CONFLICT RESOLUTION, ONE-SHOT PROBLEM & AIR TRAFFIC CONTROL

Alfred Inselberg

IBM Scientific Center
11601 Wilshire Boulevard
Los Angeles, CA 90025-1738
aiisreal@ibm.com

&

Department of Computer Science
University of Southern California
Los Angeles, CA 90089-0782
inselber@oberon.usc.edu

The goal of Air Traffic Control is to direct aircraft safely to their destination while minimally interfering with their intended trajectories. This involves maintaining a minimum separation distance \( d \) between aircraft (considered as points), detecting conflicts (when the distance between aircraft is less than \( d \)) and resolving the conflicts according to certain maneuvering priorities. Conflict Resolution being an instance of the Asteroid Problem is an NP-Hard Problem. For two aircraft \( AC_k \) and \( AC_i \) the front and back limiting trajectories \( \hat{F}_{ik}, B_{ik} \) are found. They are the trajectories of two points, moving at the same velocity \( V_k \) and the same initial horizontal coordinate \( x_1 \) as \( AC_k \), which are tangent to a circle centered at \( AC_i \) with diameter \( 2d \), passing the circle in the front and back respectively. A parallelogram enclosing the circle \( AC_i \) is found. The representation of lines in \( R^N \) (here \( N = 4 \)) in parallel coordinates in terms of \( N - 1 \) indexed planar points is used. Relative to \( AC_i \) for each other aircraft \( AC_k \) there is a vertical interval \( I_{ik} \) consisting of points representing all the paths between the front and back limiting trajectories. The interval \( T_{ik} \) consists of the times between when the two limiting trajectories enter the parallelogram of \( AC_i \). Aircraft \( AC_i \) and \( AC_k \) are in conflict \( \iff \) the point \([1:2]_k \) representing the path of \( AC_k \) (in the \( x_1x_2 \)-plane) \( \in I_{ik} \) and entering the parallelogram of \( AC_i \) at a time \( t_k \in T_{ik} \). In that case the closest trajectory, with the same velocity, for \( AC_i \) is represented by the closest endpoint of the union of all intervals \( I_{ij} \) with the point, representing the path of \( AC_i \), \([1:2]_k \in I_{iy} \) and \( t_k \in T_{ij} \). This trajectory is free of conflicts with all other aircraft. The time available needs to be matched with the appropriate maneuvers for feasibility. The algorithm is constrained to place the aircraft on a new conflict-free trajectory only if it is within \( d \) distance from the original. Any convex shape for the protected airspace can serve just as well. To handle the problem of high complexity the algorithm operates with rules in 3 levels. The first two levels involve one initial maneuver (the first level without and the second level with some form of recursion) per aircraft while the advanced rules permit an arbitrary number of maneuvers. At present the complexity for level 1 rules is \( O(N^2 \log N) \), level 2 \( O(N^4 \log N) \) and 3 \( O(qN^4 \log N) \) here \( N \) is the total number of aircraft and \( q \) is the maximum allowable number of maneuvers. The algorithm was checked on some FAA supplied complex scenarios. The algorithm is parallelizable and certain other extensions (e.g. velocity changes) look promising.

For a set of moving spheres in a space of arbitrary dimensionality, the search for a trajectory of a point moving with constant velocity which will hit all the spheres was recently shown to be in an NP-complete problem (One-Shot Problem). In our case this occurs when the intersection of all the vertical intervals i.e. \( \cap I_{ik} \neq \emptyset \), any point in this intersection representing the trajectory of a point travelling with the same velocity as \( AC_i \) and which is also time compatible. Resolution methods have been proposed which resolve the conflicts subset by subset, leading to backtracking and very high complexity. Using solutions for the one-shot problem found in this way, it was shown that such partial resolution schemes can lead to worse conflicts than the ones they resolved in addition to posing some fundamental difficulties in program proving; hence the preference for the global resolution method proposed here. "One-shot patterns" also look very useful in understanding and classifying inherently dangerous configurations.