

# DEVELOPMENT OF A CLOSED-LOOP TESTING METHOD FOR A NEXT-GENERATION TERMINAL AREA AUTOMATION SYSTEM

John E. Robinson III and Douglas R. Isaacson  
NASA Ames Research Center, Moffett Field, California

## Abstract

A next-generation air traffic decision support tool, known as the Active Final Approach Spacing Tool (aFAST), will generate heading, speed and altitude commands to achieve more precise separation of aircraft in the terminal area. The techniques used to analyze the performance of earlier generation decision support tools are not adequate to analyze the performance of aFAST. This paper summarizes the development of a new and innovative fully closed-loop testing method for aFAST. This method, called *trajectory feedback testing*, closes each aircraft's control loop inside of the aFAST scheduling algorithm. Validation of trajectory feedback testing by examination of the variation of aircraft time-of-arrival predictions between schedule updates and the variation of aircraft excess separation distances between simulation runs is presented.

## 1 Introduction

The United States' air traffic system has experienced significant growth during the past twenty years. This growth has resulted in substantial increases in delays at nearly every major airport. However, environmental and geographic constraints limit the opportunities to increase system capacity by building new airports or adding new runways at existing airports. Therefore, the National Aeronautics and Space Administration (NASA) and the Federal Aviation Administration (FAA) have worked to develop several decision support tools (DSTs) to assist the air traffic manager and controller workforces. These tools evaluate the air traffic situation in real-time to enable more efficient management and control of the traffic with increased safety and reduced workload.

The sophistication of air traffic DSTs has advanced considerably throughout the past decade. First-generation DSTs increased the human's efficiency by simply providing very accurate predictions of the future traffic situation. These tools provided predictive information, such as times of arrival and sector counts. Second-generation DSTs further increased the human's efficiency by augmenting the accurate predictions of the traffic situation with an efficient schedule to manage the traffic. These tools provided "strategic" traffic management advisories, such as meter fix crossing times, runway assignments and landing sequences.<sup>1,2</sup> In the near future, third-generation DSTs will further increase the human's efficiency by providing trajectories that meet the desired schedule and maintain the

desired aircraft separation. These tools will provide "tactical" traffic control advisories, such as heading, speed and altitude commands.<sup>3,4</sup>

As the sophistication of air traffic DSTs has advanced, the methods for testing these DSTs have had to mature at a similar pace. First-generation DSTs were tested by simple statistical comparisons of their predictions of the future traffic situation with the actual traffic situation that arose. Similarly, second-generation DSTs were tested by online and offline expert evaluation of the proposed schedules and human-in-the-loop simulations which used the proposed schedules to manage the traffic situation. Both of these types of DSTs could be suitably tested by analyzing the open-loop or static stability of the system. However, third-generation DSTs, with their tactical traffic control advisories, will require more in-depth testing methods. These methods must not only capture how the system responds to the evolving traffic situation, but also how the traffic situation progresses when aircraft follow the system's trajectories. Thus, third-generation DSTs can only be tested by analyzing the closed-loop or dynamic stability of the system. This paper discusses the development of a fully closed-loop testing method suitable for third-generation DSTs.

## 2 Background

In response to growth in the amount of air traffic and increased terminal area delays, NASA, in coordination with the FAA, has been researching and developing air traffic DSTs for nearly twenty years. This effort has resulted in the development of a suite of DSTs called the Center-TRACON Automation System (CTAS).<sup>5</sup> One CTAS DST, the Final Approach Spacing Tool (FAST), is designed to provide terminal air traffic controllers with computer-generated traffic management and traffic control information that safely increases the landing rate at an already busy airport. The original concept of FAST contained four functional advisories: runway assignments, landing sequences, and heading and speed commands.<sup>6</sup> The updated concept of FAST now includes altitude commands, as well. In order to expedite the operational deployment of FAST, initial research concentrated on a subset of this complete functionality. This core capability, known as Passive FAST (pFAST), encompassed only the so-called *passive* advisories: runway assignments and landing sequences.<sup>2</sup> Current research is now focused on the further

development of the remaining functionality. This additional capability, known as Active FAST (aFAST), includes the so-called *active* advisories: heading, speed and altitude commands.<sup>3</sup>

### 2.1 aFAST Scheduling Algorithm

The aFAST scheduling algorithm is a cycle-driven process that systematically orders and merges aircraft from their current positions to their runway thresholds along an appropriate series of flight segments. This scheduling process consists of two fundamental but integrated decisions: sequencing and conflict resolution.<sup>7</sup> The sequencing algorithm uses fuzzy reasoning to determine the relative sequence of each pair of aircraft that share one or more flight segments.<sup>8</sup> Meanwhile, the conflict resolution algorithm uses decision trees of air traffic control tactics to resolve all predicted conflicts.<sup>9</sup> Through schedule iteration, solution trajectories are generated that are designed to be conflict-free with minimal excess separation between aircraft. Thus, each aircraft's solution trajectory reflects its earliest time of arrival while maintaining proper separation from all surrounding aircraft.

The essential foundation of the aFAST scheduling algorithm is a trajectory synthesis engine that produces 4-dimensional (4D) trajectories by combining horizontal routes with appropriate airspeed and altitude restrictions, atmospheric weather predictions, and aircraft performance models.<sup>10</sup> Each 4D trajectory represents a kinematic solution of the aircraft's equations of motion assuming standard turn rates (~3 degree per second), accelerations (~1 knot per second) and flight path angles (~1000 feet per 3 nautical miles).

### 2.2 aFAST Graphical User Interface

As mentioned earlier, aFAST provides three types of traffic control advisories: heading, speed and altitude commands. These advisories are simply reflective of each aircraft's underlying conflict-free solution trajectory. Heading advisories reflect changes in the aircraft's horizontal route, speed advisories reflect the positioning of airspeed adjustments and altitude advisories reflect the positioning of altitude adjustments. The active advisory information is provided to the air traffic controllers by means of their standard planview radar display.<sup>11</sup> Textual information, indicating the extent of each heading, speed or altitude adjustment, is displayed in the aircraft's full data block (FDB). Additional graphical symbology, indicating the location where the heading, speed or altitude adjustment should be commanded, is drawn on the planview map. Figure 1 shows an example of several aircraft with heading, speed and altitude advisories. In this example, AAL1472 will be instructed to "turn right to heading 090 degrees" at the blue diamond, EGF124 will be instructed to "reduce speed to 170 knots" at the orange dot and AAL2084 will be

instructed to "descend and maintain 2,300 feet" at the yellow square.

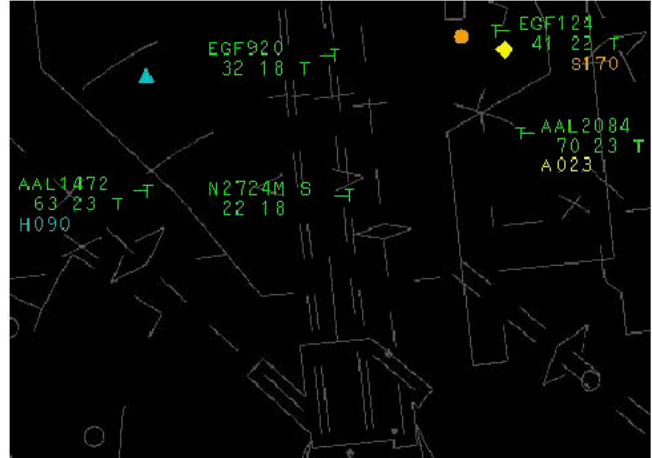


Figure 1 aFAST Graphical User Interface Example

### 2.3 Previous Testing Methods

In order to best understand why a fully closed-loop testing method is needed for aFAST, it is necessary to briefly review how earlier DSTs, like pFAST, were tested. The most common method that was used for pFAST testing is called *shadow testing* or *shadowing*. Shadowing refers to the evaluation of a DST's performance while it operates open-loop (i.e., the air traffic controllers are not using its advisories to manage or control the traffic situation). During shadow tests, the DST can be generating advisory information for live, recorded or even simulated traffic scenarios. Both objective and subjective evaluations of the system's performance can be made using the results of shadow testing. For example, objective evaluation of the pFAST landing sequences measured the cumulative accuracy of the predicted landing sequences as a function of time until touchdown for all aircraft. Similarly, subjective evaluation of the pFAST runway assignments used expert air traffic controllers and domain experts to assess the acceptability of the landing runway assigned to each aircraft. Two other methods used for pFAST validation were human-in-the-loop simulations and operational tests. Unlike shadowing, these methods evaluate a DST's performance while it operates closed-loop with air traffic controllers providing the feedback path by verbally issuing commands to the pilots. As a result, both of these methods allow more in-depth and accurate analyses of the system's actual performance to be made.

### 3 Motivation for Fully Closed-Loop Testing

Ideally, a testing method should be able to evaluate the aFAST's performance, under a wide range of operating conditions, in an experimentally controlled environment, at any point during the development process. Unfortunately, the existing testing methods, namely shadow tests, human-in-the-loop simulations and operational evaluations, do not sufficiently satisfy these requirements.

Shadow testing cannot be used to analyze aFAST's performance, because it does not capture the dynamic interaction of the system and the traffic situation. The open-loop nature of the shadow tests allows the traffic situation to affect the system, but does not allow the system to affect the traffic situation. Open-loop testing was appropriate for analyzing pFAST's performance since its runway assignments and landing sequences were strategic advisories. Therefore, their initial accuracy and their overall responsiveness to the evolving traffic situation could be evaluated even when the aircraft were not following runway assignments or landing sequences. Open-loop testing is inappropriate for analyzing aFAST's performance since its heading, speed and altitude commands are tactical advisories based upon the underlying trajectory predictions. The accuracy of the aFAST trajectories can only be analyzed when the aircraft are following the heading, speed and altitude advisories.

Human-in-the-loop simulations and operational evaluations can still be used to analyze aFAST's performance, but both methods have several major limitations. In particular, they are not able to expose the system to a wide range of traffic conditions, and they cannot be used to evaluate the system during the early stages of its development process. High fidelity traffic scenarios for human-in-the-loop simulations are painstakingly difficult to create, and operational evaluations typically cover only a very limited set of traffic conditions. Similarly, both methods must obviously be conducted in real-time rather than fast-time. As a result, physical time constraints further limit the number of testable traffic conditions. Also, both methods involve air traffic controllers and pilots and thus require relatively mature algorithms to provide reasonable heading, speed and altitude advisories. Otherwise, human-in-the-loop simulations will not be effective and operational evaluations would jeopardize safety.

Lastly, no existing testing method can systematically isolate and analyze aFAST's individual sub-elements. The aFAST scheduling process consists of many inter-dependent steps, such as state estimation, conformance monitoring, sequencing, conflict prediction, conflict resolution and advisory extraction. During shadow tests, human-in-the-loop simulations and operational tests, neither these sources of uncertainty nor the scheduling decisions themselves, can be experimentally controlled. Therefore, these testing methods do not provide the best mechanism for understanding how the operation of one particular sub-element affects aFAST's overall operation

The limitations of the existing testing methods have motivated the development of a new and innovative technique to analyze aFAST's performance while it operates fully closed-loop. During fully closed-loop operation, the algorithm simulates the flight of aircraft along the system's solution trajectories. Since fully closed-

loop testing does not involve air traffic controllers or pilots, testing can be conducted in fast-time. Therefore, a greater number of live, recorded and simulated traffic scenarios can be analyzed by fully closed-loop testing than by other methods. Furthermore, during fully closed-loop testing, the sources of uncertainty and the scheduling decisions can be tightly controlled, even predetermined. For example, the sequencing decisions can be preset while various conflict resolution strategies are compared; or the conflict resolution strategies can remain fixed while different sequencing heuristics are evaluated. Clearly, fully closed-loop operation provides a powerful testing method that can be used throughout the entire aFAST development process.

#### **4 Definition of Fully Closed-Loop Testing**

The fundamental purpose of fully closed-loop testing of aFAST is to verify that one necessary, but not sufficient, condition is satisfied. aFAST must maintain the desired separation when all aircraft fly exactly as the system expects them to fly. While this condition may appear self-evident, it has proven difficult to guarantee such behavior in earlier generation DSTs. The verification of this necessary condition has resulted in the development and implementation of an innovative and fully closed-loop testing method for aFAST. This method, called *trajectory feedback testing*, closes the each aircraft's control loop inside of the aFAST scheduling algorithm. This feedback path allows aircraft to be automatically "flown" along their solution trajectories.

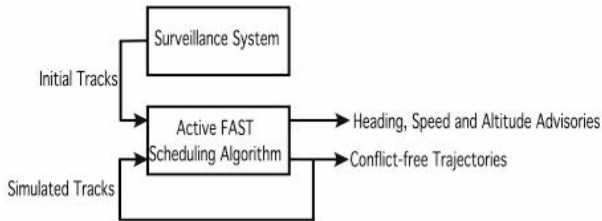
##### 4.1 Trajectory Feedback Testing:

The fully closed-loop testing method, known as trajectory feedback testing, is meant to verify that aFAST maintains the desired separation when all aircraft follow the system's solution trajectories. During this mode of operation, aFAST simulates a perfect environment completely free of errors in state estimation, trajectory prediction and controller/pilot conformance. This is accomplished by creating a feedback path between each aircraft's output (i.e., a 4D trajectory) and the system's input (i.e., a surveillance track) inside of the aFAST scheduling algorithm. For flexibility, trajectory feedback testing can be activated either manually on an aircraft-by-aircraft basis or automatically based upon the aircraft's flight segment, sector or distance from touchdown.

When trajectory feedback testing is activated for a particular aircraft, aFAST uses simulated tracks produced from its solution trajectories, rather than actual tracks received from an external source of surveillance information. Figure 2 shows a block diagram of the trajectory feedback testing process. The process begins when a surveillance track is used to produce a solution trajectory for an aircraft. This solution trajectory is then used to generate a simulated track in lieu of the aircraft's next surveillance track. The process continues when this

simulated track is used to produce a new solution trajectory during the next schedule update. The schedule, solution trajectories and simulated tracks are recomputed every five seconds. The process finishes when the aircraft reaches the end of its solution trajectory.

Figure 2 Block Diagram of Trajectory Feedback Testing



During trajectory feedback testing, an aircraft follows its predicted trajectories exactly. The aircraft is not, however, following a single trajectory. Rather, the aircraft follows each predicted trajectory for only one schedule update interval, at which point a new trajectory is calculated based on its new simulated track. The importance of this distinction is that trajectory feedback testing truly reflects the dynamic interaction of aFAST and the traffic situation in an error-free environment.

#### 4.2 Discussion of Trajectory Feedback Testing

One important limitation of trajectory feedback testing is that its results are only as realistic as the solution trajectories themselves. If the solution trajectory is physically unrealistic (e.g., exceedingly high bank angles or unreasonable rates of ascent or descent), the simulated tracks will accurately model a trajectory that would be physically unachievable. Furthermore, if the solution trajectory is operationally unrealistic (e.g., excessive maneuvers), the simulated flight paths will accurately model a flight path that would be operationally unacceptable. Therefore, trajectory feedback testing cannot be used to 1) independently validate the accuracy of underlying aircraft dynamics models, or 2) validate the usability and suitability of the aFAST advisories themselves. However, trajectory feedback testing can be used to verify that aFAST maintains the desired separation when all aircraft behave exactly as expected.

### 5 Validation of Fully Closed-Loop Testing

Before using trajectory feedback testing to analyze the performance of the aFAST system, it was necessary to validate the closed-loop method itself. The purpose of this validation was to determine the consistency and repeatability of the closed-loop results. This section summarizes the validation of the trajectory feedback testing method.

#### 5.1 Validation Traffic Scenario

While the aFAST scheduling algorithm is site independent, it does require a sophisticated airspace database to define the aircrafts' trajectories. An extensive airspace database of the Dallas/Fort Worth Terminal Radar Approach Control (DFW TRACON), called D10, was developed during countless hours of human-in-the-loop simulation and offline testing of pFAST. This same D10 airspace database was used to validate the trajectory feedback testing method.

The traffic scenario consisted of 42 aircraft entering the D10 airspace at an approximate rate of 55 aircraft per hour. The traffic scenario, representative of a portion of the D10 "noon balloon" arrival push, contained a realistic mix of aircraft types from D10's five primary arrival metering fixes. All aircraft were assigned to Runway 17C at the Dallas/Fort Worth International Airport (DFW). Therefore, the traffic load was high for a single runway operation. For comprehensive analysis of the aFAST system, realistic multiple runway scenarios will be necessary. However, for the validation of the closed-loop testing method, a single runway scenario was considered sufficient.

#### 5.2 Estimated Time-of-Arrival (ETA) Variability

A fundamental premise of the trajectory feedback testing method is that reliable closed-loop operation can be achieved by repeatedly using an aircraft's solution trajectory to produce its next simulated track. In order to verify that the trajectory feedback testing method satisfied this assumption, the variation of aircraft time-of-arrival predictions between schedule updates was examined.

The peak-to-peak variation and the overall change of each aircraft's ETA was analyzed during its simulated flight from the arrival metering fix to the runway threshold. The peak-to-peak variation was defined as the difference between the aircraft's earliest ETA and its latest ETA. The overall change was defined as the difference between the aircraft's initial ETA and its final ETA (i.e., its landing time).

In order to eliminate coupling between aircraft, conflict resolution was not performed and the aircraft flew unconstrained (fastest) trajectories. Ideally, the ETA would remain constant since each aircraft's unconstrained trajectory should remain steady and each aircraft should be following this trajectory perfectly. However, in practice, the ETA will not remain constant for several reasons. First, aFAST assumes each aircraft maintains its current indicated airspeed and flight level until it encounters the next speed or altitude restriction. As a result, any errors related to the trajectory synthesis scheme will inevitably propagate from the simulated track along some part of the aircraft's next trajectory. Second, aFAST defines the remaining portion of

each aircraft's unconstrained trajectory separately for each flight segment. Therefore, small discontinuities in the definition of the unconstrained trajectory can be encountered as the aircraft transitions between flight segments.

Figure 3 shows the ETA time history for one typical aircraft's simulated flight during trajectory feedback testing. The peak-to-peak variation of its ETA was only 0.56 seconds. Also, the overall change of its ETA was only 0.21 seconds. Several key features of this example should be noted. First, there were small discontinuities in the ETA at t=11:39:20 and t=11:40:50 where the aircraft transitioned between different flight segments. These discontinuities contributed more than one-third ( $0.21/0.56 = 38\%$ ) of the peak-to-peak variation. Fortunately, slight modifications to the trajectory's turn construction logic will eliminate most of this variation. Second, aside from the turn discontinuities, the ETA slowly drifted between simulation time, t=11:35:30 and t=11:37:30. This drift occurred during a constant speed/constant altitude trajectory segment and was presumably due to the finite precision of the trajectory's numerical integration scheme. The cumulative results for all 42 aircraft were similar. The ETA peak-to-peak variations had a mean of 0.41 seconds and a standard deviation of 0.15 seconds; the ETA overall changes had a mean of 0.17 seconds and a standard deviation of 0.10 seconds.

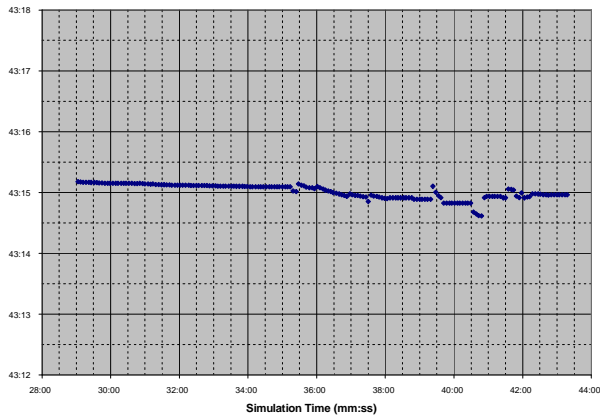


Figure 3 Time History of Typical ETA Results

### 5.2 Excess Separation Distance Variability

Another fundamental premise of the trajectory feedback testing method is that its results can be used to analyze the operation of the aFAST system. In order to verify that the trajectory feedback testing method also satisfied this assumption, the variation of the aircraft separation distances across simulation runs was examined.

The statistical variation of the excess separation distance at the runway threshold was analyzed for each pair of aircraft. The excess separation distance was defined as the

difference between the actual intrail separation distance and the applicable wake turbulence separation standard.

Unlike the previous analysis of ETA variability, conflict resolution was performed and the aircraft flew constrained (conflict-free) trajectories. The sequencing logic used a predefined landing order and the conflict resolution used an iteration tolerance of 0.1 seconds. Ideally, there would be no excess separation distance variation between repeated runs since each aircraft's initial conditions should remain constant and each aircraft should be fixed in the landing sequence. However, in practice, the coupling between aircraft due to conflict resolution, combined with ETA variability, will cause the excess separation distances to vary between runs.

Figure 4 shows the histogram of one typical simulation run's excess separation distances during trajectory feedback testing. Several key features of this example should be noted. For this particular simulation run, the excess separation distances had a mean of 0.0066 nautical miles (approximately 0.15 seconds) and a standard deviation of 0.021 nautical miles (approximately 0.49 seconds). The excess separation distance variation is approximately four times larger than the ETA variation. This magnification is due to the coupling of aircraft through the conflict resolution process. The cumulative results for 10 simulations were similar. Across all 10 simulations, the mean excess separation distances varied by less than 0.003 nautical miles (approximately 0.06 seconds) nautical miles.

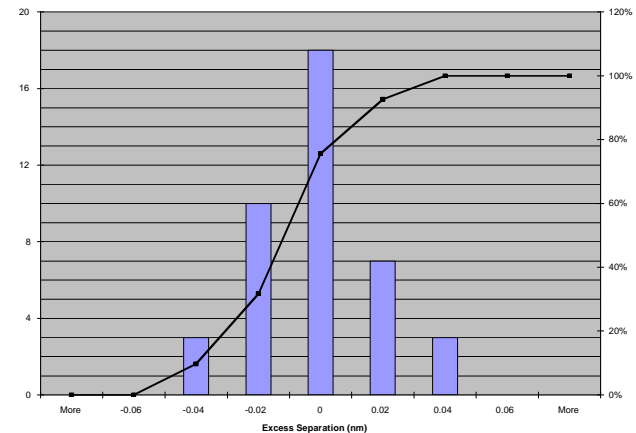


Figure 4 Histogram of Typical Excess Separation Distances

These results clearly show that trajectory feedback testing will be an effective method for analyzing aFAST's performance. In the noisy environment of the real-world, the most skilled air traffic controllers are able to achieve a mean excess separation of 0.25-0.50 nautical miles without assistance from active advisories. The trajectory feedback testing results, in an error-free environment, are more than 25-50 times more precise than these manual results. Therefore, the results achieved during trajectory feedback

testing can, in fact, serve as the baseline for ideal aFAST performance.

### 6 Concluding Remarks

An innovative and fully closed-loop testing method has been developed for a next-generation terminal area automation system, known as aFAST. This method, called trajectory feedback testing, closes each aircraft's control loop inside of the aFAST scheduling algorithm by using the aircraft's solution trajectories to generate simulated tracks.

The trajectory feedback testing method was validated by examining the variation of aircraft ETAs and excess separation distances during closed-loop operation. The reliability of the aFAST trajectories resulted in a mean ETA peak-to-peak variation of only 0.41 seconds during each aircraft's unconstrained flight. Similarly, the reliability of the aFAST schedule resulted in a mean excess separation of 0.0066 nautical miles during each aircraft's constrained flight.

Trajectory feedback testing allows a greater number of live, recorded and simulated traffic scenarios to be analyzed in fast-time, since it does not involve air traffic controllers and pilots. It also enables the sources of uncertainty and the scheduling decisions to be experimentally controlled. Thus, trajectory feedback testing provides a powerful method that can be used throughout the entire aFAST development process.

### References

- <sup>1</sup> Swenson, H.N., T. Hoang, S. Engelland, D. Vincent, T. Sanders, B. Sanford, and K. Heere (1997). "Design and Operational Evaluation of the Traffic Management Advisor at the Fort Worth Air Route Traffic Control Center." 1st USA/Europe Air Traffic Management R&D Seminar, Saclay, France.
- <sup>2</sup> Davis, T.J., D.R. Isaacson, J.E. Robinson III, W. den Braven, K.K. Lee, and B. Sanford (1997). "Operational Field Test Results of the Passive Final Approach Spacing Tool." IFAC 8<sup>th</sup> Symposium on Transportation Systems, Chania, Greece.
- <sup>3</sup> Titan Systems Corporation, SRC Division (2001). "Active Final Approach Spacing Tool (aFAST) General Description, Operational Concept for ATM – Year 2002 Update." Report No. NAS2-98005.
- <sup>4</sup> Johnson, C.W., D.R. Isaacson and K.K. Lee (1999). "Expedite Departure Path (EDP) Operational Concept." MIT Lincoln Laboratory Report No. NASA/A-3.
- <sup>5</sup> Erzberger, H., T.J. Davis and S.M. Green (1993). "Design of Center-TRACON Automation System." AGARD Meeting on Machine Intelligence in Air Traffic Management, Berlin, Germany.
- <sup>6</sup> Davis, T.J., K.J. Krzczowski and C. Bergh (1994). "The Final Approach Spacing Tool." IFAC 13<sup>th</sup> Symposium on Automatic Control in Aerospace, Palo Alto, CA.
- <sup>7</sup> Robinson III, J. E. and D.R. Isaacson (2000). "A Concurrent Sequencing and Deconfliction Algorithm for Terminal Area Air Traffic Control." AIAA Guidance, Navigation, and Control Conference, Denver, CO. Paper No. 2000-4473.
- <sup>8</sup> Robinson III, J.E., T.J. Davis and D.R. Isaacson (1997). "Fuzzy Reasoning-Based Sequencing of Arrival Aircraft in the Terminal Area," AIAA Guidance, Navigation and Control Conference, New Orleans, LA. Paper No. 97-3542
- <sup>9</sup> Isaacson, D.R. and J.E. Robinson III (2001). "A Knowledge-based Conflict Resolution Algorithm for Terminal Area Air Traffic Control Advisory Generation." AIAA Guidance, Navigation, and Control Conference, Montreal, Canada. Paper No. 2001-4116.
- <sup>10</sup> Erzberger, H. (1982). "Automation of On-Board Flightpath Management." 24<sup>th</sup> Annual Israel Conference on Aviation and Astronautics, Tel Aviv, Israel.
- <sup>11</sup> Quinn, C. and J.E. Robinson III (2000). "A Human Factors Evaluation of Active Final Approach Spacing Tool Concepts." 3<sup>rd</sup> USA/Europe Air Traffic Management R&D Seminar, Napoli, Italy.