COMPUTING PROVABLY SAFE AIRCRAFT TO AIRCRAFT SPACING FOR CLOSELY SPACED PARALLEL APPROACHES

Rodney Teo, Department of Aeronautics and Astronautics, Stanford University, Stanford, CA

Claire J. Tomlin, Department of Aeronautics and Astronautics, Stanford University, Stanford, CA

Abstract

One of the requirements for closely spaced parallel approaches (CSPA) is the proof that possible blunders of the adjacent aircraft do not lead to a loss of separation. Our paper addresses this problem by considering the blunderer as a "pursuer" and the other aircraft as an "evader". We then use differential game methodology to find the safe region in which it is guaranteed that blunders do not lead to a loss of separation. We apply the technique to a practical case to show how the minimum safe longitudinal separation (MSLS) can be computed. Finally, simulation is performed to show how the technique can be used as an alerting algorithm for onboard on-line computation.

Introduction

Pairs of aircraft approaching closely spaced parallel runways have to be spaced apart such that they can perform the approach safely. The wake vortex and the possible blunders of one aircraft define the unsafe operating region for the other aircraft. Our paper addresses the computation of the unsafe region due to the latter phenomenon. We assume that position and velocity vector information of each aircraft are available from a Wide Area Augmentation DGPS system (WAAS) and transmitted to the other through a datalink, using, for example, Automatic Dependent Surveillance-Broadcast (ADS-B) [1].

Previous research [2,3,4] has focused on finding the minimum separation between the aircraft so that one aircraft can perform safe emergency evasive maneuvers (EEMs) should the other aircraft blunder in its approach; alerting and collision avoidance algorithms have been developed to provide directives to the evading aircraft. These works used empirical and Monte Carlo simulation techniques to obtain the results. An alternative method using analysis based on differential game theory was proposed by previous researchers [5,6] for aircraft collision avoidance problems. Our paper applies this approach specifically to the closely spaced parallel approach (CSPA) problem.

The first section of the paper describes how the CSPA is posed as a differential game problem. It also provides the analysis and explains the results obtained. The second section of the paper discusses how uncertainties can be incorporated into the results; the third describes how this approach can be implemented as an alerting algorithm for CSPA. The fourth provides the results on the minimum longitudinal separation required for safe CSPA. The last section demonstrates the implementation of the algorithm through simulation using a higher order aircraft model.

Computing unsafe blunder zones

An unsafe blunder zone is one that contains starting points from which an evading aircraft (evader) may not conduct the EEM safely because the blundering aircraft (blunderer) can find at least one trajectory to cause a loss of separation. Its boundary is the set of starting points of the EEM for which the blunderer needs to do its "best" by using its worst-case (i.e. minimum-time) control inputs to cause a loss of separation during the EEM. It is thus guaranteed that if the evader starts its EEM outside the unsafe zone, the blunderer will not be able to cause a loss of separation regardless of its choice of control inputs (within certain bounds) during the evasion.

To find the unsafe blunder zone or rather its boundary, our approach involves the following steps:

1. Choose an EEM.
2. Determine the end conditions in terms of relative position and orientation between blunderer and evader (terminal states) for which a loss of separation can occur.
3. Determine the worst-case control inputs required of the blunderer to drive the relative position and orientation between blunderer and evader (the

states) to these terminal states in the minimum time for the given EEM.

4. Propagate the states backwards in time from these terminal states using these worst-case control inputs.
5. Obtain the unsafe boundary from the analytical solutions of the state propagation.

The analytical solution of this unsafe zone is based on the solution of a two-person zero-sum differential game [7]. The blunderer is assumed to be the "pursuer" which tries its best to cause a loss of separation with the "evader".

The EEM could be a climbing turn, for example. We assume that if the evader completes the EEM safely, it would remain safe thereafter.

A set of equations of motion has to be selected for the analysis. The use of the higher order full point mass model of an aircraft [8] would be difficult for on-line implementation. The lower order kinematic model [9] is thus selected. This allows us to obtain analytical solutions of the worst-case trajectories, enabling on-line computation of the unsafe zones. However, we need to assume that speed and turn rate of the blunderer can be changed instantaneously. This gives conservative results which is consistent with the problem as safety is critical.

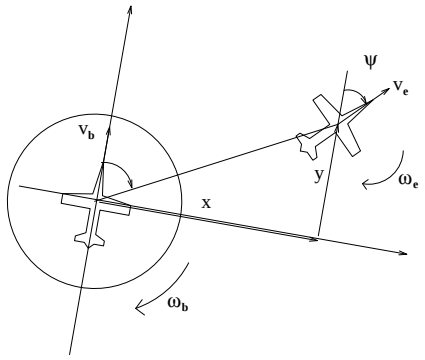

Figure 1. Reference frame

The kinematic model in coordinates relative to the blunderer is:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v_e \sin\psi - \omega_b y \\ v_e \cos\psi - v_b + \omega_b x \\ \omega_e - \omega_b \end{bmatrix} = f(x, y, \psi) \qquad (1)$$

The notation is shown in Figure 1 above.

Though this only models motion in a horizontal plane, the climbing turn EEM can still be analyzed by considering only the turn component of the maneuver.

The bounds on the control inputs of the blunderer are as follows:

$$\begin{aligned} v_{\min} \le v_b \le v_{\max} \\ |\omega_b| \le \omega_{\max} \end{aligned} \qquad (2)$$

Terminal conditions

A loss of separation is defined to occur when the two aircraft are 500 ft apart. As such, the terminal surface in the relative coordinates x, y and ψ is:

$$T = \{(x, y, \psi) : x^2 + y^2 = 500^2\} \qquad (3)$$

In differential game terms, the surface can be subdivided into the Usable Part (UP) and the Non-usable Part (NUP). The UP is the part of the terminal surface in which a further reduction in separation distance between the aircraft would occur immediately when the blunderer uses its worst-case control inputs. The NUP is that part of the terminal surface in which the separation distance would increase immediately even when the blunderer uses its worst-case control inputs. The boundary between the UP and NUP is the Boundary of the Usable Part (BUP) which is:

$$\{(x, y, \psi) \in T \mid \dot{r} = \sqrt{\dot{x}^2 + \dot{y}^2} = 0\} \qquad (4)$$

Substituting (1) into (4), we obtain:

$$\tan(s_1) = \frac{v_b - v_e \cos(\psi)}{v_e \sin(\psi)} \qquad (5)$$

where $s_1 = \tan^{-1}(x / y)$

Worst-case control inputs

As discussed earlier, the solution involves finding the worst-case control inputs that minimize the time to drive the state (x, y, ψ) to the terminal surface. The associated cost function is thus:

$$J = \int_0^t L\,dt = \int_0^t 1\,dt = t \qquad (6)$$

The Hamiltonian, H, is defined as:

$$H = p^T f = \begin{bmatrix} p_x & p_y & p_\psi \end{bmatrix} f \qquad (7)$$

where p is known as the adjoint vector.

The worst-case control inputs satisfy the following condition:

$$\min_{v_b,\omega_b}[H+L]=\min_{v_b,w_b}[p_x\dot{x}+p_y\dot{y}+p_\psi\dot{\psi}+1]=0 \quad (8)$$

Substituting (1) in (8), we obtain:

$$\min_{v_b,w_b}[-\omega_b(p_x y-p_y x+p_\psi)-v_b p_y + p_x v_e\sin(\psi)+p_y v_e\cos(\psi)+\omega_e p_\psi]=-1 \quad (9)$$

The adjoint vector satisfies the equations:

$$\begin{aligned}\overset{o}{p}_x &= p_y\omega_b\\ \overset{o}{p}_y &= -p_x\omega_b\\ \overset{o}{p}_\psi &= p_x v_e\cos(\psi)-p_y v_e\sin(\psi)\end{aligned} \quad (10)$$

with initial conditions $p_o=[x \quad y \quad 0]$. The ^{o} symbol represents $\frac{\partial}{\partial t}$ in retrograde time.

From (9), the worst-case control inputs are:

$$\omega_b=\omega_{\max}\,\mathrm{sgn}(A)$$

$$v_b=\begin{cases} v_{\max} & if \quad p_y>0\\ undefined & if \quad p_y=0\\ v_{\min} & if \quad p_y<0\end{cases} \quad (11)$$

where $A=-p_x y+p_y x-p_\psi$.

Worst-case control inputs on terminal surface

On the terminal surface, $A=0$. The worst-case control input, ω_b, is undefined. Thus, we need to look at how A changes over time. Taking the derivative of A with respect to time, we obtain $\overset{o}{A}=-p_x v_b$.

Similarly, on the terminal surface, when $y=0$, $p_y=0$. The worst-case control input, ω_b, is undefined. Taking the derivative of p_y with respect to time, we obtain $\overset{o}{p}_y=-p_x\omega_b$.

From the above derivatives and (11), we obtain the worst-case control inputs on the terminal surface as:

$$\omega_b=\omega_{\max}\,\mathrm{sgn}(x)$$

$$v_b=\begin{cases} v_{\max} & if \quad y>0\\ v_{\min} & if \quad y\le 0\end{cases} \quad (12)$$

This result is consistent with intuition. The blunderer's worst-case control inputs steer him towards the evader. Note however that when $x=0$ and $y=\pm 500$, ω_b is still undefined. This is possibly a terminal point of a worst-case trajectory that is formed by the coming together of infinitely many other worst-case trajectories, like the coming together of infinitely many tributaries to form a large stream. Such a worst-case trajectory is known as a universal surface in differential games. To check if these terminal points are where universal surfaces end, we check the following necessary but not sufficient conditions:

$$\begin{aligned}&A(x,y,\psi,p)=0\\ &H(x,y,\psi,p,\omega_b^*,v_b^*)=0\\ &\overset{o}{A}(x,y,\psi,p)=0\end{aligned} \quad (13)$$

where ω_b^*, v_b^* are the worst-case control inputs. We find that these conditions are satisfied for these terminal points. The worst-case control input, if it exists, is $\omega_b=0$. Upon plotting the trajectories from these terminal points, we find that a universal surface exists from $x=0$ and $y=500$ but not from $x=0$ and $y=-500$. A different surface exists from the latter terminal point. This is a surface from which infinitely many worst-case trajectories branch out, like infinitely many distributaries branching from a large stream. Such a surface is known as a dispersal surface.

Switching Times

The worst-case control inputs given above are those specifically at the terminal surface. As in (11), these inputs are a function of the state and adjoint vectors. From (1) and (10), we see that these vectors vary with time. Typically, (1) and (10) are integrated numerically and (11) is evaluated at each step of the integration to check when the worst-case control inputs change. However, in this case, we are able to integrate them analytically and consequently obtain analytically the switching times of the worst-case control inputs. This is an important advantage as it saves on computation effort and enables on-line computation.

Integrating (1) and (10) analytically, we obtain different sets of solutions depending on whether ω_b,ω_e are zero. For the case when $\omega_b,\omega_e\neq 0$, the solutions for $x,y,\psi,p_x,p_y,p_\psi,A$ are:

$$x = x_o\cos(\omega_b t)+y_o\sin(\omega_b t)+\frac{v_b}{\omega_b}[1-\cos(\omega_b t)] + \frac{v_e}{\omega_e}[\cos(\omega_b t+\psi_o)-\cos(\omega t+\psi_o)] \quad (14.1)$$

$$y = -x_o\sin(\omega_b t)+y_o\cos(\omega_b t)+\frac{v_b}{\omega_b}\sin(\omega_b t) - \frac{v_e}{\omega_e}[\sin(\omega_b t+\psi_o)-\sin(\omega t+\psi_o)] \quad (14.2)$$

$$\psi = \psi_o+\omega t \quad (14.3)$$

$$p_x = V\sin(\omega_b t+\alpha) \quad (14.4)$$

$$p_y = V\cos(\omega_b t+\alpha) \quad (14.5)$$

$$p_\psi = p_{\psi o}+\frac{v_e}{\omega_e}V[\cos(\alpha-\psi_o)-\cos(\omega_e t+\alpha-\psi_o)] \quad (14.6)$$

$$A = \frac{v_b}{\omega_b}V[\cos(\alpha+\omega_b t)-\cos(\alpha)] + V[x_o\cos(\alpha)-y_o\sin(\alpha)]-p_{\psi o} \quad (14.7)$$

where $V = \sqrt{p^2{}_{xo}+p_{yo}{}^2}$

$\alpha = \tan^{-1}(p_{xo}/p_{yo})$

Using these analytical solutions, we obtain the switches during the evasion. They are as follows for the different "starting" points parameterized by s_1 on the terminal surface (recall that $s_1 = \tan^{-1}(x/y)$):

1. For $0 < s_1 < \frac{\pi}{2}$, v_b switches from $v_{\max}$ to $v_{\min}$ after $t = \frac{1}{\omega_{\max}}(\frac{\pi}{2}-s_1)$

2. For $\frac{\pi}{2} \geq s_1 > \pi$, ω_b switches from $\omega_{\max}$ to $-\omega_{\max}$ after $t = \frac{2}{\omega_{\max}}(\pi - s_1)$

Similar results are obtained for $0 > s_1 > -\pi$. From $s_1 = 0$, we have a universal surface. To obtain the trajectory that makes up the universal surface, we set the control input to $\omega_b = 0$ and propagate the state vector backward in time. Recall that a universal surface is where many worst-case trajectories converge. These trajectories arrive at the universal surface through using worst-case control inputs corresponding to either $+\omega_{\max}$ or $-\omega_{\max}$ because this satisfies (11). Thus, to obtain these trajectories, we propagate the state vector backward in time from the universal surface with $\pm\omega_{\max}$.

Results

With the switching times, worst-case control inputs, BUP, UP, and the analytical trajectory equations, we compute the unsafe boundary. The bounds on the speed and turn rate of the blunderer are chosen as 150 ± 20 kt and ± 4.2 deg/s respectively. The speed and turn rate of the evader making the evasion are 150 kt and 4.2 deg/s respectively. These correspond to a possible real approach condition. With these values, we compute the result shown in Figure 2. As with the rest of the paper, we assume that the blunderer is meant to land on the left runway and the evader is meant to land on the right runway. Thus during the approach, the evader will normally be on the right of the blunderer. In this context, both left and right EEMs were considered. A left EEM flies the evader towards the approach path of the blunderer. The EEM considered is a constant speed climbing turn where the heading change begins after a delay of 2.6 sec and is constant at 4.2 deg/s up to a heading change of 45 deg. The delay is to account for the fact that when an aircraft starts a turn, the full turn rate can only be achieved after it has rolled to the required bank angle for the turn. Also, as explained, climbing turns may be considered in our model (1) by ignoring the climb component of the maneuver.

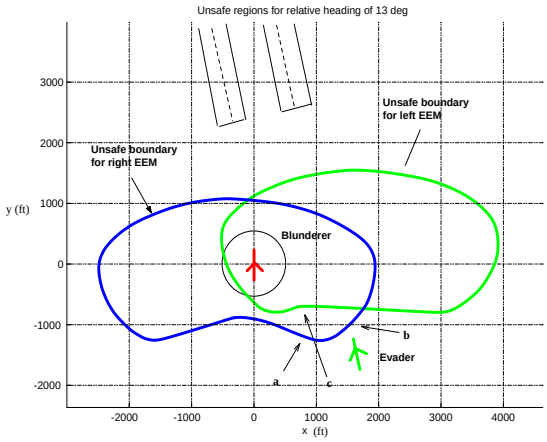

Figure 2. Unsafe boundary for left / right EEM

To illustrate the worst case control inputs of the blunderer required to minimize the time to cause a loss of separation, the trajectories for initial conditions marked 'a' and 'b' in Figure 2 are given in Figure 3. The trajectory corresponding to initial condition 'a' (in terms of relative position and orientation between blunderer and evader) requires

the blunderer to first turn away from the evader and then turn towards the evader; this is the shortest-time trajectory which terminates at 500 ft from the evader. This is known as a swerve in differential game theory. The trajectory corresponding to initial condition 'b' requires a velocity switch from $v_{\max}$ to $v_{\min}$ at the switching time computed using (15).

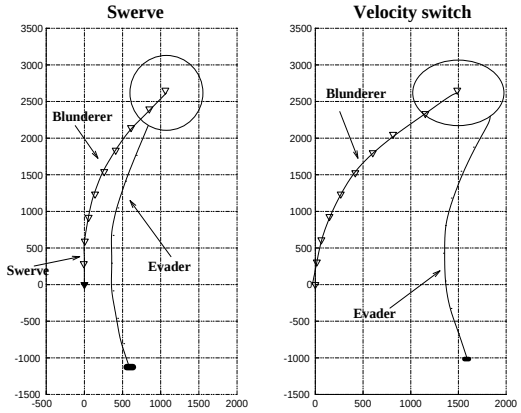

Figure 3. Swerve and velocity switch (relative distance never less than 500 ft)

There are instances in which a left EEM allows the evader to have a shorter longitudinal separation from the blunderer as shown by 'c' in Figure 2. To illustrate this, Figure 4 shows the trajectories from 'c' where a right and a left EEM were taken whilst the blunderer uses its worst-case control inputs. As expected, the right EEM leads to a loss of separation whereas the left EEM keeps the evader safely away from the blunderer. In fact, the evader grazes the 500 ft radius circle around the blunderer. This is the characteristic of reaching the terminal surface on the BUP. A left EEM, however, is not usually used as the evader will have to fly into the wake of the blunderer.

Incorporating uncertainties

As discussed, the distance from the boundary of the unsafe zone to the evader's position is a function of the blunderer and evader's velocities and turn rates, and their relative position and heading. The sources of uncertainty are thus:

- the blunderer's actual speed and turn rate (these are considered in the differential game framework),
- the measurement of the blunderer and evader's positions and headings, and
- the "drift" in position and heading in between sensor updates.

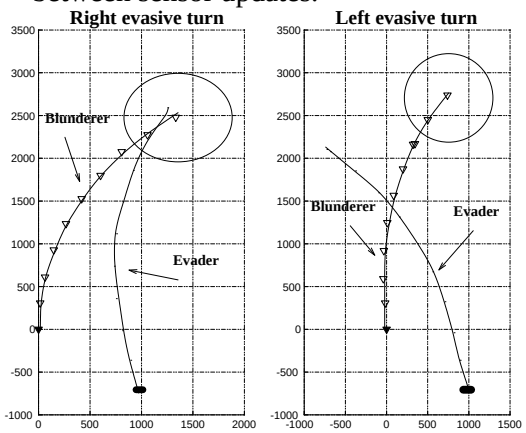

Figure 4. Right EEM (relative distance less than 500 ft); Left EEM (relative distance never less than 500 ft)

We assume that the position uncertainties due to measurement and drift to be uniform in all directions. Let them be denoted by δz and δz^{drift} respectively. Let $\delta\psi$ and $\delta\psi^{drift}$ be the corresponding uncertainties in heading. To obtain the worst-case relative uncertainties, we sum the uncertainties due to measurement and drift as follows:

$$d = \delta z_b + \delta z_e + \delta z_b^{drift} + \delta z_e^{drift}$$
$$\delta\psi = \delta\psi_b + \delta\psi_e + \delta\psi_b^{drift} + \delta\psi_e^{drift}$$

The relative position error is thus contained in a circle around the evader with radius d as shown in Figure 5. As a result of the relative heading error, we have a range of the unsafe boundaries for relative headings $\psi \pm \delta\psi$ that must not intersect the error circle for safety. Thus, for on-line computation, we need to check that the error circle remains outside the range of unsafe boundaries within the region contained within the two tangents to the error circle that intersect at the blunderer as shown in Figure 5.

Typically, δz^{drift} is larger in the in-track direction than in the lateral direction. This will give us an error ellipse instead. In this approach, we assume an error circle. This simplification is, however, not overly conservative if we are interested in the position of the evader with respect to the front and back of the unsafe boundary rather than the side of the boundary. This is the case for approaches to runways spaced 750ft apart with the parameters used

above. If the evader were on the side of the boundary, we can use a less conservative δz^{drift}. Also, if we miss updates, the $\delta z^{drift}, \delta\psi^{drift}$ terms would grow with time.

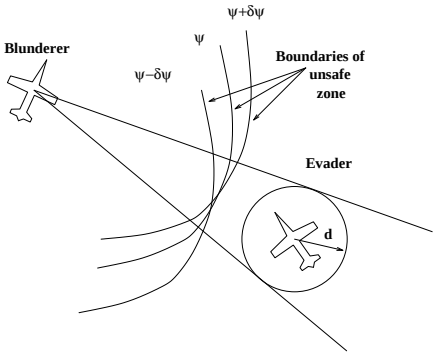

Figure 5. Incorporating uncertainties

Implementation

The implementation of the above algorithm in CSPA involves two aspects:

- the computation of the MSLS (minimum safe longitudinal separation), and
- the on-line computation of the unsafe boundary.

The MSLS is that which ensures the unsafe boundary does not intersect the error circle around either aircraft for almost all of the deviations in lateral position and heading that can be expected in a normal approach. This is computed by finding the furthest forward and backward the unsafe boundary can shift within the approach corridor of the adjacent aircraft, for such deviations in lateral position and heading.

During the approach, the aircraft should fly their approach speeds and maintain separation greater than the MSLS. At the same time, both aircraft compute on-line the unsafe boundary based on the current states of both aircraft and checks if they are outside the unsafe zones. If both aircraft fly nominal approaches and maintain the MSLS, the on-line computation would show that each aircraft is outside the unsafe zone. When deviations in the approach occur, the on-line computation will check if such deviations cause the unsafe boundary to encroach upon the error circle around the aircraft. If so, an EEM as assumed in the unsafe boundary computation, should be conducted. Such an EEM is guaranteed to maintain separation under those conditions assumed.

MSLS required for CSPA

The MSLS required for CSPA is computed for the case of two similar general aviation aircraft making the approach to parallel runways. The lateral and heading deviation data used in the computation is taken from [10] which contains flight test results of a Beechcraft Queen Air general aviation aircraft performing instrument approaches. Two of the various approaches used in the flight tests are termed the CDI (Course Deviation Indicator) and the 'tunnel' approaches. They both use the WAAS corridor approach as opposed to the conventional angular approach using ILS (Instrument Landing System). They differ however in the cockpit displays. The first uses the CDI whilst the second uses the 'tunnel-in-the-sky' visual display. The latter displays a tunnel describing the approach path over a synthetic display of the view ahead. This was demonstrated to result in lower Flight Technical Error (FTE).

Of the 27 approaches flown in the flight tests in [10], it is reported that 95% of the approach path deviations are within the bounds given in Table 1. (The heading deviation was estimated from the lateral deviation.) Therefore we will use these bounds in our analysis.

Table 1. 95th percentile deviation bounds

Deviation	CDI	Tunnel
Lateral	+/- 129 ft	+/- 32 ft
Heading	+/- 8.4 deg	+/- 4.2 deg

For the computation of the safe longitudinal separation, the bounds on the speed and turn rate of the blunderer are 100 ± 5 kt and ± 10.9 deg/s respectively. The speed and turn rate of the evader making the EEM are 100 kt and 8.9 deg/s respectively. The fact that the 8.9 deg/s turn rate cannot be achieved instantaneously is also accounted for in the model by including a delay of 2.5 sec. In addition, a pilot response delay of 2 sec is considered in the EEM. This delay was observed in flight tests conducted in [11]. The sensor error for position and heading are 15 ft and 0.3 deg respectively. The drift error for position and heading are 15 ft and 5 deg

respectively. The drift error is estimated assuming a 1 Hz sensor update rate.

The results, given in Figure 6, shows that for the nominal case on 750 ft runways, the MSLS is about 1500ft for FTE corresponding to the CDI approach. The tunnel approach, which has a better FTE, requires slightly less separation than the CDI approach for 750ft runways and this improvement becomes significant for 1700 ft runways. This is because the tunnel approach results in smaller lateral deviation, which keeps the aircraft outside of the unsafe zone. Also, the significant improvement is because of the geometry of the unsafe zone as in Figure 2.

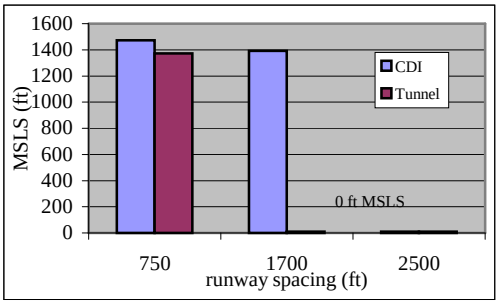

Figure 6. Cockpit display vs MSLS

Figure 7 compares the MSLS required between EEM types when the CDI approach is assumed. For right only EEMs (recall that the evader is assumed to be on the right of the blunderer), the MSLS is only slightly more than for the right and left EEM (i.e. the evader can choose between a left or a right EEM) for 750 ft runways. On the other hand, for 1700 ft runways, there is no difference in the MSLS. However, a safe left EEM has the advantage of being safe even beyond the end of the EEM (Figure 4) if it can be made without encountering the wake vortices of the blunderer.

Figures 8 and 9 show the change in MSLS when certain parameters are varied from the nominal case that assumes 750 ft runways and the CDI approach. Figure 8 shows that the MSLS increases as the bank rate of the evader in the EEM decreases. The increase becomes more significant at lower bank rates. This is because low bank rates can be thought of as having long delays in responding to a blunder. Figure 8 also shows the relationship between the heading change of the evader in the EEM versus MSLS. Recall that the analysis guarantees safety up to a given heading change in the EEM. When the EEM assumes a larger heading change, we have to guarantee safety for a longer duration. Thus, the larger the heading change, the larger the unsafe zone and consequently, the larger the MSLS.

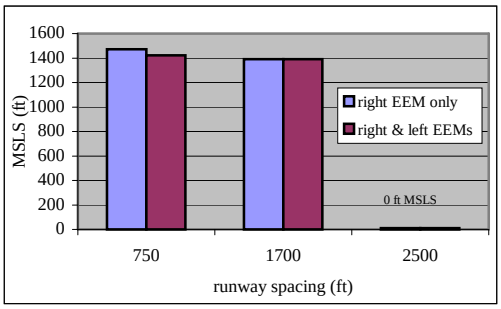

Figure 7. EEM effect on MSLS

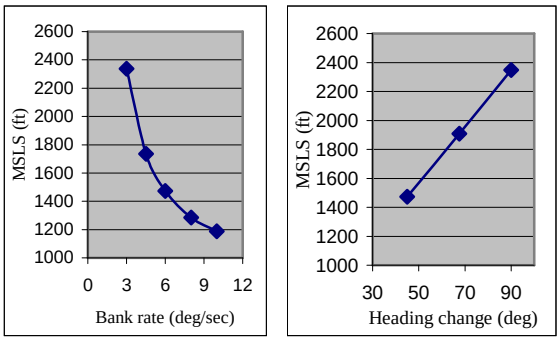

Figure 8. Evader bank rate and heading change vs MSLS

Figure 9 shows that the larger the bound on the speed of the blunderer, the larger the MSLS. Figure 9 also shows that the larger the bound on the turn rate of the blunderer, the larger the MSLS.

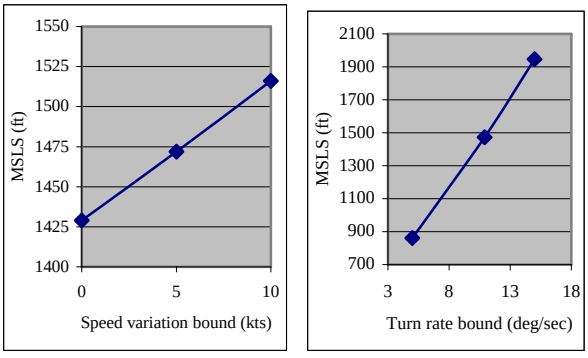

Figure 9. Bounds on blunderer speed and turn rate vs MSLS

Simulation

To test the implementation of the on-line computation of the algorithm, simulation using a linearized low speed Boeing 747 model [12] is conducted for approaches to 750 ft runways. The chosen EEM is a climbing turn away from the adjacent aircraft. It consists of a roll to a 30 deg bank angle and a heading change of 45 deg whilst maintaining constant speed and climbing at a flight path angle of 3 deg.

For computing the MSLS and the simulation, the normal deviations during approach assumed are given in Table 2:

Table 2. Normal deviations during approach

Deviation	CDI	Tunnel
Lateral	+/- 100 ft	+/- 30 ft
Heading	+/- 8 deg	+/- 3 deg

The bounds on the speed and turn rate of the blunderer are 150 ± 5 kt and ± 2.7 deg/s respectively. The speed and turn rate of the evader making the evasion are 150 kt and 2.7 deg/s respectively. The sensor errors in position and heading are 15 ft and 0.2 deg respectively and the drift errors in position and heading are 15 ft and 2 deg respectively. The drift error is estimated based on a sensor update rate of 1 Hz. The MSLS for the given parameters is 1191 ft. The simulation is conducted with an initial longitudinal separation of 1191 ft.

The blunderer maneuvers are taken from [4]. They consist of 8 approach types:

1. 30 deg heading blunder.
2. 15 deg heading blunder.
3. Constant 5 deg bank angle blunder.
4. Slow 10 deg heading blunder.
5. Slow 5 deg heading blunder.
6. Fake blunder.
7. Drift away then over adjust blunder.
8. Normal approach.

In all approaches except Approaches 6, 7 and 8, the blunderer crosses the approach path of the evader. Altogether, 16 simulations are performed, using the 8 approach types above with the evader in the rear and in the front.

In addition to the unsafe boundary computation, the on-line computation includes a check on the lateral separation between the two aircraft. When the lateral separation becomes closer than 200 ft when the evader is behind the blunderer, an alert is issued to conduct an EEM. This is to keep the evader out of the wake vortex region of the blunderer.

The simulation shows that the on-line computation is possible at a rate of about 2 Hz using MATLAB. For all the blunders simulated, the two aircraft never came closer than 500 ft. All the blunders with the blunderer in front result in an EEM except for Approach 6. All the blunders with the evader in front did not warrant an EEM. This is because the MSLS for an evader flying in front is less than the MSLS for an evader flying in the rear.

As expected, the simulation for Approach 8 shows that no EEM is required.

Figure 10 shows the results of Approach 1 and Approach 7. Approach 7 required an evasion because the heading of the blunderer became large when it was trying to get back to its approach path.

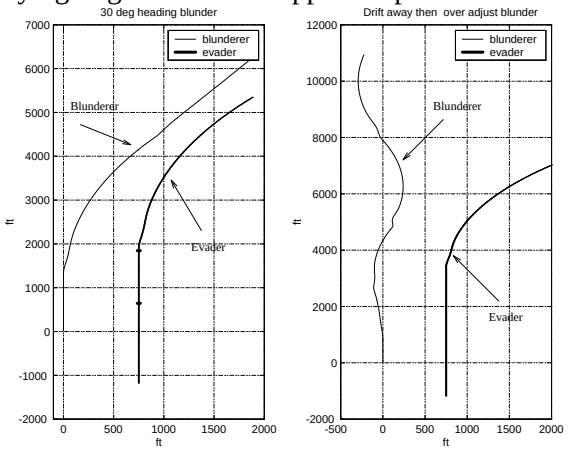

Figure 10. 30 deg heading change, and drift away and over adjust blunders

Conclusion

A game theoretic approach for computing the unsafe blunder zone in CSPA has been proposed. The approach provides guarantees on safety within the bounds on the control inputs used. The approach also provides insight on what the worst case blunders are. By using a kinematic model, the computation of the unsafe boundaries can be performed on-line whilst considering uncertainties. The implementation of the algorithm is described and simulation is conducted to demonstrate its implementation. In addition, some sensitivity studies on the parameters affecting the MSLS is conducted.

The proposed approach is conservative as it does not consider vertical separation and it assumes that the blunderer can change its turn rate and speed instantaneously. Nevertheless, it provides a means of guaranteeing safety against blunders for CSPA subject to the parameters used in the computation.

References

[1] *Radio Technical Commission for Aeronautics. Minimum Aviation System Performance Standards for Automatic Dependent Surveillance-Broadcast (ADS-B)*, 1997, Technical report, RTCA-186, DRAFT 4.0.

[2] J. Hammer, 1999, *Study of the Geometry of a Dependent Approach Procedure to Closely Spaced Parallel Runways*, Proceedings of the IEEE/AIAA 18th Digital Avionics Systems Conference, 4.C.3-1.

[3] B. Carpenter and J. Kuchar, 1997, *Probability-Based Collision Alerting Logic for Closely-Spaced Parallel Approach*, Proceedings of the AIAA 35th Aerospace Sciences Meeting and Exhibit, AIAA 97-0222, Reno, NV.

[4] S. Koczo, 1996, *Coordinated Parallel Runway Approaches*, NASA Contractor Report 201611.

[5] Vincent, T. L., Cliff, E. M.,Grantham, W.J., and Peng, W. Y.,1972, *A Problem of Collision Avoidance*, EES Series Rept. 39, Univ. of Arizona.

[6] Tomlin, C. J., 1998, *Hybrid Control of Air Traffic Management Systems*, Phd. Thesis, Univ. of California, Berkeley.

[7] Isaacs, R., 1965, *Differential Games*, John Wiley and Sons, New York.

[8] Ardema, M.D., Rajan, N., 1987,*An Approach to Three-Dimensional Aircraft Pursuit-Evasion*, Journal of Computational Mathematics Applications, Vol. 13, NO. 1-3, pp. 97-110.

[9] Merz, A. W., 1972, *The Game of Two Identical Cars*, Journal of Optimization Theory and Applications, Vol 9, No. 5, pp 324-343.

[10] S. Houck, A. Barrows, B. Parkinson, P. Enge and D. Powell, 1999, *Flight testing WAAS for use in closely spaced parallel approaches*, Proceedings of the ION-GPS 1999 Meeting, Nashville, TN.

[11] S. Houck, D. Powell, 2000, *Dual Airplane Flight Dynamics*, AIAA Atmospheric Flight Mechanics Conference 2000, AIAA 2000-4316, Denver, CO.

[12] Nelson, R.C., 1989, *Flight Stability and Automatic Control*, McGraw-Hill.