

# Fast Algorithm Based on Computational Geometry for Probabilistic Aircraft Conflict Detection

Yang Yang

School of Electronic and Information Engineering, Beihang University, Xueyuan Road No. 37, Haidian District, Beijing 100191, China  
buaayangyang@buaa.edu.cn

Kai-quan Cai

School of Electronic and Information Engineering, Beihang University, Xueyuan Road No. 37, Haidian District, Beijing 100191, China  
caikq@buaa.edu.cn

Maria Prandini

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milano 20133, Italy  
maria.prandini@polimi.it

## ABSTRACT

Aircraft conflict detection plays a main role in the air traffic controller's task to maintain flight safety. Due to the impact of uncertainty on the aircraft motion, in particular, stochastic winds, the actual trajectory differs from the planned trajectory which makes aircraft conflict detection more challenging. A probabilistic approach can be less conservative and more efficient than the deterministic or the worst-case approaches. However, this approach resorting to a large number of actual trajectories is intensive computationally. In order to tackle this problem, a fast algorithm based on Computational Geometry for Probabilistic Conflict Detection (CG-PCD) is proposed combining the advantages of both deterministic and probabilistic approaches. At first place, the ellipsoidal reach sets are computed offline and adopted to describe the uncertainty affecting aircraft motion with some probabilistic guaranteeing. Then, an approximate computational geometry algorithm is introduced to determine the intersection between ellipsoids to reduce computational time. Specifically, the external ellipsoid of the Minkowski sum of two ellipsoidal reach sets are calculated analytically. Some numerical experiments are used to verify the efficacy and efficiency of the presented algorithm while comparing with a standard Monte Carlo based Probabilistic Conflict Detection (MC-PCD) algorithm. The results show that it takes about  $5.60 \times 10^{-5}$  seconds with CG-PCD instead of 4.26 hours with MC-PCD to detect a conflict instance maintaining no less than 95% probabilistic level on a standard personal computer.

## CCS Concepts

•Theory of computation → Numeric approximation algorithms

## Keywords

Aircraft conflict detection; Reach set computation; Scenario approach; Computational geometry.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

ICRAI 2017, December 29–31, 2017, Shanghai, China

© 2017 Association for Computing Machinery.

ACM ISBN 978-1-4503-5358-8/17/12...\$15.00

DOI: <https://doi.org/10.1145/3175603.3175616>

## 1. INTRODUCTION

An aircraft conflict occurs when the prescribed safe separation distance between two aircraft is violated which is one of the most dangerous events on route. The priority for flight safety of air traffic controller is to detect the potential conflict in advance and then resolve it [1], [2]. In the task of conflict detection, the contribution of the different sources of uncertainty (mainly, the stochastic winds) affecting the predicted aircraft motion cannot be neglected [3], [4], [5], [6]. Probabilistic Conflict Detection (PCD) [2], [7] thus becomes a suitable approach since it avoids the conservativeness of the worst case approach while being more effective than the deterministic approach.

The state-of-the-art PCD approaches are characterized by repeated simulations of the aircraft trajectories according to the model describing their motion and corresponding uncertainty, and the fraction of trajectories that generate a conflict is then an unbiased estimate of the probability of conflict, including Monte Carlo based PCD (MC-PCD), and its variants, such as, Markov Chain Monte Carlo or Sequential Monte Carlo based PCD [3], [4], [5], [6], [7], [8], [9], [10], [11], [12]. These approaches avoid the difficult task to analytical compute the probability of conflict in which some complex model of aircraft motion can be adopted. However, they are intensive computationally resorting to a large number of aircraft trajectory realizations, which makes the resulting PCD algorithms hardly be applied in real time.

In this paper, we present a fast algorithm based on Computational Geometry for PCD (CG-PCD) combining both the advantages of deterministic and probabilistic viewpoints. Firstly, the probabilistic reach set to account for the uncertainty (mainly, the stochastic winds) with some prescribed probabilistic guaranteeing ( $\mathcal{P} \geq 1 - \epsilon$ , where  $\epsilon$  is the violation probability) is introduced [13], [14]. In particular, the deviations of an aircraft from its planned reference trajectory are confined within ellipsoidal reach sets. As  $\mathcal{P}$  grows to 1, the size of ellipsoid grows and the probabilistic approach resembles the worst-case approach, thus becoming conservative. Note that, the computations of the parameterized reach sets with different  $\mathcal{P}$  can be performed offline based on some realizations of aircraft trajectory. Then, PCD is formulated as the determination of the interaction between two ellipsoids (the same probabilistic levels  $\mathcal{P} \geq 1 - \epsilon$  are adopted) in a deterministic form with the probabilistic guaranteeing of  $\mathcal{P}_c \geq 1 - 2\epsilon$  for the detected results. The computational geometry algorithm is presented based on the concept of Minkowski sum, where the externally approximate ellipse of the Minkowski sum is obtained analytically so as to reduce computational time. The theoretical proof of this reformulation is also given in this paper. Some numerical examples are given to show the efficacy of the proposed CG-PCD approach. The results are also compared with

MC-PCD approach using the air traffic simulator developed within the group of Prof. J. Lygeros, ETH Zürich [9].

The rest of the paper is organized as follows. Section 2 introduces the proposed CG-PCD scheme for conflict detection issue, specifically, the probabilistic reach set and the computational geometry algorithm. Section 3 describes the numerical results. Finally, some conclusions and future directions are given in Section 4.

## 2. ALGORITHM

Suppose that aircraft  $i$  and  $j$  are tracking some flight plan, and denote by  $\bar{p}_i(t)$ ,  $\bar{p}_j(t)$  their reference positions along their planned trajectories during the lookahead time horizon  $[t_s, t_d]$ , where  $t_s=0$  represents the current time instant. The aircraft future positions are uncertain due to the stochastic winds. We then denote by  $p_i^\delta(t), t \in [t_s, t_d]$ ,  $p_j^\delta(t), t \in [t_s, t_d]$  the aircraft actual positions, where  $\delta$  represents the stochastic uncertainty and takes values in the set  $\Delta$  of all possible realizations of the stochastic winds. Then, the goal of PCD is to determine the loss of safe separation  $d_s$  between the actual positions of any two aircraft with probability at least  $1-\epsilon$ , where  $\epsilon \in (0,1)$  is the violation parameter, and the  $\{t_k\}_{k=1}^{n_s}$  are time instants of  $[t_s, t_d]$ , as follows:

$$\mathcal{P}_c \left\{ \delta : \left\| p_i^\delta(t_k) - p_j^\delta(t_k) \right\| \leq d_s, k \in \{1, \dots, n_s\} \right\} \geq 1 - \epsilon \quad (1)$$

where, the probabilistic distribution  $\delta \in \Delta$  is unknown or too complex. Thus, one can resort to randomized method for an approximate solution.

In this paper, we adopt some ellipsoidal reach set  $\mathcal{E}(\bar{p}_i(t_k), S_\theta(t_k))$  and  $\mathcal{E}(\bar{p}_j(t_k), S_\theta(t_k))$  for aircraft  $i$  and  $j$  to describe the set of possible actual position with the probabilistic level  $1-\epsilon_i$  and  $1-\epsilon_j$  satisfying  $\epsilon \geq \epsilon_i + \epsilon_j$  centering on  $\bar{p}_i(t_k)$  and  $\bar{p}_j(t_k)$  respectively with the shape matrix  $A(t_k) = S_\theta(t_k)^T S_\theta(t_k)$ . Then, conflict detection is reformulated as the problem of determining the separation of two ellipsoids deterministically with the following condition:

$$\left\| \mathcal{E}(\bar{p}_i(t_k), S_\theta(t_k)) - \mathcal{E}(\bar{p}_j(t_k), S_\theta(t_k)) \right\| \leq d_s, k \in \{1, \dots, n_s\} \quad (2)$$

Note that, the probabilistic reach set can be obtained offline, the computational time of the proposed CG-PCD is thus only dependent on the latter part which does not involve computing any probability but just set intersection. A fast algorithm based on computational geometry is introduced to further reduce the computational time via the approximate analytical computation of the Minkowski sum of two ellipsoids. Here,  $d_s$  is set to 0 without losing the generality. The details of the CG-PCD is shown in the following subsections.

### 2.1 Probabilistic Reach Set

The probabilistic reach set of aircraft  $i$  is computed as the Chance-Constrained Optimization Program (C-COP):

$$\begin{aligned} & \underset{\theta \in \mathbb{R}^d}{\text{minimize}} \sum_{k=1}^{n_s} \log \det S_\theta(t_k)^{-1} \\ & \text{subject to :} \\ & S_\theta(t_k) = S_\theta(t_k)^T, k = 1, \dots, n_s, \\ & \mathcal{P} \{ \delta : p_i^\delta(t_k) \in \mathcal{E}(\bar{p}_i(t_k), S_\theta(t_k)), k = 1, \dots, n_s \} \geq 1 - \epsilon_i \end{aligned} \quad (3)$$

where the cost function is to obtain the smallest region containing all possible realizations of the aircraft trajectories except for a set of probability at most  $\epsilon_i$  with some confidence probability no less than  $1-\beta$ .

Then, the scenario approach, as proposed earlier in [15], [16] is adopted to replace the probabilistic constraint with a finite number  $N$  of deterministic constraints, which are obtained by independently extracting  $N$  "scenarios"  $\delta^{(l)}, l = 1, \dots, N$ , of the uncertainty  $\delta$ . The proper selection of  $N$  can assure the probabilistic guaranteeing [15], [16].

The scenario version of problem (3) then becomes the following convex optimization program [13], [14]:

$$\begin{aligned} & \underset{\theta_N \in \mathbb{R}^d}{\text{minimize}} \sum_{k=1}^{n_s} \log \det S_{\theta_N}(t_k)^{-1} \\ & \text{subject to :} \\ & S_{\theta_N}(t_k) = S_{\theta_N}(t_k)^T, k = 1, \dots, n_s, \\ & p_{\delta^{(l)}}(t_k) \in \mathcal{E}(\bar{p}_i(t_k), S_{\theta_N}(t_k)), l = 1, \dots, N, k = 1, \dots, n_s \end{aligned} \quad (4)$$

which is always feasible since it consists in determining the minimum area set composed by  $n_s$  ellipses, each one covering a finite number of points [17].

We adopt a 4-dimensional parametrization  $\theta \in \mathbb{R}^4$  for  $S_\theta(t_k)$ ,  $j = k, \dots, n_s$  in the reach set so that the number of optimization variables is  $d=4$  and the computational load is significantly reduced with respect to the fully parameterized case where all matrices  $S_\theta(t_k)$ ,  $j = k, \dots, n_s$  have free elements, as follows (the constant value  $C = -1.3$  is selected via numerical experiments):

$$S_\theta(t_k) = \begin{bmatrix} \theta_1 \cdot k^C + \theta_2 & \theta_3 \\ \theta_3 & \theta_4 \end{bmatrix}, k = 1, \dots, n_s \quad (5)$$

where it accounts for the case that the heading angle is  $\psi = 0^\circ$ , and  $\theta = (\theta_1, \theta_2, \theta_3, \theta_4) \in \mathbb{R}^4$ . If we set  $\theta_3 = 0$ , the axes of the ellipses are parallel to the axes of the reference coordinate system.

The shape matrix of the ellipsoid is then given by  $A(t_k) = S_\theta(t_k)^T S_\theta(t_k) = \text{diag} \left( (\theta_1 \cdot k^C + \theta_2)^2, \theta_4^2 \right)$ , and its eigenvalues are  $\zeta_1(t_k) = (\theta_1 \cdot k^C + \theta_2)^2$  and  $\zeta_2(t_k) = \theta_4^2$ . Here, the semi-axis of length  $\zeta_1(t_k)^{-1/2} = (\theta_1 \cdot k^C + \theta_2)^{-1}$  corresponds to the along-track error, whereas the semi-axis of length  $\zeta_2(t_k)^{-1/2} = \theta_4^{-1}$  corresponds to the cross-track error. Note that  $\zeta_2(t_k)^{-1/2}$  is growing as a function of time, which models the fact that the along-track error increases with lookahead time, whilst the cross-

track error term is weakly dependent on lookahead time and hence it is modeled through a constant, see e.g. [18] and the references therein to justify this choice.

Generally, for any heading angle  $\psi \neq 0^\circ$  of the aircraft  $i$ , we can simply rotate clockwise both the reference and the actual sampled trajectories  $\bar{p}_i(t_k)$  and  $p_i^\delta(t_k), k=1, \dots, n_s$ , of an angle  $\psi$  through the rotation matrix

$$R_\psi = \begin{bmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix} \quad (6)$$

so that we can still use the same parametrization of matrix  $S_\theta(t_k)$  in (5) and just modify the optimization problem (4) by replacing  $p_i^\delta(t_k)$  and  $\bar{p}_i(t_k)$  with  $R_\psi p_i^\delta(t_k)$  and  $R_\psi \bar{p}_i(t_k)$ , respectively.

## 2.2 Computational Geometry Algorithm

Given two aircraft  $i$  and  $j$  with headings  $\psi_i$  and  $\psi_j$  at some lookahead time horizon  $t_k, k=1, \dots, n_s$ , and the corresponding ellipsoidal reach sets  $\mathcal{E}(R_{\psi_i} \bar{p}_i(t_k), S_\theta(t_k))$  and  $\mathcal{E}(R_{\psi_j} \bar{p}_j(t_k), S_\theta(t_k))$  with the probabilistic levels  $1-\epsilon_i$  and  $1-\epsilon_j$ . Where, the corresponding confidence levels are set  $1-\beta_i$  and  $1-\beta_j$  respectively.

We firstly introduce the definition of Minkowski sum. Let the vector set  $\mathcal{C}_i$  and  $\mathcal{C}_j$  consisting of any possible vectors originated from the center to any point belonging to the disc of ellipsoids  $\mathcal{E}(R_{\psi_i} \bar{p}_i(t_k), S_\theta(t_k))$  and  $\mathcal{E}(R_{\psi_j} \bar{p}_j(t_k), S_\theta(t_k))$ . Then, the Minkowski sum between two ellipsoids can be written as  $\{\mathcal{C}_i \oplus \mathcal{C}_j = c_i + c_j \mid c_i \in \mathcal{C}_i, c_j \in \mathcal{C}_j\}$ .

Based on the concept of Minkowski sum, the intersection of two ellipses is equal to fact that the center of one ellipse is covered by the Minkowski sum centered on the center of another ellipse. Therefore, the key idea of computational geometry algorithm is to determine the relationship between the center of ellipse and the Minkowski sum in the Euclidean space. The theoretical proof of this statement is given as follows.

**Proof.** The sufficient necessary condition of the intersection of two ellipsoids is then transformed as:  $\mathcal{C}_i \cap \mathcal{C}_j \neq \emptyset \Leftrightarrow \bar{p}_{i,j}(t_k) \in \mathcal{C}_i \oplus \mathcal{C}_j$ , where  $\bar{p}_{i,j}(t_k)$  is the vector originated from  $\bar{p}_i(t_k) \in \mathcal{C}_i$  to  $\bar{p}_j(t_k) \in \mathcal{C}_j$ .

$\Rightarrow$ : Suppose that the intersection point is  $\bar{p}_m(t_k)$ , then,  $\bar{p}_{i,m}(t_k) \in \mathcal{C}_i$  and  $\bar{p}_{j,m}(t_k) \in \mathcal{C}_j$ . Due to the symmetry of the vector set, the vector  $\bar{p}_{m,j}(t_k) = -\bar{p}_{j,m}(t_k)$  and  $\bar{p}_{m,j}(t_k) \in \mathcal{C}_j$ . Thus,  $\bar{p}_{i,j}(t_k) = \bar{p}_{i,m}(t_k) + \bar{p}_{m,j}(t_k)$ , where  $\bar{p}_{i,m}(t_k) \in \mathcal{C}_i$ ,  $\bar{p}_{m,j}(t_k) \in \mathcal{C}_j$ , so that  $\bar{p}_{i,j}(t_k) \in \mathcal{C}_i \oplus \mathcal{C}_j$ .

$\Leftarrow$ : Suppose that the  $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$  and given any point  $\bar{p}_m(t_k) \in \mathfrak{R}^2$ , the vector  $\bar{p}_{i,j}(t_k)$  can be expressed as  $\bar{p}_{i,j}(t_k) = \bar{p}_{i,m}(t_k) + \bar{p}_{m,j}(t_k)$  and there are three cases should be discussed as follows:

(1) If  $\bar{p}_m(t_k) \in \mathcal{C}_i$ , then  $\bar{p}_{i,m}(t_k) \in \mathcal{C}_i$  and  $\bar{p}_m(t_k) \notin \mathcal{C}_j$ . Thus,  $\bar{p}_{m,j}(t_k) \notin \mathcal{C}_j$  so that  $\bar{p}_{i,j}(t_k) \notin \mathcal{C}_i \oplus \mathcal{C}_j$ .

(2) If  $\bar{p}_m(t_k) \in \mathcal{C}_j$ , then  $\bar{p}_{m,j}(t_k) \in \mathcal{C}_j$  and  $\bar{p}_m(t_k) \notin \mathcal{C}_i$ . Thus,  $\bar{p}_{i,m}(t_k) \notin \mathcal{C}_i$  so that  $\bar{p}_{i,j}(t_k) \notin \mathcal{C}_i \oplus \mathcal{C}_j$ .

(3) If  $\bar{p}_m(t_k) \in \mathfrak{R}^2 \setminus \{\mathcal{C}_i \cup \mathcal{C}_j\}$ , then  $\bar{p}_{i,m}(t_k) \notin \mathcal{C}_i$  and  $\bar{p}_{m,j}(t_k) \notin \mathcal{C}_j$ , thus,  $\bar{p}_{i,j}(t_k) \notin \mathcal{C}_i \oplus \mathcal{C}_j$ .

In a word, there is no  $\bar{p}_i(t_k) \in \mathcal{C}_i$  and  $\bar{p}_j(t_k) \in \mathcal{C}_j$  satisfying  $\bar{p}_{i,j}(t_k) \in \mathcal{C}_i \oplus \mathcal{C}_j$  under the assumption of  $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ . Thus, if there is a vector  $\bar{p}_{i,j}(t_k) \in \mathcal{C}_i \oplus \mathcal{C}_j$ , then it concludes  $\mathcal{C}_i \cap \mathcal{C}_j \neq \emptyset$ .  $\square$ .

Furthermore, due to that the Minkowski sum of two ellipses is some complex polygon should be addressed via numerical algorithm, we introduce a new ellipse  $\mathcal{E}^*(\bar{p}^*(t_k), S^*(t_k))$  as the externally approximation of the Minkowski sum in an analytical form. Here, the shape matrix for the ellipsoidal reach sets are denoted as  $A_i = R_{\psi_i}^T S_\theta(t_k)^T S_\theta(t_k) R_{\psi_i}$  and  $A_j = R_{\psi_j}^T S_\theta(t_k)^T S_\theta(t_k) R_{\psi_j}$ . The formulas of the shape matrix of the external ellipse is then given as follows (the proofs can be seen in [16], [10]):

$$A^*(t_k) = \left( (r^T A_i(t_k) r)^{1/2} + (r^T A_j(t_k) r)^{1/2} \right) \left( (r^T A_i(t_k) r)^{-1/2} A_i(t_k) + (r^T A_j(t_k) r)^{-1/2} A_j(t_k) \right) \quad (7)$$

Thus, the obtained parameters of ellipse  $\mathcal{E}^*(\bar{p}^*(t_k), S^*(t_k))$  are  $A^*(t_k) = S^*(t_k)^T S^*(t_k)$  and  $\bar{p}^*(t_k)$  can be  $\bar{p}_i(t_k)$  or  $\bar{p}_j(t_k)$ . If the following condition is satisfied, then there will be a conflict at time  $t_k$ . Note that the result is guaranteed with a probability level  $1-\epsilon_i-\epsilon_j$  and a confidence level  $1-\beta_i-\beta_j$ .

$$((\bar{p}_i(t_k) - \bar{p}_j(t_k))^T A^*(t_k) (\bar{p}_i(t_k) - \bar{p}_j(t_k))) \leq 1 \quad (8)$$

## 3. Numerical Results

We adopt the air traffic simulator described in [4] and [6], and available for download at <http://people.epfl.ch/cgi-bin/people?id=234671\&op=bio\&lang=en\&cvtlang=en>. The calculations are performed using Matlab R2013b, 64-bit, on a personal computer with an Intel (R) Core (TM) i5-6600K CPU 3.50GHz processor, 8.00 GB RAM, and Windows 7 64-bit as operating system.

### 3.1 Examples of Probabilistic Reach Set

Here, 3 different probability levels  $1-\epsilon$  with parameters  $\epsilon = 0.025, 0.050, 0.100$  are considered to describe the set of actual trajectories. The reach sets at 10 minutes predicted time instance are shown in Figure 1, 2, and 3, where the area is reduced as  $\epsilon$  grows. Moreover, as the lookahead time increases, the area grows so as to depict the growing uncertainty. Note that, the confidence parameter is set as  $\beta = 10^{-8}$  for this probabilistic reach sets.

## 3.2 Results of Computational Geometry

### Algorithm

Some conflict detection examples using CG-PCD are shown where the two aircraft are heading on to each other parallel to the X axis. In Figure 4, the two ellipsoidal reach set at the lookahead time of 8 minutes with probability levels  $1-\epsilon$  where  $\epsilon=0.025$  are separated. We can see that the reference position of aircraft at left is outside of the ellipsoidal Minkowski sum marked as the black dotted line. It means that it is conflict-free at this time instance with a probability guaranteeing more than 95%. In Figure 5, the conflict is then detected because that the reference position of aircraft at right is inside of the ellipsoidal Minkowski sum centered at the reference position of aircraft at left at the lookahead time of 11.5 minutes. This result is also guaranteed with a probabilistic level of 95%. Note that, the CG-PCD algorithm can be ran iteratively with different probabilistic reach sets so as to obtain the detected results with different probabilistic guaranteeing via using the preexisting parameters of reach sets.

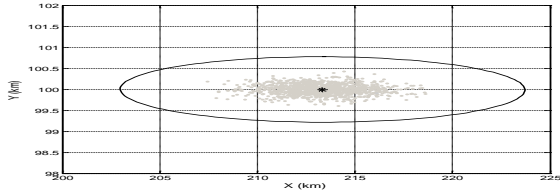


Figure 1. The reach set of  $\epsilon=0.025$  at the lookahead time of 10 minutes.

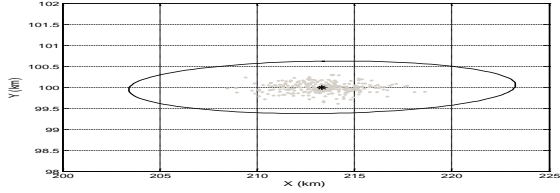


Figure 2. The reach set of  $\epsilon=0.050$  at the lookahead time of 10 minutes.

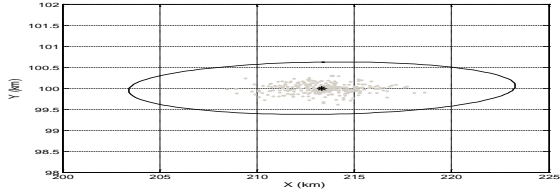


Figure 3. The reach set of  $\epsilon=0.100$  at the lookahead time of 10 minutes.

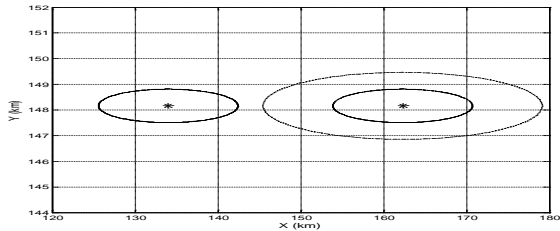


Figure 4. The conflict-free instance applied CG-PCD at the lookahead time of 8 minutes. The approximated Minkowski sum, the reach set and the reference trajectory are marked with black dotted line, black solid line, and black stars, respectively.

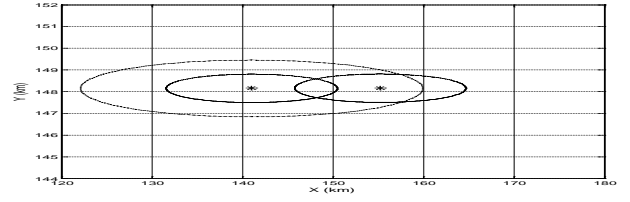


Figure 5. The conflict instance applied CG-PCD at the lookahead time of 11.5 minutes. The approximated Minkowski sum, the reach set and the reference trajectory are marked with black dotted line, black solid line, and black stars, respectively.

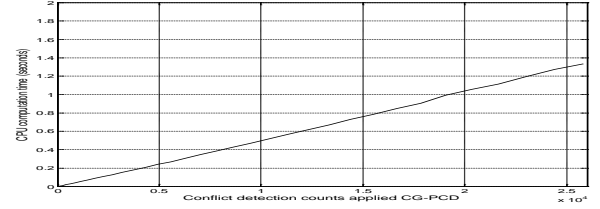


Figure 6. The CPU computational time as the function of conflict detection counts applied CG-PCD.

### 3.3 Comparison of Computational Efficiency

In order to verify the computational efficiency of the proposed algorithm, we adopt some symmetrical configurations where starting way-points of  $n$  aircraft are symmetrically distributed on a circle of radius 141.70 km centered at (148.16,148.16) and the corresponding destination way-points are on the circle as well along a lookahead time horizon  $[t_s, t_d]=[0,20]$  minutes. Reference trajectories pass through the center of the circle at the lookahead time  $t=10$  minutes in all considered conflicts. Here, the counts of conflicts are  $n_s n(n-1)/2$  and  $n_s=41$  with a sampling interval  $\Delta=0.5$  minutes.

The CPU computational time shown in Figure 6 is the average value of 100 runs of the repeated numerical experiments where the number of aircraft  $n$  grows from 2 to 36. From this plot, we can see that it takes less than 1.40 seconds for all 25000 applications of CG-PCD. Thus, it needs about  $5.60 \times 10^{-5}$  seconds for each conflict detection and the result is with a probabilistic guaranteeing of 95% and confidence of  $\beta=1-2 \times 10^{-8}$ . Note that, more efficient parallel implementations and a better performing platform could be adopted to further reduce computational time.

Furthermore, the MC-PCD is used as the alternative contrastive algorithm. Here, the required number of simulations  $N_{MC-PCD}$  with the same parameter sets ( $\epsilon=0.025, 0.050, 0.100$  and  $\beta=10^{-8}$ ) are given by the Chernoff bound of Monte Carlo

approach  $N_{MC-PCD} \geq \frac{1}{2\epsilon^2} \ln \frac{1}{\beta}$  [2] and reported in Table 1. Based

on the air traffic simulator, the CPU computational time of one actual trajectory simulation for one aircraft flying 20 minutes is 0.50 seconds by averaging 10000 times of repeated numerical experiments. Thus, we can estimate that the CPU computational time  $t_{CPU}$  by simply calculations with the parameter sets, see the Table 1 for reference. If aiming to obtain the detected results with the same probabilistic guarantee as the above one with CG-PCD, we need to generate 15291 simulations of trajectories for each aircraft and it takes about 4.26 hours which is far more than the

CPU computational time with CG-PCD. Note that, the confidence  $1-10^{-8}$  with CG-PCD here is slightly surpass, however some reach set with higher confidence (such as,  $1-10^{-9}$ ) can be adopted in CG-PCD without increasing the CPU computational time. Additionally, even if the probabilistic guarantee is lowered to (80%), it still needs about 959.19 seconds. Thus, the proposed CG-PCD has some obvious advantage to reduce the CPU computational time by combining the strengths of both probabilistic and deterministic and probabilistic viewpoints.

**Table 1. The required simulations  $N_{MC-PCD}$  for each aircraft and the estimated CPU computational time  $t_{CPU}$  for one conflict detection applied MC-PCD with different values of  $\epsilon$ .**

$\epsilon$	$N_{MC-PCD}$	$t_{CPU}$
0.025	15291	4.26h
0.050	3823	1.07h
0.100	956	959.19s

#### 4. CONCLUSIONS AND FUTURE WORKS

A fast algorithm for probabilistic conflict detection based on computational geometry was proposed in this paper, in which the probabilistic ellipsoidal reach set computed offline was adopted to describe the uncertainty affecting aircraft motion, and an approximate computational geometry algorithm was introduced to determine the intersection between ellipsoids so as to reduce the computational time while maintain some probabilistic guaranteeing. The theoretically proof of the computational geometry algorithm was also given. The results of some numerical experiments showed the efficacy and efficiency of the proposed algorithm. Based on the same computer platform, the CPU computational time for each conflict detection with CG-PCD algorithm is about  $5.60 \times 10^{-5}$  seconds to maintain a 95% probabilistic level, in contrast, it takes about 4.26 hours to apply the standard MC-PCD algorithm. In the future, the congestion detection in the multi-aircraft network with the presence of uncertainty should be addressed to enhance the flight safety in a long-term lookahead time horizon.

#### 5. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation for Young Scientists of China (No. 61401011), and National Key Technologies R&D Program of China (No.2015BAG15B01).

#### 6. REFERENCES

[1] Kuchar, J.K. and Yang, L.C. 2000. A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189.

[2] Prandini, M. and Watkins, O.J. 2005. Probabilistic aircraft conflict detection. *HYBRIDGE Deliverable n.2, Workpackage 3*.

[3] Erzberger, Paielli, R., D., Isaacson, and M. Eshowx. 1997. Conflict detection and resolution in the presence of prediction error. In 1st USA/Europe Air Traffic Management Research Development Seminars, pages 50–56, Saclay, France, June.

[4] Glover W. and Lygeros, J. 2004. A stochastic hybrid model for air traffic control simulation. *Hybrid Systems:*

*Computation and Control, Lecture Notes in Computer Science Volume 2993:372–386.*

[5] Chaloulos, G. and Lygeros, J. 2007. Effect of wind correlation on aircraft conflict probability. *Journal of Guidance Control and Dynamics*,30(6):1742–1752.

[6] Lymeropoulos and Sequential, I. 2010. Monte Carlo Methods in Air Traffic Management. Phd thesis, ETH, Department of Information Technology and Electrical Engineering.

[7] Hu, J., Prandini, M. and Sastry, S. 2005. Aircraft conflict prediction in the presence of a spatially correlated wind field. *IEEE Transactions on Intelligent Transportation Systems*, 6(3):326–340.

[8] Yang, L.C. and Kuchar, J. 1998. Using intent information in probabilistic conflict analysis. In *AIAA Guidance, Navigation, and Control Conference*, Boston, U.S.A.

[9] Lygeros, J. and Prandini, M. 2002. Aircraft and weather models for probabilistic conflict detection. In *41st IEEE Conference on Decision and Control*, Las Vegas, U.S.A.

[10] Watkins, O.J. and Lygeros, J. 2003. Stochastic reachability for discrete time systems: An application to aircraft collision avoidance. In *42nd IEEE Conference on Decision and Control*, Maui, U.S.A.

[11] Prandini, M. and Hu, J. 2009. Application of reachability analysis for stochastic hybrid systems to aircraft conflict prediction. *IEEE Transactions on Automatic Control*, 54(4):913–917.

[12] Bakker, G.J. and Blom, H. 2002. Conflict probability and increasing probability in air traffic management. In *41st IEEE Conference on Decision and Control*, Las Vegas, U.S.A.

[13] Yang, Y., Zhang, J., Cai, K., and Prandini, M. 2014. A stochastic reachability analysis approach to aircraft conflict detection and resolution. In *2014 IEEE Multi-conference on Systems and Control*, Antibes, France, October.

[14] Yang, Y., Zhang, J., Cai, K., and Prandini, M. 2017. Multi-aircraft conflict detection and resolution based on probabilistic reach sets. *IEEE Transactions on Control Systems Technology*, 25(1):309–316.

[15] Campi, M.C. and Garatti, S. 2008. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*,19(3):1211–1230.

[16] Campi, M.C. and Garatti, S. 2011. A sampling-and-discarding approach to chance-constrained optimization: Feasibility and optimality. *Journal of Optimization Theory and Applications*, 148(2):257–280.

[17] Boyd, S. and Vandenberghe, L. 2009. *Convex Optimization*. Cambridge University Press.

[18] Prandini, M., Hu, J., Lygeros J. and Sastry, S. 2000. A probabilistic approach to aircraft conflict detection. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):199–220, December.

[19] Varaiya, P. and Kurzhanskiy, A. B. 2000. On ellipsoidal techniques for reachability analysis. *Optimization Methods and Software*, 17:177–237.

[20] Kurzhanskiy, A.A. and Varaiya, P. 2007. Ellipsoidal techniques for reachability analysis of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 52(1):26–38.