

Automated Conflict Resolution for Air Traffic Control

By

Heinz Erzberger

NASA Ames Research Center

Abstract

The ability to detect and resolve conflicts automatically is considered to be an essential requirement for the next generation air traffic control system. While systems for automated conflict detection have been used operationally by controllers for more than 20 years, automated resolution systems have so far not reached the level of maturity required for operational deployment. Analytical models and algorithms for automated resolution have been studied for many years; however, the algorithms have generally not been tested under realistic traffic conditions to demonstrate that they can handle the complete spectrum of conflict situations encountered in actual operations.

The resolution algorithm described in this paper was formulated to meet the performance requirements of the Automated Airspace Concept (AAC). The AAC, which was described in a recent paper [1], is a candidate for the next generation air traffic control system. The AAC's performance objectives are to increase safety and airspace capacity and to accommodate user preferences in flight operations to the greatest extent possible. In the AAC, resolution trajectories are generated by an automation system on the ground and sent to the aircraft autonomously via data link. The algorithm generating the trajectories must take into account the performance characteristics of the aircraft, the route structure of the airway system, and be capable of resolving all types of conflicts for properly equipped aircraft without requiring supervision and approval by a controller. Furthermore, the resolution trajectories should be compatible with the clearances, vectors and flight plan amendments that controllers customarily issue to pilots in resolving conflicts. The algorithm described herein, although formulated specifically to meet the needs of the AAC, provides a generic engine for resolving conflicts. Thus, it can be incorporated into any operational concept that requires a method for automated resolution, including concepts for autonomous air to air resolution.

The algorithm described herein generates resolution trajectories by a multi-step iterative process involving a Resolution Generator (RG), a 4D trajectory Synthesizer (TS) and a Conflict Detector (CD). The Resolution Generator orchestrates the resolution process and performs analytical and logical function required for generating resolutions. As the first step the RG determines the type category of the conflict submitted by the CD and then determines the set of alternative resolution maneuvers appropriate for this type category. Next, the RG assigns priority rankings to the alternative maneuvers. Highest priority is given to the maneuver that minimizes delay or, if delay is not a significant factor, conforms to strategies and procedures controllers would typically use to resolve a similar type of conflict. In addition to generating alternative resolutions, the RG also includes rules for choosing which of the two aircraft in the conflict pair should initially be given preference in performing the resolution. The set of resolutions generated by the RG serves as a reservoir of alternative resolutions from which to choose when a particular preferred resolution fails to resolve a conflict or is found to be deficient for any of several other reasons.

In the general case, the set of maneuvers the RG generates to provide alternatives for resolving a conflict includes all three major types of maneuvers, horizontal, vertical and speed. Furthermore, within each maneuver type, the algorithm generates subsets of alternative maneuvers by cycling through sets of resolution parameters for each major type. The

Resolution Generator has been implemented as a software process written in the JAVA language. In addition to containing the analytical formulas for calculating certain maneuver parameters, the RG provides a repository of knowledge, methods and rules controllers use to resolve the various types of conflicts encountered in en route airspace.

Let it now be assumed that the RG has received a new conflict pair to resolve and, upon analysis of the conflict, has determined the preferred aircraft to maneuver, the preferred maneuver type and subtype and the parameters of the selected maneuver. The crucial next step in the resolution process is to generate the unique 4D trajectory corresponding to this information. This function is performed by a complex algorithm referred to as a 4D trajectory synthesizer (TS). It uses detailed models of aircraft performance and operational procedures, atmospheric conditions, including winds aloft, to generate 4D trajectories by integrating the aircraft equations of motion along a specified route.

At NASA Ames, two software implementations of TS suitable for this application are available. A TS designed for real time use is a key software process within the Center-TRACON Automation System (CTAS), which comprises a set of automation tools for air traffic controllers [2]. A TS is also incorporated in the Advanced Concepts Evaluations System (ACES), which is a non-real time system designed specifically for simulating advanced air traffic control concepts and traffic flows [3]. Because of its comprehensive simulation capabilities, ACES and its embedded TS were chosen as the implementation and evaluation platform for the AAC's automated resolution function.

The RG thus sends the parameters calculated for the initially selected maneuver, also called trial resolution, to the TS in ACES, which then attempts to synthesize a 4D trajectory for the trial resolution. Occasionally, the TS may fail to produce a trajectory. In the event a failure is encountered, the TS sends an appropriate diagnostic message back to the RG. The RG responds by selecting the next-in-priority resolution maneuver and again sends the trial resolution parameters to the TS for another attempt to synthesize a trajectory.

If the TS succeeds in generating a 4D trajectory, it is sent to the Conflict Detector in order to check it for conflicts. Both ACES and CTAS contain software processes for conflict detection. The CD performs the conflict check by comparing the trial resolution trajectory against the 4D trajectories of all other aircraft in the airspace of interest. New conflicts can arise along a trial resolution trajectory, since this trajectory is based on a trial resolution maneuver selected to resolve only the initial conflict and without taking into account the potential for a new conflict with a third aircraft.

If the trial resolution trajectory is free of conflicts for the specified resolution time horizon, the RG promotes the trial resolution trajectory to the status of actual resolution trajectory. The ground system can now uplink this trajectory to the conflict aircraft while also updating the data base of currently valid trajectories.

If the trial resolution trajectory is found to have conflicts within the specified resolution time horizon, the RG will pick the next-in-priority trial resolution maneuver and send it to the TS for synthesis of another trajectory. This iterative process continues until either an acceptable resolution trajectory is found or the reservoir of available trial resolution maneuvers is exhausted. The sequence of steps and the top level decision logic of the algorithm are shown in the attached figure. If no resolutions are found, the RG has additional options to extend the search for resolutions, which will be described in the paper.

The resolution algorithm described above has been implemented for development and performance evaluation in ACES, which is capable of simulating the traffic flow from gate to

gate for the U.S. national airspace system. The Cleveland Center airspace was chosen as the test environment for the algorithm because it has one of the busiest and most complex air traffic environments in the national airspace system. The performance of the resolution algorithm is being evaluated for three levels of traffic density in this airspace: current traffic level, twice current level, and 3 times current level. Although the evaluation is still in progress, initial results indicate that the resolution algorithm is able to handle the conflicts detected at the current level of traffic and also shows promise in handling traffic up to 3 times current density.

The paper will describe the design of the Resolution Generator and analyze its performance in handling the full spectrum of conflict types encountered in Cleveland Center traffic simulated in ACES.

References

1. Erzberger, Heinz; Transforming the NAS: The Next Generation Air Traffic Control System; Proceedings of the International Congress of the Aeronautical Sciences (ICAS), August 30, 2004, Yokohama, Japan.
2. CTAS website: <http://www.ctas.arc.nasa.gov/>
3. ACES website: <http://vams.arc.nasa.gov/activities/aces.html>

Flow Chart for Resolution Algorithm

