and to alert pilots when action is required to avoid violating this separation. This involves (explicitly or implicitly) a stochastic model to quantify likely intruder trajectories. In this paper, we develop algorithmic tools for computing such risk by expanding techniques developed in the target tracking community. A central feature of this approach is the use of continuous-time, maneuver-based (rather than traditional diffusion-based) stochastic models that are more representative of variations in maneuvering aircraft trajectories over longer time scales. We argue that evaluating risk using such models is computationally viable for a real-time SAA system and can provide enhanced performance in terms of the traditional detection-theoretic metrics of probability of detection (Pd) and probability of false alarm (Pfa).

1. Introduction

Sense and avoid (SAA) capabilities for UAS are meant to serve as a replacement for traditional “see and avoid” regulations1 developed by the Federal Aviation Administration (FAA) for pilots physically located in the aircraft cockpit. It is useful to keep this origin in mind when considering needs of an SAA system. In particular, “see and avoid” vigilance is critical for situations in which some or all of the traffic is operating without command from air traffic control (ATC). In such instances, traffic may be operating under visual flight rules (VFR) and information concerning the flight path or intent of potential intruders may be unavailable. Further, communication with intruders cannot be guaranteed. Though flight rules clarify right of way, there is always a risk that an intruder could blunder into a Near Mid-Air Collision (NMAC) if a pilot is not vigilant to the possibility of unexpected maneuvers.

In the FAA’s Sense and Avoid Workshop [2], the SAA capability was segmented into two functions: Collision Avoidance (CA) and Self Separation (SS). The goal of Self Separation is to keep an unmanned aircraft “a safe distance from other aircraft so as not to cause the initiation of a collision avoidance maneuver.” Many aspects of SS functionality have yet to be fully defined. The precise definition of “well clear”, the coupling between SS and CA functions, and the relationship of ATC separation services to SAA are all important issues that will influence a Self Separation capability. In this paper, we address a more general concern that will likely influence the effectiveness of an SS solution regardless of future regulations. In particular, we are concerned with quantifying the risk (i.e., probability) of future NMAC when the intruder is uncertain and possibly maneuvering. We envision an alerting scheme that signals a pilot when the computed risk exceeds a safety threshold and action is required to remain “well clear”. The algorithm for computing risk could function as a stand-alone pilot decision aid, or it could be a component within a larger (and potentially autonomous) SS capability. The main focus of this paper is on the methods for computing the risk of future NMAC and the implications of these methods for the accuracy of the generated alerts.

The remainder of the paper is structured as follows. In Section 2, we discuss previous work on risk evaluations and how it relates to the methods presented here. Section 3 focuses on general

---

1 FAA Regulation 14 CFR Part 91.113 (b)
methods for modeling the dynamics of uncertain intruders over a future time horizon. In Section 4, we outline specific computational techniques for computing risk of future conflict with maneuver-based stochastic models. This section contains the primary technical contributions of the paper. In Section 5, we quantify the performance of a Self Separation alerting scheme using detection-theoretic metrics of probability of detection (Pd) and probability of false alarm (Pfa). Concluding remarks and plans for future work are presented in Section 6.

2. Risk of Future Conflict

In general, Self Separation is concerned with longer time horizons than Collision Avoidance. Here, we will be considering a three-minute time horizon though this number is not intrinsic to our approach. At these timescales, it is important to consider the possibility that an intruding aircraft, particularly one operating under VFR, could execute one or more unexpected maneuvers (e.g., turns, climb, descent, etc.). Such possibilities should have an impact on the SS function since they represent potential threats to ownship safety.

Time-based Risk Measures

Time-based measures of conflict risk such as “time to closest point of approach (CPA)” or \( \tau \) [3] are often used because they combine information about range and closing rate into a single value. A key challenge in dealing with uncertain intruders is that the true “time to CPA” is also uncertain. Further, the potential variation in this value could be extreme. As an example, consider an aircraft flying on a roughly parallel trajectory to a UAS. If the aircraft continues along a parallel trajectory, the time to CPA is very large. However, if the aircraft executes a coordinated turn toward the UAS in the near future, then the true time to CPA is significantly smaller. As result, it is not possible to assign a single \( \tau \) value to an uncertain, maneuvering intruder. With limited knowledge about the intent of the intruder, we can only put a distribution on likely \( \tau \) values. The simple example from above shows that the possible spread of this distribution can be extremely large. Thus, while it is possible to reduce this distribution to a single value such as the average or expected \( \tau \), doing so obscures the true risk.

Probability of Future NMAC

Recent work by researchers at MIT Lincoln Labs [1] has suggested that one method of accounting for risks such as the one described above, is to consider the possible variations in future intruder trajectories. They suggest a definition of “well clear” that is based on the risk (i.e. probability) of NMAC at a future time. The basic idea is that the relative state of the intruder and the ownship (assuming the ownship continues along a current trajectory) can be associated to a probability of NMAC by considering a distribution of possible intruder trajectories. In their approach, states with a probability that exceeds some defined safety threshold would not be considered “well clear”, and a pilot that enters such a state would be required to take action to return to a “well clear” state. Their analysis limited the relative state to variables such as the time to the closest point of approach (assuming straight-line extrapolation) and relative distance. An extensive offline simulation was used to evaluate a large number of randomly generated encounter scenarios to determine the probability of future NMAC associated to all possible relative states.

Our own work has progressed along somewhat similar lines. In particular, we are pursuing the development of a Self Separation capability for UAS that is based on the probability of future conflict. If this probability exceeds a given threshold, then the pilot is alerted. The envisioned capability would be similar to Traffic Advisory (TA) alerts provided by the Traffic Alert and Collision Avoidance System (TCAS) [3]. However, unlike TCAS, the alert decision would be directly based on the probability of future conflict, and it would reflect the possibility that the intruder could execute unexpected maneuvers.

As with the work at MIT LL, we are interested in computing a probability of future NMAC associated with a given state of the intruder and ownship. However, rather than using derived quantities such as time to CPA or range, we use the entire state estimate of the intruder (e.g., current position, velocity and relevant uncertainties) and the intended trajectory of the ownship as inputs into the probability computation. Instead of using an exhaustive, offline simulation that considers all possible states, we compute the desired risk value for the specific state confronting the UAS. A
probabilistic model of future intruder dynamics is used to facilitate this computation. One advantage of this approach is that it allows more information about the specific encounter to be considered, and this can lead to a more accurate risk assessment. Such an improvement in accuracy can lead to alerts that have lower false alarm rates for a given probability of detection. However, there are key technical challenges that must be overcome. In particular, we need to develop a stochastic model that mimics the true uncertainties in future intruder trajectories and a computationally efficient method of computing the NMAC probabilities in real-time.

3. Models of Maneuvering Intruders

The core of the our approach uses a model of future intruder trajectories to compute the probability of conflict. Below we list three of the salient alternatives for modeling these trajectories.

1. **Deterministic motion** – The trajectory of the intruding aircraft is predicted using deterministic straight-line extrapolation. Computations such as “time to CPA” are typically predicated on models of this type.

2. **Stochastic, Brownian diffusions** – A common model for uncertain, dynamic aircraft trajectories is a diffusion process where white noise is added to the velocity or acceleration components. Here we consider linear diffusions such Nearly Constant Velocity (NCV) or Nearly Constant Acceleration (NCA) models. Such models are extremely common in target tracking applications (see [4] Chapter 4) and form the basis of the celebrated Kalman Filter.

3. **Jump Linear Systems (JLS)** – This technique segments target motion into a discrete set of linear maneuvering modes (e.g., level flight, coordinated turn left/right, climb, descend). Though the dynamics within each mode are deterministic, the combined trajectories are stochastic due to uncertainty in the sequence and timing of maneuvers. For this reason, we refer to this approach as a “maneuver-based” model of uncertainty. Jump Linear Systems are also very common in target tracking applications. In particular, the Interacting Multiple Model (IMM) filter that can be viewed as a discrete-time approximation of a continuous-time JLS.

Clearly, straight-line predictions are unable to account for uncertainties due to maneuvers since they are deterministic by definition. While Gaussian diffusions can prove useful for predicting long-track and cross-track errors for aircraft with known intent [5], these models are less useful for modeling aircraft with unknown trajectories over long time horizons. One reason is that the noise driving the stochastic model (i.e., “process noise”) must be large enough to accurately account for possible turning maneuvers. However, using such a large noise power can lead to an unrealistic explosion in the uncertainty (i.e., covariance) of the aircraft location a few minutes in the future. We argue that Jump Linear Systems provide the most descriptive and useful model of maneuvering intruders for predictions over a 1-3 minute horizon because they directly model the mechanism that drives the uncertainty at these timescales, namely the execution of unexpected climb, descent, and turning maneuvers.

4. Computing Risk of Future Conflict

While Jump Linear Systems are more descriptive of maneuvering intruder trajectories, they present some unique challenges when used to compute risk probabilities. In this section, we discuss algorithmic methods that will be necessary for computing conflict risk using JLS.

The purpose of obtaining a stochastic model for future intruder trajectories is to use the model to compute probabilities of future conflict events. For this paper, the event of interest is a NMAC at some point in the next 3 minutes, but it is worth noting that the approach discussed here extends to other definitions of conflict as well. For instance, a “tau-like” conflict event could be defined by a state in which the range and closing rate are such that collision is predicted to occur in less than τ seconds. This is a more complicated definition of future conflict, but may be appealing because it can accommodate an explicit response time parameter τ.

**Conflict Risk at a Future Time**

Computing the probability of NMAC at a specific future time amounts to integrating the probability density for the future intruder location over the collision volume. This is depicted visually in Figure 1. It is clear from Figure 1 that the “shape” of the density matters enormously for this computation. Indeed, one way of understanding the impact of
choosing a particular intruder model is to note the induced shape of the predicted density. Deterministic straight-line predictions give rise to degenerate densities where all the probability is concentrated at a single point. Brownian diffusions always generate Gaussian probability densities. Jump Linear Systems generate probability densities that don’t have simple analytical expressions. However, they can be made to mimic the shape of empirically derived densities because they directly model the way in which aircraft are flown.

Section 5 indicate that this choice does not have a severe negative impact on algorithm performance.

Calculations with Jump Linear Systems

One of the biggest practical advantages of using diffusions to model intruder trajectories is that the probability density for the future intruder location is a multivariate Gaussian, and Gaussian distributions have many “nice” properties that make the computation of NMAC probabilities tractable. If we treat the NMAC conflict volume as a “squashed” ellipsoid to approximate its cylindrical shape (i.e. 500 foot horizontal radius, 200 foot vertical height), then this computation can be reduced to an integral in one dimension [6].

In contrast, the future state of a Jump Linear System is non-Gaussian. Indeed, there does not exist any closed form analytical expression for the density for any cases that interest us. Figure 2 illustrates the density for a 2D scenario in which we restrict aircraft motion to three horizontal maneuvers: level flight, coordinated turn left, and coordinated turn right. It is clear from this figure that the probability density in question is highly non-Gaussian. As a result, special techniques will be needed to compute the probability of NMAC when intruders are modeled using JLS. In particular, we will need methods that are amenable to a real-time SAA system, since exhaustive numerical sampling techniques (such as those used to construct the density shown in Figure 2) are much too slow for real-time application.

Figure 2. Evolution Of Probability Density For Jump Linear System

Moments Of Jump Linear Systems

While the analytical expressions for the density of a JLS are not available, it is possible to construct expressions for the evolution of the moments. In this section we provide such an expression.
We begin with a mathematical description for JLS dynamics. Define the state of the intruder at time \( t \) with the pair of variables \( \{x(t), z(t)\} \). \( x(t) \) defines the continuous state (e.g., position and velocity) of the intruder, while \( z(t) \) defines the discrete maneuvering mode (e.g., level flight, coordinated turn, climb, descend, etc.). We let \( M \) denote the number of maneuvering modes and the set \( \{A_1, \ldots, A_M\} \) specify the linear dynamics for each mode. For instance, a 6 DOF model during a coordinated turn of rate \( \omega \) would be described by the following linear system:

\[
A = \begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & -\omega & 0 & 0 \\
0 & 0 & 0 & 0 & \omega & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}.
\]

The dynamics of the intruder can then be described by the evolution equation

\[
\frac{dx}{dt} = A z(t) x
\]

where the subscript \( z(t) \) indicates the particular maneuver mode that is active at time \( t \). The evolution of \( z(t) \) is in turn described by a continuous-time Markov chain (CTMC). The value \( \lambda_{ij} \) is the transition rate for transitions from maneuvering mode \( i \) to mode \( j \) in this CTMC. The precise values of these rates could be informed by an offline analysis of the characteristics of VFR traffic. For example, if state \( i \) is level-flight, then \( \frac{1}{\sum_j \lambda_{ij}} \) could be set to the average time spent in level-flight as observed from empirical data.

We are interested in a computational procedure that allows calculating the \( k \)th moments of the density. For notational convenience, we define the \( k \)th Kronecker product as \( x \otimes^k \) (i.e., \( x \otimes^k = x \otimes x \otimes \cdots \otimes x \) \( k \) times) and the \( k \)th Kronecker sum as \( \oplus_k A \) [7]. The \( k \)th moment for maneuver mode \( i \) is expressed as \( m_i^k \).

Techniques outlined in [8] and references therein provide methods for deriving analytical expressions for the evolution of moments. In particular, the \( k \)th order moments satisfy the system of ordinary differential equations given below.

\[
\frac{d}{dt} \begin{bmatrix}
m_1^M \\
m_2^M \\
\vdots \\
m_M^M
\end{bmatrix} = \begin{bmatrix}
\bigoplus_k A_1 - \sum_k \lambda_1 I^\otimes_k & \cdots & \lambda_M I^\otimes_k \\
\vdots & \ddots & \vdots \\
\lambda_M I^\otimes_k & \cdots & \bigoplus_k A_M - \sum_k \lambda_M I^\otimes_k
\end{bmatrix} \begin{bmatrix}
m_1^1 \\
m_2^1 \\
\vdots \\
m_M^1
\end{bmatrix}
\]

Thus, computation of moments at a specific future time is equivalent to evaluating a matrix exponential.

**Refinement of the JLS Density**

The techniques discussed in the previous section provide a method for computing moments of the JLS density, but they do not provide a direct way of evaluating the density itself. However, it is these density values that are important for the risk computations. Simply using the moment calculations to compute first and second order cumulants (i.e., mean and covariance) will not be able to represent the non-Gaussian shape of the JLS density. In principle, it would be possible to compute all moments up to some high order \( k \), and then use this sequence to solve the “moment inversion problem.” The main stumbling block for this approach is computational limitations. Higher order moments require computing increasingly larger matrix exponentials and it would not be possible to perform these evaluations in a real-time system.

Instead of generating high order moments, we have developed methods to *partition* the set of possible intruder trajectories into multiple sets and then compute lower order statistics within each set of the partition. We can represent this decomposition symbolically with the expression

\[
\rho(x) = \sum_{a \in A} P(a) \rho(x | a) \approx \sum_{a \in A} P(a) \hat{\rho}_a(x)
\]

where \( \rho(x) \) is the true density, \( A \) is the set of partitions, \( P(a) \) is the probability contained in set \( a \), and \( \hat{\rho}_a(x) \) is an approximation of the density in that partition. By computing the mean and covariance for each partition, we use this approach to construct a “sum of Gaussians” approximation to the true density. It is important to note that this is an approximation because the mapping from the statistics to the density (e.g., mean and covariance to multivariate Gaussian) is approximate. The statistics themselves will be *exact*. Thus if a partitioning scheme allows partitions of an arbitrarily fine scale such that the variance in each set of the partition tends to zero, then the “approximate” density will converge toward the true density.
The success of this approach hinges on the specific partitioning scheme. In general, we are interested in choosing a scheme to meet two goals:

1. **Analytical methods are available for (quickly) evaluating statistics within each set of the partition.** – This allows the density approximation to be used in a real-time SAA system.

2. **Partitions approximately divide the density spatially.** – This allows the partitioning scheme to be effective in representing the non-Gaussian shape of the JLS density.

The key insight of the developed approach is that one can partition the intruder trajectories by partitioning the paths of the continuous-time Markov chain \(z(t)\) that drives the maneuver transitions. This can be done independently of the maneuver dynamics.

There are myriad ways to partition the sample paths of the CTMC that achieve the goals described above. One particularly effective method is to construct partitions based on the initial transitions of the CTMC. For instance, one simple partition would divide trajectories into two sets: (i) trajectories that do not maneuver (i.e., no transition in the CTMC) and (ii) those that do (i.e., at least one transition in the CTMC). The first set would consist of a single trajectory, the straight-line prediction. The second set would contain the rest of the maneuvering trajectories. It is possible to derive an auxiliary CTMC that describes the dynamics within partition (ii), and then use this auxiliary CTMC to compute statistics for the set with maneuvering trajectories. This provides a method for computing relevant statistics within both sets of the partition.

The binary partition (no maneuver, maneuver) in the example above can be generalized in the following way. A generic set in a partition specifies the first \(n\) maneuvers \(i_1, i_2, \ldots, i_n\) and (possibly) an indication of whether or not there are additional maneuvers within the time horizon. We use the symbol \(\emptyset\) to indicate there are no additional maneuvers and \(F\) to indicate that there is at least one additional transition. In this way a set within a partition can be described using a string of maneuvers states and (possibly) an ending character. Using this representation, the binary partition from above consists of the two sets \(0\emptyset\) and \(0F\) (where we have assumed that the CTMC always begins in state 0). One possible refinement of this involves three sets: (i) trajectories that do not maneuver, (ii) trajectories that turn left (maneuver mode 1), and (iii) trajectories that turn right (maneuver mode 2). Such a refinement would be represented by the set \(\{0\emptyset, 01, 02\}\).

![Figure 3. Partition Tree](image)

A partition tree can be used to represent a wide range of such partitions. The leaves of the tree define the sets of the partition and the path from the root to the leaf determines the string defining the given set. An example partition tree is illustrated in Figure 3. In this example, we assume three maneuvering modes: level flight, turn left, and turn right. It is assumed that right and left turns are always separated by level flight. Representative trajectories for each set in the partition are drawn beneath the corresponding leaf. The statistics for each set in the partition tree can be derived using moment computations as discussed earlier.

**Adaptive JLS Density Refinement**

The partition tree structure defined above suggests an extension to the process of refining the JLS density that is very relevant for a real-time SAA system with limits on computational resources.
While accurately representing the probability density is central to computing collision risk, it is not necessary to have a high fidelity representation everywhere. Ultimately, it is only the values of the density inside the collision volume that matter because these are the values that are integrated during the computation of NMAC probability (see Figure 1). Any time spent refining the probability density outside this area is wasted. On the other hand, if computational effort is adapted to refining the density inside the collision volume then a fixed level of accuracy can be achieved with far less computational effort when compared to approaches that refine the entire density uniformly.

The basic idea is to iteratively refine the density by “growing” the partition tree in ways that improve the estimate inside the region defined by the collision volume. Figure 4 shows how the evolution of partition tree can be directed to improve the accuracy of the density within a given region while leaving the representation outside this region at a comparatively coarse level. In this figure, a sum of Gaussians is used to approximate the density and is constructed using a density-partitioning scheme. The covariance ellipses are shown in black. The growth of the partition tree is adapted to the location of the collision volume (indicated by the red ellipse) to improve the accuracy of the density in that area. Note that the partition structure displayed in this figure contains sets that involve both specific maneuver sequences (as discussed above) as well as time-based windows for those maneuvers (e.g., right turn occurs between \( t_1 \) and \( t_2 \)).

![Figure 4. Adaptive Refinement Of Non-Gaussian JLS Density.](image)

5. Conflict Detection Performance

The utility of computing conflict risk for Self Separation is that it provides information to a pilot to help determine whether or not it is safe to continue along the currently planned trajectory. Ultimately, the pilot will use the risk assessment to decide whether or not a maneuver of the ownship is required. To facilitate an analysis of different methods for computing conflict risk, we will assume that this decision is made using a threshold test. Namely, we assume some minimum acceptable risk \( \tau \), and the pilot is alerted whenever the risk exceeds \( \tau \). Collision risks below this threshold will be considered safe and no alert will be issued.
**ROC Performance Curves**

In practice, the “best” value of $\tau$ would likely depend on a number of specific factors such as airspace class, aircraft platform, pilot preferences, regulatory requirements, etc. Picking such a value for $\tau$ is outside the scope of our current discussion. However $\tau$ is ultimately chosen, it will need to balance two competing factors: probability of detection ($Pd$) and probability of false alarm ($Pfa$).

**Simulation Study**

In this section we evaluate the performance of a Self Separation alert mechanism in terms of the generated ROC curve. The basic approach is to construct a series of engagement scenarios as depicted in Figure 5.

**Encounter Geometry**

The angle of the intruder with respect to the ownship ($\theta$) is varied between 0 and $2\pi$. The angle of the initial velocity vector with respect to the intruder location ($\phi$) is oriented between $-\pi/2$ and $+\pi/2$. The initial range ($R$) to the intruder is varied between 2.5 nmi and 5 nmi. We specify the encounter geometry with the triplet ($\theta$, $\phi$, $R$). For the purpose of generating ROC curves, we draw a series of encounters uniformly over the possible triplets.

**Aircraft Specifications**

The airspeed of the ownship is set to 80 knots (similar to the MQ-1 Predator) and is assumed to be headed due North for the length of the encounter. The intruder’s airspeed is 140 knots. The future trajectory of the intruder is unknown to the UAS ownship but may include unexpected maneuvers. Coordinated turns are executed at a two-minute turn rate. The statistics of intruder maneuvers are governed by a CTMC (i.e., the intruder is assumed to behave as a JLS). The transition rates of the CTMC are set such that the average time spent in level flight is 3 minutes. As a result, the intruder will frequently remain in level flight for the extent of the scenario. If a turn is executed it lasts for an average of 30 seconds.

**Onboard Sensor Characteristics**

In VFR environments, the exact position and velocity of the intruder may not be known. Instead these values need to be estimated using either measurements from an onboard sensor (i.e. Airborne Sense and Avoid - ABSAA) or a ground-based sensor (i.e. Ground-Based Sense and Avoid - GBSAA). For this simulation, we assume an ABSAA paradigm.

The sensor model is derived from a onboard collision avoidance radar with notionial accuracy characteristics given in Table 1.
### Measured Quantity

<table>
<thead>
<tr>
<th>Accuracy (σ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
</tr>
<tr>
<td>0.5 m</td>
</tr>
<tr>
<td>Range Rate</td>
</tr>
<tr>
<td>0.1 m/s</td>
</tr>
<tr>
<td>Azimuth Angle</td>
</tr>
<tr>
<td>0.25 degree</td>
</tr>
<tr>
<td>Measurement Update Rate</td>
</tr>
<tr>
<td>1 s</td>
</tr>
</tbody>
</table>

**Table 1. Noise Characteristic For Notional Radar**

**Evaluation of Alerting Performance**

The purpose of this simulation study is to understand the performance of a pilot alerting scheme in terms of ROC performance curves. To generate these curves, the initial engagement geometry is randomly sampled. Noisy sensor measurements are used to estimate the intruder state. Using only this state estimate a computation is performed to determine whether or not a pilot alert should be issued. We consider three methods based on the three intruder modeling approaches considered in Section 3: the JLS method for estimating the probability of NMAC, a Gaussian diffusion method (in which the noise statistics are matched to the JLS) for estimating the probability of NMAC, and a deterministic, straight-line extrapolation for estimating the closest point of approach. A threshold test is then performed to determine whether or not a pilot alert is generated.

The “true” intruder trajectory is drawn randomly according to the specified JLS model. This sample is then used to determine whether or not an NMAC would actually occur in this scenario by checking if the true intruder trajectory penetrates the collision volume of the ownship at some point during the simulated time horizon. By drawing a large number of possible trajectories and comparing alert behavior with various thresholds to “true” outcomes we can generate approximate ROC performance curves.

The results of this analysis are presented in Figures 6 and 7. Figure 6 compares the ROC curves for intruder prediction models using JLS, Brownian Diffusions, and CPA determined by straight-line extrapolation. We see from this comparison that the JLS outperforms the other two methods. Surprisingly, the deterministic prediction performs better than the diffusion prediction, though we expect that this is primarily driven by the large number of random intruder trajectories that do not maneuver or maneuver only slightly.

![Figure 6. ROC Performance Curves For Different Intruder Prediction Models](image1)

Figure 6. ROC Performance Curves For Different Intruder Prediction Models

![Figure 7. ROC Performance Curves For Relevant Trajectories](image2)

Figure 7 illustrates a follow-on analysis that focuses on those intruder trajectories in which an unexpected maneuver is relevant to the conflict prediction. In particular, the ROC curves in this figure are restricted to those scenarios in which the intruder (in truth) maneuvers before the time to closest point of approach. In general maneuvers that occur after this point would not have any impact on the occurrence of a NMAC. Here we see a more pronounced difference between the three methods. Again the JLS method provides the best performance, particularly at higher Pd values.

© 2012 Numerica Corporation
6. Conclusion

In this paper, we have presented a methodology for modeling maneuvering, uncertain intruders and developed algorithms for computing the probability that such intruders could cause a NMAC at a future time. These methods could be very useful in developing a Self Separation capability for UAS because they have the ability to alert pilots to potential NMAC risks in a 1-3 minute time horizon while limiting the occurrence of false alarms.

This work could be extended in a number of ways. Clearly, a deeper simulation analysis will be required to determine the detection performance that could be expected in a fielded system. One natural way of improving fidelity is to use statistically relevant dataset or a simulation infrastructure such the MIT Lincoln Labs “Uncorrelated Encounter Model” of VFR traffic to benchmark performance. The methods of estimating the NMAC probability using JLS have a number of possible extensions as well. A benefit of the JLS approach is that it has the ability to provide information about future NMAC risk as well as the possible intruder maneuvers that give rise to this risk. Such information could be very useful for suggesting avoidance maneuvers for a pilot or automated system if an avoidance action is required.

Acknowledgements

The author would like to thank NASA for the financial support of this research. Funding has been provided through a Small Business Innovation Research (SBIR) Phase II grant.

References


31st Digital Avionics Systems Conference
October 16-20, 2012

© 2012 Numerica Corporation