Abstract—Medium Term Conflict Detection (MTCD) is a planning tool used mainly by the planning controller but also by the tactical controller. MTCD continuously monitors each active aircraft in a given airspace and notifies air traffic controllers if two aircraft have a potential conflict within a certain look ahead time. Various field trials have shown that the safety benefits of MTCD tool may be limited because of large numbers of False Alarms and Missed Detects. There are two main methods employed for conflict detection in MTCD viz. Fixed Threshold and Covariance. In this paper we investigate the failure patterns in terms of conflict characteristics that lead to False Alarms and Missed Detects. We have used a computational “Red teaming” concept to challenge (in-silico) MTCD conflict detection methods for a massive number of conflict scenarios that are evolved based on the MTCD algorithm’s accuracy and correctness in identifying a potential conflict. In particular we look into MTCD conflict detection time window threshold (8-20mins) and its effect on the performance of the algorithm. The key findings of the paper are, in Fixed Threshold method, by reducing the conflict alert threshold window from 8 min-20 min to 8min-15min can reduce the False Alarms by almost 52%. By eliminating conflicts where duration is less than 30 seconds can reduce False Alarms by 13%. In Covariance method by reducing the conflict alert threshold window to 8min-15 min can reduce False Alarms by up to 66% and eliminating the conflict alerts with conflict duration of less than 30 seconds can further reduce the False Alarms by 19%. For Missed Detects reducing the conflict alert threshold window by raising the lower end from 8 min to 9 min can reduce the Missed Detect rate by 52.4% and 75.1% for Fixed Threshold and Covariance method respectively. Results thus indicate that refining of the system is in process. The field trial results indicated high confidence of ATC controllers in MTCD, as it significantly improve the performance of MTCD in terms of reducing False Alarms and Missed Detects.

Keywords: Conflict Detection, MTCD, Missed Detects, False Alarms, Safety Assessment

I. INTRODUCTION

Air transport continues to change and expand both in volume and in the areas of the world it serves. Current air traffic control (ATC) operations are highly structured and restrictive to help air traffic controllers (ATCs) manage their workload within the constraints imposed by safety requirements [1]. Air traffic providers world wide are exploring new paradigms (e.g. SESAR [2] and NextGen [3]) and advanced ATM tools (e.g. Conflict Detection & Resolution [4]), to fly more cost effective routes while increasing the capacity of the ATC system while maintaining the same level of safety.

Medium Term Conflict Detection (MTCD) [5] is one such automated tool developed by the European Organization for the Safety of Air Navigation (Eurocontrol) to help ATCs detect and predict aircraft involved in conflicts, thus relieving them of much of the routine workload and cutting down on unnecessary interventions [6]. MTCD continuously monitors each active aircraft in a given sector/airspace, notifying the controller if two aircraft have a potential conflict within a certain look ahead time. Trajectories are examined in pairs and reported if they come close within a pre-defined separation threshold [7]. It is expected that MTCD will facilitate moving from the current reactive form of air traffic control to more proactive control, thereby balancing the workload of tactical and planning ATCs, enhancing sector efficiency and providing safer and better service to airspace users. It is also hoped that MTCD, by pro-actively solving problems during sector planning, will help to reduce tactical workload [8].

The problem of assuring safety in MTCD is one of evaluating that the system identifies all possible conflicts correctly, which is a very challenging task. This evaluation process must ensure the ability of MTCD to cope with the most safety-critical situations and complex scenarios.

MTCD has been under development for some time now, and trials have been undertaken at various European Air traffic control centres (e.g. Amsterdam ACC and Maastricht UAC [9]) as well as in the U.S. FAA’s User Request Evaluation Tool (URET) [10]. Advanced field trials were conducted in shadow mode at Rome ACC in 2004 [11], and further operational refinement of the system is in progress. The field trial results indicated high confidence of ATC controllers in MTCD, as it provided better insight into conflict problem geometry due to display of minimum distance and information on aircraft position well ahead of time. However, these benefits were limited by a high rate of False Alarms and Missed Detects.

The European Organisation for the Safety of Air Navigation (EUROCONTROL) is an international organization whose primary objective is the development of a seamless, pan-European Air Traffic Management (ATM) system. The goal for EUROCONTROL is to develop, coordinate and plan for implementation of pan-European ATM strategies and their associated action plans.
generated by MTCD system in Maastricht UAC and Rome trials.

False Alarms are a necessary evil due to inherent uncertainties in air traffic environment and inefficiencies in existing communication, navigation and surveillance (CNS) infrastructure [12]. With Missed Detects, problem lies in when to flag a potential conflict as a real conflict. However, by understanding the nature of conflict characteristics that lead to False Alarms and Missed Detects in MTCD, we can gain a better insight into the algorithm’s performance and limitations in the given ATC environment.

In our previous work we evaluated the performance of Airborne Separation Assurance algorithms by using an evolutionary computational framework that evolves complex air traffic conflict scenarios [13] using the “Red Teaming” concept [14]. Red teaming is a concept, normally used in defence, which refers to studying a problem by anticipating adversary behaviours [15]. When done in simulations, the behaviour space is divided into two groups: one controlled by the red team which represents the set of adversary behaviours, while the other is controlled by the blue team which represents the set of defenders.

Through red teaming, analysts can learn about the future by forward prediction of scenarios. This in turn provides a wider and deeper understanding of potential adversary options and behaviour that can expose potential vulnerabilities in a system [16]. This provides a “fence” against the accepted assumptions and the accepted solutions. When applying red teaming to MTCD, the blue team represents the concept (the MTCD algorithm) and the red team represents the adversaries (the conflict scenarios). The primary use of red teaming in our case is to understand the capabilities and susceptibilities of the MTCD system in order to exploit inherent weaknesses in the system to improve the overall performance of the system.

In this paper, we attempt to use this concept to identify patterns in conflict characteristics that lead to False Alarms and Missed Detects by MTCD. We extend our previous work by incorporating conflicts that involve turns in air traffic scenarios, and by additional checks (Flight plan, cleared flight level) that reduce the possibility of False Alarms. We specifically analysed the alert-time window threshold to identify “when detecting early is too early and when detecting late is too late”. We evaluate two different probe methods used in MTCD’s conflict detection process, i.e. Fixed Threshold Conflict Detection (used in EUROCONTROL’s MTCD tool [7]) and Covariance Method Conflict Detection (used in FAA AERA-2 tool and tested in the URET [17], [18]).

The paper is organized as follows. We first present the formulation of MTCD probe methods, followed by conflict characteristics used in generating conflict scenarios. We then present the simulation environment and framework used for evaluating MTCD probe methods. This is followed by the evaluation metrics used in the paper, and the experimental setup. We then conclude by interpreting some results and discussion along with our future work.

II. MEDIUM TERM CONFLICT DETECTION

MTCD is a controller’s tool which assists them to identify potential conflicts within a certain look ahead time. A typical MTCD system performs the following functions [19]:

- Calculation of aircraft trajectories: It computes the current as well as projected trajectory of all the aircraft for the given look ahead distance;
- Monitoring of an aircraft’s progress against the trajectory, so that the aircraft is within the required navigation performance for a given airspace;
- Detection of conflicting trajectories; and
- Presentation of this information to the controllers from 8 to 20 minutes ahead.

A. Probing Methodologies in MTCD

The method used in MTCD to predict an aircraft’s future position and to determine whether or not two aircraft are in conflict is called a probing method. Two different probing methods are commonly used in MTCD [20]:

1. Fixed Threshold Conflict Method: In this method, the time (T2CPA) and distance (CPA) of closest point of approach (CPA) are first computed for each potential conflict pair. Then these two thresholds (CPA and T2CPA) are used to recognise the event of a conflict. If the T2CPA is greater than a fixed look-ahead interval, or if the CPA distance is greater than a threshold distance, the aircraft pair is declared non-conflicting. If the CPA time is less than the look-ahead interval (normally 8 min to 20 min) and if the CPA distance is less than the threshold (normal 5 nm, which is the usual separation standard for en-route airspace in Europe), the aircraft pair is tentatively declared conflicting.

2. Covariance Method: In this method, the covariance matrix is used to determine whether the aircraft are likely to be in conflict. The covariance matrix is calculated for each aircraft pair and compared to a threshold to determine if a conflict is likely to occur. The threshold is determined based on the expected covariance of the aircraft trajectories.

Figure 1. Covariance method for conflict detection in MTCD
2. **Covariance Method**: In this method, error ellipse path uncertainty regions are computed at the T2CPA together with the CPA distance for each potential conflict pair. The error ellipses are based on covariance calculations obtained by modelling surveillance errors and aircraft path following errors. Figure 1 illustrates the Covariance Method Conflict Detection in horizontal plane relative to an ownship aircraft. The intruder path relative to the reference path is a straight line with CPA distance (Rmin) at T2CPA (Tmin). The horizontal separation is a circle around the origin with radius equal to separation standard (Rsep). An intruder is not conflicting if the predicted uncertainty ellipse at each time point does not intersect the Rsep circle. The uncertainty ellipse shown is an aggregate error obtained by combining individual covariance errors in predicting the ownship and intruder paths. The uncertainty ellipse is only evaluated at the minimum approach time Tmin. The closest approach is then found by sliding the ellipse along the intruder path, generating the dotted line in Figure 2. The uncertainty error Runc can be determined as the maximum distance of the ellipse from the intruder path. If Rmin < Rsep + Runc, a potential conflict exists.

In order to make the MTCD conflict probe process more robust, tentative conflicts are further validated on flight plan (for turns) and cleared flight levels (CFL) for conflicts in transition phase i.e. climb and descent.

---

**B. Level off and Turn Segment Check**

To make the conflict detection process in MTCD more robust, tentatively conflicting flights are checked for altitude change segments at or before the conflict position and for turn segments before the closest point of approach. This may reduce the rate of False Alarms.

- **Flight Level Check**: As illustrated in Figure 2 top, when a tentative conflict is declared based on any of the methods mentioned above, it is then checked whether any of the two flights are in transition phase (climbing or descending). If they are, then their cleared flight level (level off altitude) is checked to see whether any one of them level off before reaching the altitude where conflict is projected. If the level off altitude difference is greater than the vertical separation standard, the tentative conflict pair is removed from the potential conflict list.

- **Flight Plan check for Turn Segments (Trajectory Change Point)**: As illustrated in Figure 2 bottom, turn segments are a major cause of False Alarms. When a tentative conflict is declared based on any of the methods mentioned above, the flight plans of the conflict flight pair are checked for any trajectory change point (turn) before the projected conflict position. If the turn distance before the projected position is greater than the separation standard, the tentative conflict pair is removed from the potential conflict list.

---

**C. MTCD Conflict Probe formulation [20], [21], [7]**

The MTCD conflict probe module uses the position and velocity data from the radar signatures of air traffic. It calculates for the loss of separation with a certain look-ahead time interval. If so, it then calculates the position of aircraft at the closest point of approach. It also calculates the time at which the separation will be lost and the time of closest point of approach. In 2D the closest point of approach could be outside the 3D conflict interval. Therefore the MTCD algorithm first calculates the conflict interval for the horizontal dimension (R) and then for the vertical dimensions (H), and then combines them.

To calculate the conflict intervals the relative position $dx$ and speed $dv$ of the intruder is calculated in Cartesian coordinates for the vector calculation. We used right handed reference frame with origin at the ownship position. The equation of relative motion $\mathbf{x}$ of an intruder with reference to the ownship is given by:

\[
\begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{bmatrix} =
\begin{bmatrix}
  dx_1 \\
  dx_2 + t \times dv_2 \\
  dx_3
\end{bmatrix}
\]

where $dx_1$, $dx_2$, $dx_3$, $dv_1$, $dv_2$, $dv_3$ are the changes in position and velocity of the intruder with respect to the ownship.
The time \( t \) in this equation is relative, implying \( t = 0 \) is now. First the vertical conflict interval \([t_{in\text{-vert}}, t_{out\text{-vert}}]\) is found by using vector calculation and solving for \( t \)

\[
\begin{align*}
|x_j| &= H \\
|dx_j + t \cdot dv_j| &= H
\end{align*}
\]

(1)

(2)

\[dx_j + t \cdot dv_j = H \cup dx_j + t \cdot dv_j = -H\] (3)

\[t_1 = \frac{H - dx_j}{dv_j}, \quad t_2 = \frac{-H - dx_j}{dv_j}\] (4)

which gives

\[ t_{in\text{-vert}} = \min(t_1, t_2) \] (5)

\[ t_{out\text{-vert}} = \max(t_1, t_2) \] (6)

Then the horizontal conflict interval is calculated as the intersection of line and circle in the horizontal plane. To find these times, the following equation is solved for \( t \):

\[x_1 + x_2 = R^2\] (7)

\[\left( dx_1 + t \cdot dv_1 \right)^2 + \left( dx_2 + t \cdot dv_2 \right)^2 = R^2\] (8)

\[ (dv_1^2 + dv_2^2) \cdot t^2 + 2(dx_1 dv_1 + dx_2 dv_2) t + \left( dx_1^2 + dx_2^2 - R^2 \right) = 0\] (9)

This is a quadratic equation form with:

\[a = dv_1^2 + dv_2^2\]

\[b = 2(dx_1 dv_1 + dx_2 dv_2)\]

\[c = dx_1^2 + dx_2^2 - R^2\]

and discriminant \( D = b^2 - 4ac \)

If the discriminant is negative, there is no intersection and hence no conflict. If the discriminant is positive, the interval of horizontal conflict is given by:

\[t_{in\text{-horz}} = \frac{-b - \sqrt{D}}{2a}\] (10)

and

\[t_{out\text{-horz}} = \frac{-b + \sqrt{D}}{2a}\] (11)

If time is negative this refers to a time in the past (conflict already occurred).

The vertical and horizontal intervals are combined and checked for overlap. For the combined \( t_{in} \) the maximum of both values is used (conflict only if it has simultaneously intruded the protected zone horizontally and vertically)

\[ t_{in} = \max(t_{in\text{-vert}}, t_{in\text{-horz}}) \] (12)

For the time of leaving the conflict the minimum of both values is used.

\[ t_{out} = \min(t_{out\text{-horz}}, t_{out\text{-vert}}) \] (13)

If \( t_{out} \) is before \( t_{in} \) there is no overlap and hence no conflict. The time when conflict will happen can then be computed as:

\[ t_{conflict} = t_{now} + t_{in} \] (14)

For Covariance Method we assumed a lateral offset rms error for each aircraft of 0.5 nm, i.e. on each trial the ownship and intruder are offset laterally from the nominal flight path by a random distance with mean zero and standard deviation 0.5 nm. At each probe update time (ten seconds) the ownship and intruder paths are predicted forward using pseudo tracker estimates of along-track position and speed, and estimated along-track wind shear. Nominal surveillance and wind forecast rms errors assumed for CPA predictions in this paper are:

- Radar along-track position error = 0.15 nm
- Along-track wind forecasting error = 6 knots
- Along-track wind shear error = 4 knots per 100 nm
- Radar tracker steady velocity error = 3.8 knots

These assumptions were used in [20] and are consistent with the wind forecasting and target tracking given modern monopulse radars.

**III. CONFLICT SCENARIOS**

We previously developed a methodology for algorithmically generating air traffic scenarios with desired conflict characteristics [22]. We break from the classical approach of pre-scripting conflict events in air traffic scenarios, and use genetic algorithms [23] instead to evolve conflicts. Since our aim is to evolve scenarios that challenge an MTCD algorithm, the objective of the genetic algorithm is to evolve increasingly complex conflict scenarios so that the MTCD algorithm can incur maximum failure (in terms of evaluation metrics).

The performance of any conflict detection algorithm largely depends upon the characteristics of the conflict scenario [24]. A head-on conflict between two cruise level aircraft may be easy to detect as compared to an in-trail conflict between two descending aircraft.
Based on [25], the following conflict characteristics at the CPA affect the performance of a conflict detection algorithm; these are encoded in the chromosomes data structure which is used by the genetic algorithm:

- **Horizontal separation (HS):** The horizontal distance between two aircraft at the CPA.
- **Vertical separation (VS):** The vertical distance between two aircraft at the CPA.
- **Conflict geometry Intruder (CGI):** The phase of intruder aircraft at the CPA. This can be climb, cruise or descent.
- **Conflict geometry Ownship (CGO):** The phase of ownship aircraft at the CPA. This can be climb, cruise or descent.
- **Conflict angle (CA):** The relative conflict angle between the two aircraft at the CPA.
- **Turn angle (CA):** The turn angle for the ownship, starting 60–50 nm before the two aircraft reaches their CPA.

A real-valued representation with a linear chromosome structure is chosen to represent an air traffic scenario. Every gene of the chromosome encodes the characteristics of a conflict-pair, representing a conflict between a pair of aircraft.

In this paper, we extend the chromosome representation by incorporating the turn angle, to deliberately introduce conflicts that have a turn angle. This is achieved by initially adding a random value between 0 degrees and 30 degrees to the ownship heading as it enters into a conflict with the intruder, and then later is guided by the genetic algorithm evaluation process.

As illustrated in Figure 3 and 4, every chromosome represents an air traffic scenario, where each pair of conflicting aircraft in the scenario is represented as a gene of the chromosome.

### IV. SIMULATION ENVIRONMENT AND EVALUATION FRAMEWORK

#### A. Air Traffic Operations & Management Simulator (ATOMS)

To simulate air traffic scenarios and evaluate the performance of the MTCD algorithm, we use the ATOMS air traffic simulator [26]. ATOMS is a medium fidelity air traffic simulation system where a scenario of real and/or artificial traffic can be simulated in real or fast time mode. The fast time mode simulation enables testing a large number of scenarios in a reasonable time. The two above-mentioned MTCD probe methods were programmed into ATOMS and every flight pair is checked for conflict based on the selected probe method at 10 second time interval. ATOMS is thus used as the evaluation objective function for air traffic scenarios: every time it is called with a scenario, it evaluates the performance of the MTCD algorithm in a given scenario and returns performance measures i.e. Missed Detects and False Alarms.
B. Evolutionary Computation framework

Figure 5 illustrates our methodology. The initial population (initial scenarios) is used to further generate complex conflict scenarios, which are then evaluated using ATOMS. A state of the art genetic algorithm (NSGA-II) [27] is used to evolve increasingly complex air traffic scenarios. Scenarios with higher fitness (i.e. higher MTCD failures) survive the evolutionary mechanism of the genetic algorithm and breed further to come up with more difficult conflict scenarios.

This evolutionary mechanism helps to evolve complex conflict scenarios that cause MTCD algorithms to fail; as the evolution proceeds, it will find scenarios in which the MTCD algorithm fails even more. Therefore, the fitness of a scenario is based on how many conflicts are not resolved successfully by the MTCD algorithm. If the MTCD algorithm performs well (detects all the conflicts) in a scenario, the scenario fitness is low; if it performs poorly (fails to detect the conflicts), the fitness of the scenario is high.

C. Evaluation Metrics

We use Missed Detects and False Alarms as our primary metrics for evaluating MTCD algorithm performance. They are defined as follows:

- Missed Detects (MD): This metric represents the number of potential conflicts that resulted in a separation violation but the MTCD algorithm failed to detect them.
- False Alarms (FA): This metric represents the number of conflict alerts that didn’t actually materialize into a separation violation, but the MTCD algorithm labelled them as potential conflicts.

Thus the objective functions can be defined as a maximization problem in which the objective of the evaluation process is to maximize the events of Missed Detects and False Alarms in an air traffic scenario on which MTCD is applied.

\[
\begin{align*}
\text{MAX} & \begin{cases} 
  f_1 = FA \\
  f_2 = MD
\end{cases}
\end{align*}
\] (15)

V. EXPERIMENTAL DESIGN

For genetic algorithm the crossover probability is set to 1.0, the mutation probability is set to 0.01, the number of generations is 100, each experiment is repeated 10 times with 10 different seeds, the same 10 seeds are used in all experiments. These parameter settings are not claimed to be optimal but our previous work suggests that they are reasonable for this problem.

We use a population size of 50 where there are 50 flights in each scenario, with 100 pre-programmed conflicts with different conflict characteristics (conflict angle, relative speed, geometry (climb, cruise, descent), horizontal and vertical distance at CPA). This gives 5 million flights with at least 2.5 million conflicts. More conflicts may result from over lap of aircraft trajectories in a scenario.

No conflict alert is generated if two flights are already in conflict state (i.e. $R < 5\text{nm}$ and $H < 1000 \text{ ft}$) and if the conflict alert time (CPA) is less than the MTCD threshold (8 mins). Flights continue their flight paths as programmed unless they reach the sector boundary, where they are deactivated (removed from the simulation). If an aircraft pair has several conflicts, it is recorded each time until the alert is removed or it reaches the 8 minutes threshold window. At the end of each scenario execution, the ATOMS simulator reports all the conflicts that occurred as well as the conflicts that were identified by the MTCD algorithm. The MTCD algorithm reports all the conflicts it identified along with their probe characteristics. As illustrated in the Venn diagram in Figure 6, this is then matched with the actual conflicts to compute the Missed Detects and False Alarms. A generic sector in the Australian National Airspace region [S32.0 E142.0 S38 E 150] is selected. Minimum flight altitude is set to 15,000 ft and maximum flight altitude is set to 38,000 ft. Speed of the aircraft is within the band of 300 knots to 550 knots. All flights are activated within the sector and deactivated at the sector boundary.

VI. RESULTS & INTERPRETATIONS

We first report results for each probe method of MTCD in terms of the False Alarms and Missed Detects. We begin by presenting the overall False Alarms and Missed Detects in the 100th evolved generation of conflict scenarios. In the given alert window of 8-20 minutes, 185,216 conflict alerts were raised by the Fixed Threshold method, of which 8,660 were False Alarms, and there were 1,463 Missed Detects. With the Covariance Method 129,652 conflict alerts were raised, of which 4,760 were False Alarms, and there were 1,053 Missed Detects.

We analysed the sensitivity of the probe methods to conflict alert time. In particular we looked into the conflict alert time window interval of 8 mins to 20 mins for both the methods, where we looked into when a conflict was first alerted and when it was last alerted in the conflict alert time window. We also analyzed the duration of a conflict alert for both the
methods. For both the probe methods we also looked into the conflict geometry of the aircraft pair and the conflict angle for the conflicts that resulted in False Alarms and Missed Detects.

<table>
<thead>
<tr>
<th>100th Generation</th>
<th>Fixed Threshold</th>
<th>Covariance Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>False Alarms</td>
<td>4.6%</td>
<td>3.6%</td>
</tr>
<tr>
<td>Missed Detects</td>
<td>0.82%</td>
<td>0.84%</td>
</tr>
</tbody>
</table>

Figure 7. Time to CPA for conflict alerts that lead to False Alarms in Fixed Threshold method sorted on first detect time in the 8–20 min window.

Figure 8. Time to CPA for conflict alerts that lead to False Alarms in Fixed Threshold sorted on last detect time 8–20 min window.

Figure 9. Duration of conflict alerts that lead to False Alarms in Fixed Threshold.

Figure 10. Conflict alerts with duration greater than 30 seconds that lead to False Alarms in Fixed Threshold method.

A. Fixed Threshold Conflict Probe

1) False Alarms: Figure 7 and Figure 8 show the time to CPA for conflict alerts that lead to False Alarms with the Fixed Threshold method, sorted on first detect time and by last detect time, in the 8–20 mins threshold window. It can be seen that until 15 minutes (representing 48% of conflict flight pairs) there is almost a linear relationship between the False Alarms and conflict alert time. In 16–19 minutes window (25% of conflict flight pairs) the False Alarm rate increases non-linearly, and in the last 20 minutes duration (27% of conflict flight pairs) the False Alarm rate was maximum.

Figure 9 shows the duration of conflict alerts that lead to False Alarms in Fixed Threshold method. It can be seen that for a large number of flights the conflict duration is small i.e. the conflict was flagged and it was subsequently removed. We investigate two time intervals i.e. 30 seconds and 60 seconds and eliminated the flight pairs that have conflict duration in these time intervals.
False Alarms and conflict alert time. In 16–19 minutes window (for 43% of conflict flight pairs) the False Alarm rate increases non-linearly, and in the last 20 minutes duration (23% of conflict flight pairs) the False Alarm rate was maximum.

Figure 12. Valid Conflicts that do not continue in the time window and lead to Missed Detects in Fixed Threshold method, sorted on first detect time in the 8–20 min window in Covariance Method.

Figure 13. Valid Conflicts that were discontinued before 9 min in conflict alert window and lead to Missed Detects in Fixed Threshold method.

Figure 17 shows the duration of conflict alerts that lead to False Alarms in the Covariance method. In this method also there are a large number of flights for which the conflict duration is small. We did a similar investigation on two time intervals i.e. 30 seconds and 60 seconds and eliminated the flight pairs that have conflict duration in these time intervals.
Figure 14. Valid Conflicts that were discontinued before 10 min in conflict alert window and lead to Missed Detects in Fixed Threshold method.

Figure 15. Time to CPA for conflict alerts that lead to False Alarms in Covariance Method sorted on first detect time in the 8-20 min window.

Figure 16. Time to CPA for conflict alerts that lead to False Alarms in Covariance Method sorted on last detect time in the 8-20 min window.

Figure 17. Duration of conflict alerts that lead to False Alarms in Covariance Method.

Figure 18 and Figure 19 presents the False Alarms for conflicts that have duration of more than 30 seconds and 60 seconds respectively.

Eliminating the flight pairs that have conflict duration less than 30 seconds reduces the False Alarms by more than 19%, by eliminating the flight pairs that have conflict duration less than 60 seconds reduces the False Alarms by more than 34%.

2) Missed Detects: Figure 20 shows, for the Covariance method, the time footprints of conflicts that actually happened but did not continue in the conflict alerts window of the probe method till 8 minutes threshold and lead to a Missed Detect.

Similar to Fixed Threshold method, we reduced the conflict alert time window by raising the lower end from 8 min to 9 minutes and then to 10 minutes, to see how many Missed Detects that were not captured by the probe method can then be captured if the time window is reduced. Figure 21 and Figure 22 show the valid conflict alerts that were discontinued before 9 min and 10 min respectively in the conflict alert window and lead to Missed Detects.

It can be seen that in the Covariance method, by reducing the 8–20 min threshold window to 9–20 min Missed Detects can be reduced by 75.1%, and by further reducing the threshold window to 10–20 min the Missed Detects can be reduced by another 58.9%.

VII. CONFLICT CHARACTERISTICS

A. Conflict Geometry

Figure 23 shows the conflict geometry between the ownership and intruder for False Alarms generated by the Fixed Threshold and Covariance conflict detection methods. The maximum number of False Alarms in both methods is generated when the ownership and intruder are in cruise mode. For the Fixed Threshold method, 20% of the False Alarms were generated if the ownership is in transition mode i.e. climbing or descending, while 10% of False Alarms were generated when both aircraft are in transition phase. For the Covariance method, 16% of the False Alarms were generated if the ownership is in transition mode, while 10% of False Alarms
were generated when both aircraft are in transition phase. Both
probe methods have similar behaviour when it comes to False
Alarms in terms of conflict geometry.

Figure 24 shows the conflict geometry between the
ownship and intruder for Missed Detects generated by the
Fixed Threshold and Covariance conflict detection methods.
Both probe methods generate the largest proportion of Missed
Detects when the conflict pair is in Cruise phase. However,
Fixed Threshold method is more susceptible than the
Covariance method to generating a Missed Detect when the
ownship is climbing or descending and the intruder is in cruise
phase.

Figure 18. Conflict alerts with duration greater than 30 seconds that lead to
False Alarms in Covariance Method

Figure 19. Conflict alerts with duration greater than 60 seconds that lead to
False Alarms in Covariance Method.

Figure 20. Valid Conflicts that do not continue in the time window and lead to
Missed Detects in Covariance method, sorted on first detect time in the 8–20
min window.

Figure 21. Valid Conflicts that were discontinued before 9 min in conflict
alert window and lead to Missed Detects in Covariance method.

Figure 22. Valid Conflicts that were discontinued before 10 min in conflict
alert window and lead to Missed Detects in Covariance method.
Figure 23. Flight conflict geometry of False Alarms generated by Fixed Threshold and Covariance method.

Figure 24. Flight conflict geometry of Missed Detects generated by Fixed Threshold and Covariance method.

Figure 25. Conflict angle of flight pairs that generated False Alarms for Fixed Threshold and Covariance method.

Figure 26. Conflict angle of flight pairs that generated Missed Detect for Fixed Threshold and Covariance method.

VIII. CONCLUSIONS

In this paper, we attempt to analyse the sensitivity of MTCD algorithms to conflict threshold time, and to identify patterns in aircraft conflict that lead to False Alarms and Missed Detects by MTCD algorithms, for two given probe methods.

In Fixed Threshold method, reducing the conflict alert threshold window from 8–20 min to 8–15 min can reduce the False Alarms by almost 52%. Further eliminating the conflict alerts where the duration of conflict is less than 30 seconds can reduce it further by 13%. In Covariance method also, reducing the conflict alert threshold window to 8–15 min can reduce up to 66% of False Alarms. Eliminating the conflict alerts with conflict duration of less than 30 seconds can further reduce the False Alarms by 19%. Conflict pairs where ownship is climbing are more susceptible of generating False Alarms (16%) in both the methods. Further monitoring or delayed alert of conflicts with wider convergence angles (90–180 degrees) can also reduce False Alarms and Missed Detects in both the methods.

Though the Missed Detects rate is low for both the probe methods (Fixed Threshold: 0.82% and Covariance: 0.84%) raising the lower end of the conflict alert threshold window from 8 min to 9 min can reduce the Missed Detect rate by another 52.4% and 75.1% for Fixed Threshold and Covariance method respectively. Overall, results indicate that a conflict threshold window of 9 min to 15 mins is more suitable for MTCD system instead of 8 min to 20 min and can significantly reduce the False Alarms and Missed Detects.

The performance of a given MTCD method is sensitive to parameters such as look-ahead time, surveillance and wind forecast error and conflict characteristics. We have used the best suggested parameter settings for each method; the performance may vary with different settings and may affect its performance. The performance of given MTCD methods can also be affected by some of the common problem of aircraft turning early, late, or missing turns as well as for turn and

However, both methods are susceptible to generating False Alarm and Missed Detect when ownship and intruder have wider conflict angle (90–180 degrees).
climb manoeuvres and the two methods are likely to have better or worse statistics in different conflict scenarios. In future we will be extending our work by incorporating other probe methods of MTCD (such as Conformance Bound) and other controllers based conflict detection methods such as Tactical Controller Tool (TCT). We will further our investigations by employing data mining techniques to identify intrinsic patterns, if any, in conflict characteristics that lead to Missed Detects and False Alarms.

ACKNOWLEDGMENT

This work has been co-financed by the European Organisation for the Safety or Air Navigation (EUROCONTROL) under its University Research Grant programme. The content of the work does not necessarily reflect the official position of EUROCONTROL on the matter.

REFERENCES


