A spatial division approach to aircraft route conflict

**Abstract**: Routes Conflict detection is of great importance to air traffic control. Due to previous route conflict detection algorithm has a large number of computations, high time complexity and low reliability. This paper proposes a fast route conflict detection method based on space division, dividing the space to be detected into several conflict detection units, calculating the conflict detection units to which the route passed, and predicting the distance between two subsequent routes if they intersect at same detection unit, and warning if the distance between two routes gradually decreases. The simulation results show that the algorithm has low time complexity, and high efficiency, and is suitable for parallel processing and large-scale route conflict detection, which has high application value.

**Keywords**: conflict detection, air traffic management, spatial division, geodetic coordinate system

**0 Introduction**

The risk of collision exists when the route of multiple aircraft has less horizontal distance than the safe horizontal distance and the altitude difference is less than the safe altitude distance at a given moment. Route conflict detection is an important part of air traffic control and military operational command and is significant to flight safety and mission assurance.

The first model in the field of aircraft conflict, the aircraft collision model [1], was proposed by Reich et al. in the 1960s. This model sets the airspace of an aircraft at a certain moment as a rectangular body of specified size, thus converting the conflict probability of two aircraft into the probability of conflict between a point and a rectangular body. Havel et al. proposed a cylindrical flight model [2] based on the aircraft collision model. The cylindrical flight model uses the centre of mass of the aircraft as the centre of a circle, the flight safety radius as the radius, and the height of the cylinder according to different flight altitudes. This model is more realistic than the vehicle collision model. The probabilistic analysis method of airspace safety proposed by Prielli et al. introduces a probabilistic algorithm [3] for the conflict detection of two aircraft located in the same altitude layer. The probability of one vehicle entering the safety zone of the other vehicle at a future moment is found using a probability distribution function, assuming a fixed series of coordinate point locations on the route of the two vehicles, and that both vehicles have their own constant flight speed. Fulton et al. analysed the complexity of multiple vehicles flying through the air and proposed the concept of Voronoi polygons[4], which can be used to simplify the number of vehicles and increase the speed of computation.

The route detection algorithm research has developed rapidly in recent years in China. Zhao Hongyuan et al. conducted an in-depth study on the aircraft conflict between two intersected routes [5],

proposed the concept of air conflict region, and studied the situation of intersected routes belonging to the same flight altitude layer. Based on this, the number of flight conflicts occurring per unit time for each aircraft in the airspace was calculated. Chen Chen et al. processed the flight target conflict detection with a probabilistic statistics-based algorithm [6], analyzed various factors affecting the airspace flying, and calculated and simulated the flight conflict probability based on two-dimensional Brownian motion. Through the simulation experiments of the airspace traffic control system improved by this algorithm, it is proved that this algorithm can meet the basic requirements for flight conflict detection in the airspace traffic control system. Liu Xing et al. conducted computational and simulation experiments on flight impulse detection using genetic algorithm according to domestic air control regulations [7]. However, due to the high algorithmic complexity of genetic algorithm, it is not suitable for detection in large-scale flight scenarios. If there are multiple aircraft flying at the same time, it will lead to an enormous amount of calculations. Generally speaking, many aircraft are far away from each other at the same time and will not clash, so it is not necessary to perform comparisons between all aircrafts one by one. Appropriate methods can be used to filter out aircraft pairs that do not clash at all in advance. Traditional filtering algorithms are based on 3D projection or distance judgments, these have high time complexity [3,8-10]. To reduce the computational complexity of flight conflict detection, the Delaunay theory in imaging has been used for flight conflict detection in recent years, and the Delaunay triangulation method is used to filter potential conflicts, and the time used is independent of the number of aircraft, which reduces the computational effort [11-12], and the locally updated triangulation algorithm [13 -15] can use the temporal inheritance of the position relationship between aircraft to update the grid locally, which is computationally more efficient for single time period conflict detection, and therefore is mostly used for short-time conflict alerts.

To improve the route conflict detection function in the ATC system, this paper proposes a fast route conflict detection method based on spatial division. First, the space of the area to be detected is divided into several rectangular-shaped conflict detection units, and the trajectory is divided into areas according to the conflict detection units. And, only for the traces in the same conflict detection cell, route prediction is performed to determine whether there is a risk of route conflict.

**1 Route conflict detection method**

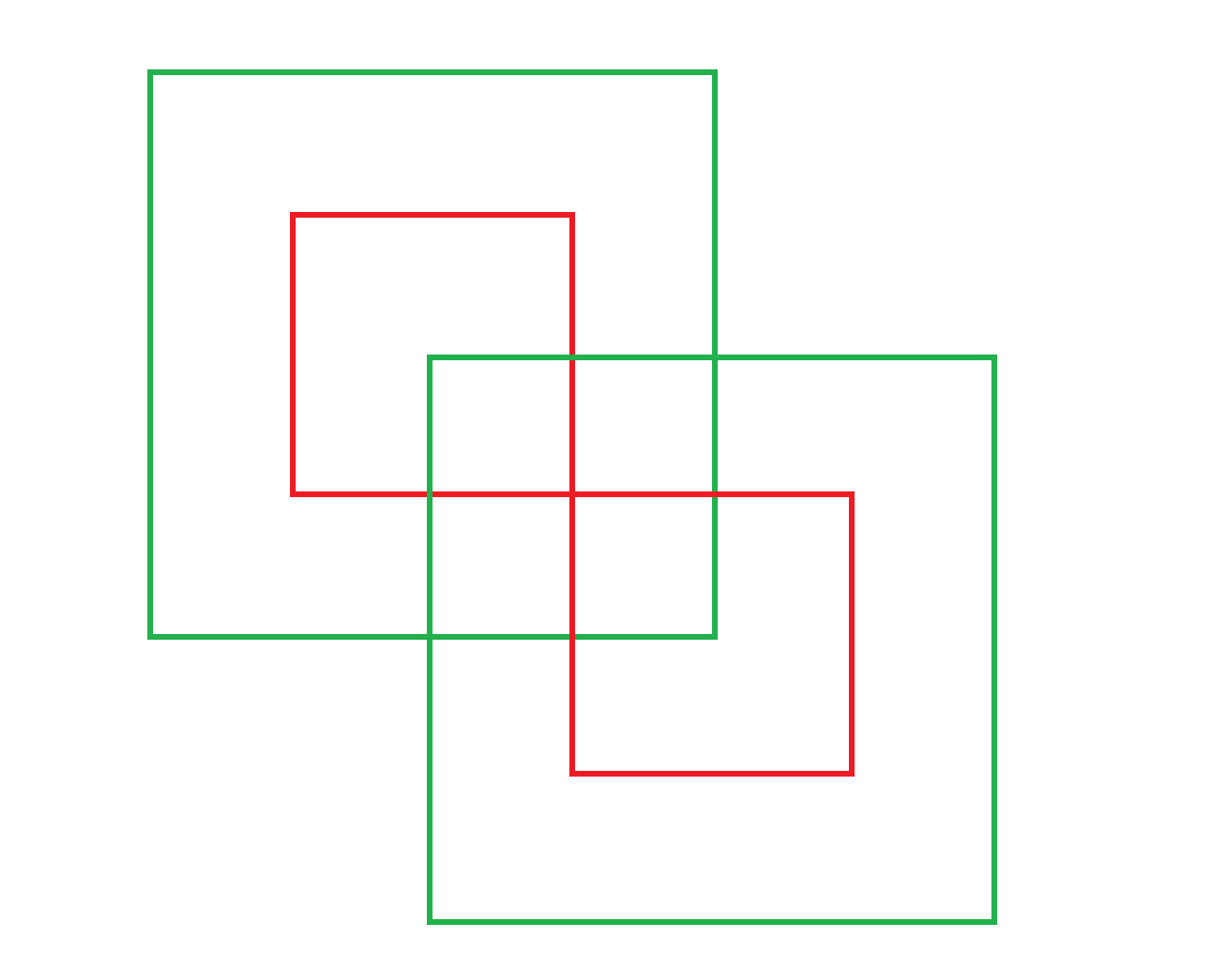
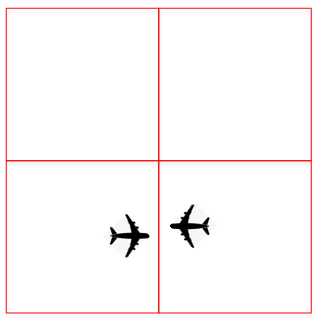
* 1. **Coordinate system selection**

Usually, both ADS-B and radar systems use a geodetic coordinate system to report air situations, but a geodetic coordinate system is not convenient to deal with spatial position relationships, therefore, in this paper, the relevant calculation will be carried out under the spatial right angle coordinate system, and the geodetic coordinates are also converted to spatial right angle coordinates first in the actual engineering application.

**1.2 Conflict detection unit division**

The air space to be detected Cs={x\_0,y\_0,z\_0,Len, Wid, Hig}, where x\_0,y\_0,z\_0 are the starting coordinates of the space, and Len,Wid,Hig are the length, width and height of the space. The space of the region to be detected is divided into numbers of rectangular conflict detection cells, each of which is represented by a number, centre coordinates, length, width and height. The conflict detection cell Cu={No,cx,cy,cz,tl,tw,th}, where No is the conflict detection cell number in the format {ijk|i,j,k∈Ν;|i|≤3;|j|=3;|z|=3} , cx,cy,cz are the center coordinates of the detection cell. tl,tw,th are the length, width, and height of the detection cell.

If the space to be detected is directly sliced into equal parts, collisions near the boundary of detection units cannot be detected, as shown in Figure 1(a). To solve this problem, the coverage of conflict detection units should overlap each other as shown in Fig. 1(b). The overlap factor ck=100% in the case of Fig. 1(b) indicates that the coverage is completely overlapped once. The overlap is 200% maximum.



Conflict detection unit

center

centereer

Figure 1(a) Figure1(b)

The starting coordinates x\_0,y\_0,z\_0 and the length, width and height Len,Wid,Hig of the detection space Cs, the length, width and height tl,tw,th of the conflict detection unit Cu, and the total number of detection units Cu\_Sum() are calculated as follows.

Formula 1 Calculate the total number of the detection units

The centre coordinates Cu\_Center(ijk) of the detection unit Cu numbered ijk is calculated as follows:

Formula 2 calculate the center coordinate of detection unit

* 1. **Track prediction**

The track prediction is a direct projection of the current position Track\_p(), according to the current heading and speed. track\_p(t) after moment t is calculated as follows:

Formula 3 route prediction

* 1. **Calculation of the detection unit**

Knowing the track position (x,y,z), use Formula 4 to calculate the detection unit the route belongs to.

Formula 4 calculate the detection unit a track belongs to

The detection unit is overlapped with each other by a overlap factor ck, which means a track could belongs to more than one detection unit. Thus Formula 4 is not sufficient to calculate all the detection units a track belongs to. We can calculate the 26 detection units that surround (i,j,k) and check whether the track belongs to any one of them

Knowing the track position (ptx,pty,ptz), the method to determine whether the track belongs to the detection unit numbered ijk is as follows:

(ca,cb,cc is the center coordinates of detection unit (a,b,c) calculated using formula 2)

If:

\*(1+ck)\*0.5

\*(1+ck)\*0.5

Then:

(ptx,pty,ptz) belongs to the detection unit (a,b,c)

Formula 5 Check whether a track belongs to

* 1. **Calculating the distance between track**

The trajectory t1(x1,y1,z1) and the trajectory t2(x2,y2,z2) are known and the distance Dis between the two tracks is calculated by:

Equation 6 Calculation of the distance between two traces

* 1. **Conflict detection calculation process**

Based on the definition of the above equations, the flow of the fast route conflict detection method is shown in Figure 2. First, initialize the space of the area to be detected , calculate the number of conflict detection units and the center coordinates of the conflict detection units (Equation 1 and Equation 2); calculate the conflict detection units to which all traces belong (Equation 4 and Equation 5); for the case that the number of traces in each conflict detection unit is greater than one, calculate the predicted position at moments t1, t2, t3⋯⋯tn, and calculate the distance between two traces at moments t1, t2, t3⋯. ⋯...tn (Equation 6), and if the distance is decreasing, an alarm message is added to the track marker to alert traffic controllers.

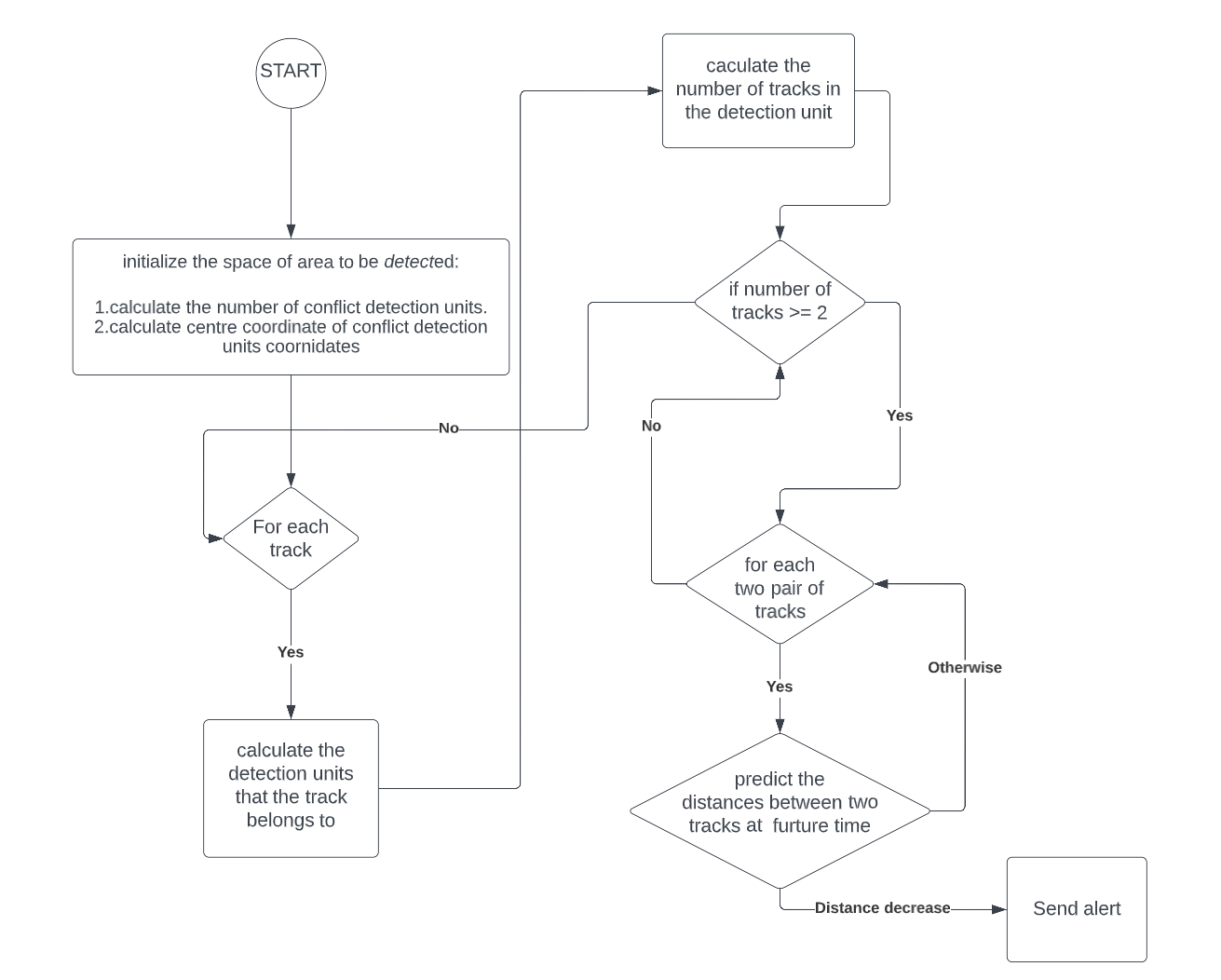


Figure 2

1. **Algorithm Time Complexity**

Suppose the number of tracks is m, the number of conflict detection units is k, and the trajectory position is predicted n times. In the most complex case, the track initialization operation is m+1 times, each time the detections units a track belongs to are calculated 27 times(3\*3\*3), then the operation of the track prediction is 27m times. The computational complexity is O(3m+ mn/2).

1. **Performance simulation experiments**