AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum
20 - 22 September 2004, Chicago, Illinois

AIAA 2004-6347

Effect of Uncertainty on En Route Descent Advisor (EDA) Predictions

K. Tysen Mueller*, Richard Bortins†, David R. Schleicher‡, and Doug Sweet.§
Seagull Technology, Inc., Campbell, CA 95008

and

Richard A. Coppenbarger**
NASA Ames Research Center, Moffett Field, CA

The En Route Descent Advisor (EDA) is one of the Center TRACON Automation System (CTAS) decision support tools under development at the NASA Ames Research Center. EDA generates maneuver advisories for arrival aircraft to meet scheduled arrival times at the arrival meter fix, sometimes 20 – 25 minutes ahead of the aircraft's scheduled meter fix arrival time. This work determined the sensistivity of the EDA advisories to system uncertainties, including initial condition, environmental, and aircraft performance data errors. Using a Monte Carlo simulation that incorporates a Matlab Trajectory Synthesizer (TS) simulation, the sensitivities of the EDA predicted trajectory to these data error sources were obtained. The key metric is the meter fix crossing time error since this metric directly measures the performance of EDA. This performance analysis involved a minimum of 200 Monte Carlo trials per error parameter. In addition to the single aircraft performance analysis, the impact of aircraft prediction errors on conflict detection between closely-spaced aircraft was also explored. These Monte Carlo performance analyses determined how robust the EDA advisories are to input parameter uncertainties.

Nomenclature

g	=	*gravitational acceleration constant*
h	=	*altitude*
m	=	*aircraft mass*
$\bar{q}$	=	*dynamic pressure*
s	=	*distance*
t	=	*time*
x, y	=	*east and north position*
γ_{TAS} , γ_i	=	*aerodynamic and inertial flight path angle*
Ψ	=	*heading*
V_{TAS}	=	*true airspeed*
V_{CAS}	=	*calibrated airspeed*
V_W	=	*wind speed*
V_G	=	*ground speed*
M, M_0, M_{CAS}	=	*Mach number, constant descent mach number, and constant CAS descent mach number*
S	=	*aerodynamic wing area*
CAS_0	=	*constant descent calibrated airspeed*
C_L, C_D	=	*lift and drag coefficients*

* Senior Research Engineer, Air Transportation Group, 1700 Dell Ave., Campbell, CA.
† Senior Research Engineer, Air Transportation Group, 1700 Dell Ave, Campbell, CA, Senior AIAA Member.
‡ Senior Systems Engineer, Air Transportation Group, 1700 Dell Ave, Campbell, CA, Senior AIAA Member.
§ Senior Systems Engineer, Air Transportation Group, 1700 Dell Ave, Campbell, CA, AIAA Member.
** Aerospace Engineer, Automation Concepts Research Branch, AIAA Senior Member.

T = *thrust*
D = *drag*
L = *lift*
W = *weight*
κ = *throttle setting*
θ_{rw} = *relative wind angle*
ϕ = *bank angle*
ρ = *air density*
δt = *time error*
δS = *distance error*
δh = *altitude error*
$\delta\gamma_{TAS}$ = *flight path angle error*
δV_{TAS} = *true airspeed error*
δV_{CAS} = *calibrated airspeed error*
δV_W = *wind speed error*
δV_G = *ground speed error*
δM = *Mach number*
δT = *thrust error*
δD = *drag error*
δW = *weight error*

I Introduction

THE En Route Descent Advisor (EDA) is one of the Center TRACON Automation System (CTAS) decision support tools under development at NASA Ames Research Center. EDA computes advisories for the air traffic controller to help deliver the aircraft to an arrival meter fix in conformance with a scheduled time-of-arrival constraint. The time-of-arrival constraint is derived by another CTAS decision support tool, the Traffic Management Advisor (TMA), which computes the scheduled meter fix time-of-arrival to optimize the traffic flow into the terminal airspace. By computing strategic maneuver advisories, up to 25 minutes prior to the meter fix scheduled time-of-arrival, EDA can extend the capacity, efficiency, and workload benefits already attributed to TMA[20]. The efficiency of the EDA advisories depends on the accuracy of the information used to predict the nominal and the advisory aircraft trajectory from the current location to the meter fix. EDA relies heavily on the CTAS Trajectory Synthesizer (TS) to determine the nominal and flight time-adjusted predicted trajectories. To support this analysis, the TS requires initial (current) conditions (positions and velocities) from the surveillance radars, aircraft-specific performance models (part of the CTAS database), predicted atmospheric conditions (Rapid Update Cycle 2 (RUC-2) winds and temperature), filed flight plan information, and flight path constraints.

The motivation for this study was to determine how sensitive the EDA advisories are to system uncertainties. A preliminary review of the existing research literature reveals that no thorough sensitivity analysis of the range of potential EDA-specific descent maneuver options exists. Therefore, this study was initiated to obtain a better understanding of the effects of system sensitivities on EDA advisories, including conflict detection.

The first step was to identify the EDA system uncertainties and perform a literature survey of past trajectory uncertainty studies to identify a set of useful input parameter statistics and models. Separately, a Matlab simulation of the CTAS TS was developed that could be run efficiently with both nominal and stochastic input parameters.

Next, a number of standard EDA scenarios were selected. These consisted of typical arrival trajectories for a Boeing 737-300 aircraft flying into Dallas Fort Worth (DFW) that absorb varying levels of delay using EDA-specific advisories. The EDA-specific advisories included cruise speed change only, descent speed change only, and path stretch maneuvers.

Using a Monte Carlo simulation that incorporates the Matlab TS simulation, the trajectory sensitivities to various initial condition, environmental, and aircraft performance data errors were obtained for a single aircraft. In addition to the single aircraft performance analysis, the impact of aircraft prediction errors on conflict detection between closely-spaced aircraft was explored. Two specific conflict scenarios were explored to illustrate the potential for false alarms and missed alerts when performing conflict detection in the presence of the uncertainties associated with the single aircraft analysis.

These Monte Carlo performance analyses determine how robust the EDA advisories are to input parameter uncertainties. In addition, it identifies the parameters whose accuracy may need to be improved. The results of this study will also aid in setting various EDA decision parameters.

II EDA System Overview

An overview of the EDA system is depicted graphically in Figure 1. As shown, there are two main components to the EDA system, the Plan component and the Execution component.

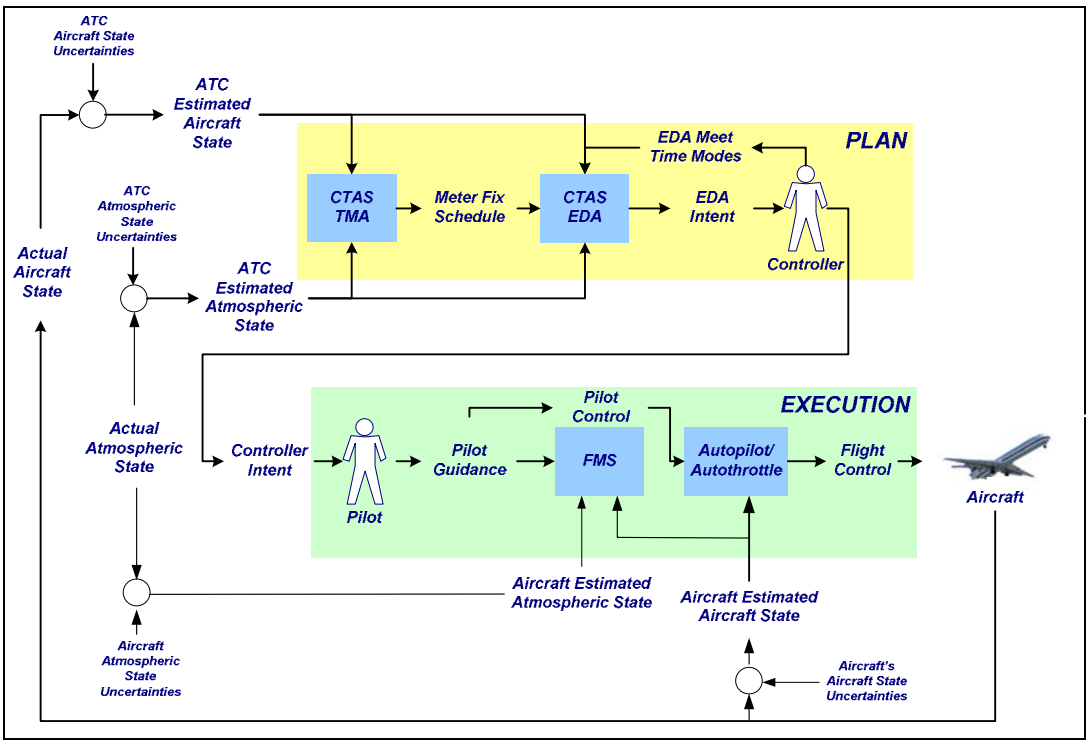

Figure 1 The EDA System Functional Flow Diagram

The purpose of the Plan component is to derive the EDA controller advisory for a specific aircraft to meet the aircraft's Scheduled-Time-of-Arrival (STA) at the meter fix. To accomplish this, the CTAS Traffic Management Advisor (TMA) develops a plan for all arrival aircraft based on the airport configuration / capacity, flight plan data, aircraft state data, and environmental data. The result of this overall plan is the scheduling and assignment of aircraft to a specified meter fix and an STA for each aircraft at their assigned meter fix. EDA uses the TMA-derived STA, along with the estimated aircraft state and estimated atmospheric state, to derive the specific EDA advisory for the controller to issue to the aircraft to achieve the desired STA. As depicted in Figure 1, there are uncertainties associated with the aircraft state and atmospheric input data that can introduce errors into this advisory development process.

The Execution component of EDA uses the EDA-derived advisory as an input to the pilot, who then executes the advisory through the various automation systems on-board the aircraft (e.g., the FMS and autopilot/autothrottle). Additional uncertainties are introduced in the execution phase due to errors in the aircraft's sensors and data with regard to the aircraft's state (location, speed) as well as the atmospheric state. It is important to note that while some of the error types are common between the Plan and Execution components, the error sources and the uncertainties associated with them can be quite different, as summarized in Table 1. For example, the speed error associated with the radar system providing data to EDA for the Planning effort exhibits a relatively large level of uncertainty compared to the speed error associated the Execution phase, which uses the on board aircraft sensors to determine aircraft speed.

The EDA decision logic uses inputs from TMA, air traffic control (ATC), and the airline operations center (AOC) to define the aircraft trajectory horizontal and vertical profile parameters. These parameters are combined

with trajectory prediction input parameters by the EDA TS to determine the predicted cruise/descent trajectory to the meter fix. The EDA TS also uses calculations performed by the FMS to compute the top of descent (TOD) location to achieve the EDA-advised descent speed. Based on this predicted trajectory, EDA provides a maneuver advisory to the aircraft that specifies the TOD location, the descent Mach-CAS schedule, and any path stretch maneuvers. The aircraft combines the EDA advisory parameters with aircraft trajectory input parameters to fly the aircraft over the meter fix.

Table 1 EDA Trajectory Prediction Parameters

DATA		DATA SOURCE	
Parameter	**Symbol**	**TMA/EDA Planning**	**Aircraft Execution**
Waypoints	λ_{WP}, θ_{WP}	Flight Plan (Radar)	FMS
Current Position	X, Y, h	Radar	FMS
Current Velocity	V_G, ψ_G, γ_G	Radar	FMS
Aircraft Bank Angle	ϕ	TS Database	FMS
TOD Location	s_{TOD}	(Computed)	EDA
Mach/CAS Schedule	M, V_{CAS}, h_{Switch}	(Computed)	EDA
Path Stretch Start	$s_{PS,S}$/ $t_{PS,S}$	(Computed)	EDA
Path Stretch Return	$s_{PS,F}$/ $t_{PS,F}$	(Computed)	EDA
Aircraft Weight	W	TS Database	FMS
Aircraft Thrust	T	TS Database	FMS
Aircraft Drag	D	TS Database	FMS
Winds	v_W, ψ_W	RUC-2 (1hr)	FMS (3hr)
Air Temperature	τ	RUC-2 (1hr)	FMS (3hr)
Air Pressure	p_0	RUC-2 (1hr)	FMS (3hr)
MF Crossing Speed	$V_{CAS,MF}$	TS Database	FMS
MF Crossing Altitude	h_{MF}	TS Database	FMS
MF STA/ETA	t_{STA}, t_{ETA}	TMA	EDA

A list of the input parameters used by EDA is summarized in Table 1. The parameters are separated into Planning and Execution parameters. Also, parameters that fall into groups such as initial conditions, advisory parameters, aircraft performance, and weather are shown with different shaded rows. One of the things to note in Table 1 is that there are more Execution parameters than Planning parameters. Also, when both Planning and Execution require the same parameters, their data sources differ. This latter feature will be exploited during the Monte Carlo analysis to argue that the Planning and Execution input errors are for the most part independent.

The interaction between the various operations in Figure 1 can also be examined is by drawing a state diagram as shown in Figure 2. This figure shows that the EDA planning process requires both an inner and outer control loop. The inner control loop flies a single aircraft trajectory to the meter fix and checks to see whether the estimated time of arrival (ETA) matches the TMA-specified scheduled time of arrival (STA). The outer loop then is used to iterate on the TOD location and Mach-CAS schedule to achieve a trajectory that meets the STA.

The aircraft execution involves only a single loop. This loop predicts the aircraft trajectory and steers the aircraft to the EDA advisory-specified TOD location and on to the MF using the specified Mach-CAS schedule.

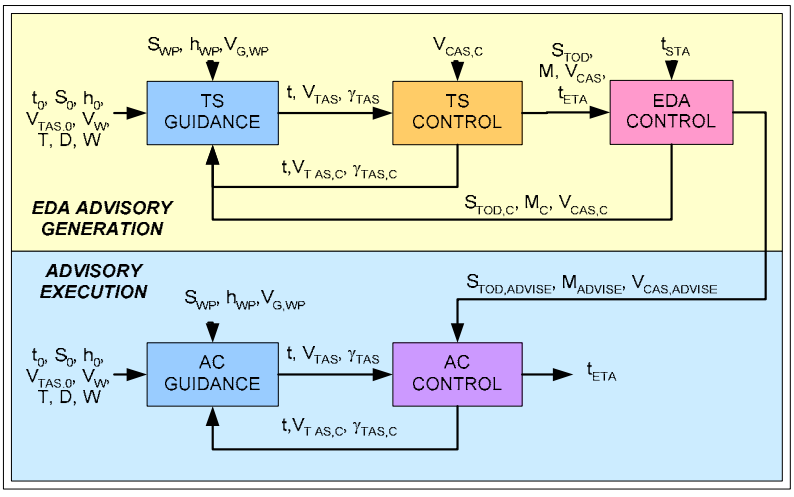

Figure 2 EDA-TS State Diagram

III Monte Carlo Performance Simulation

The general approach selected to determine the EDA performance was a Monte Carlo simulation of all the input errors that contribute to key output parameter metrics. The key metric that will be used is the meter fix crossing time error since this metric directly measures the performance of EDA.

The general structure of the simulation is illustrated in Figure 3. This figure is similar to the state diagram of Figure 2, except that the latter replaces the parameters with their errors. Also the latter diagram has decoupled the Planning and the

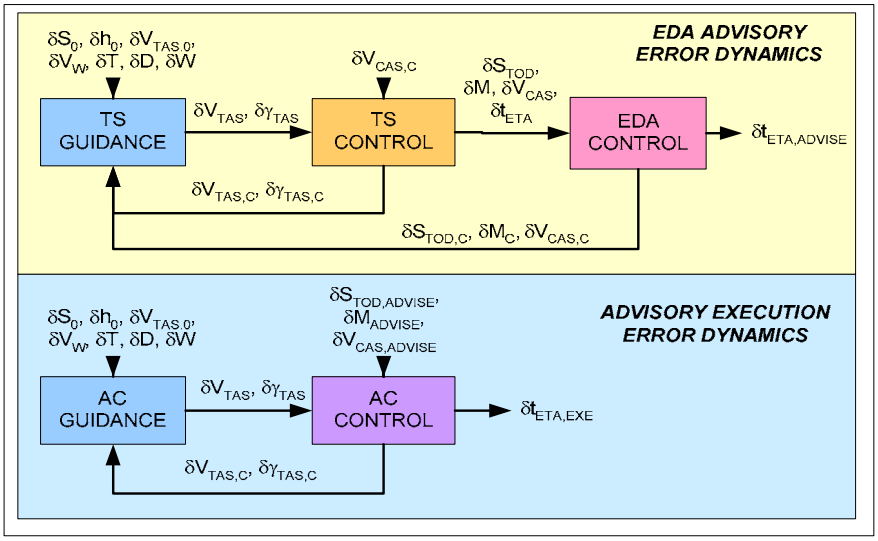

Figure 3 Decoupled EDA/TS Planning and Execution Monte Carlo State Error Diagram

Execution errors.

Decoupling the Planning and Execution Monte Carlo simulations is motivated by the fact that the Matlab TS simulates (open-loop) trajectories while the TS synthesizes (closed-loop) trajectories. The TS accomplishes this trajectory synthesis by using a backward integration from the MF required crossing conditions to the TOD. In addition, it combines the backward integration with a forward integration from the MF to the TOD. This process is used to assure that the TS trajectories satisfy both the initial and MF end conditions.

With this decoupling, the impact of MF crossing altitude and speed errors on TOD descent location and Mach-CAS schedule cannot be evaluated. The MF crossing altitude and speed errors arise due to the initial condition and enroute prediction errors. Since the CTAS TS synthesizes each trajectory such that the initial and final conditions are simultaneously satisfied, any prediction errors will lead to an adjustment to the predicted TOD location and the descent Mach-CAS schedule. Hence, with this decoupling the assumption is made that the remaining advisory errors are more significant than those due to MF crossing altitude and speed errors. The remaining advisory errors consist of the advisory quantization errors and flight technical errors (FTE).

Monte Carlo runs were made with all the mean and one sigma errors used at the same time for each trajectory. This produced the accuracy of the initial EDA advisory planning phase after the aircraft has entered Center airspace and TMA has provided a scheduled time of arrival (STA) at the MF. A separate set of Monte Carlo runs were made with all mean and one sigma input errors turned on for the aircraft to evaluate the performance of the advisory execution phase.

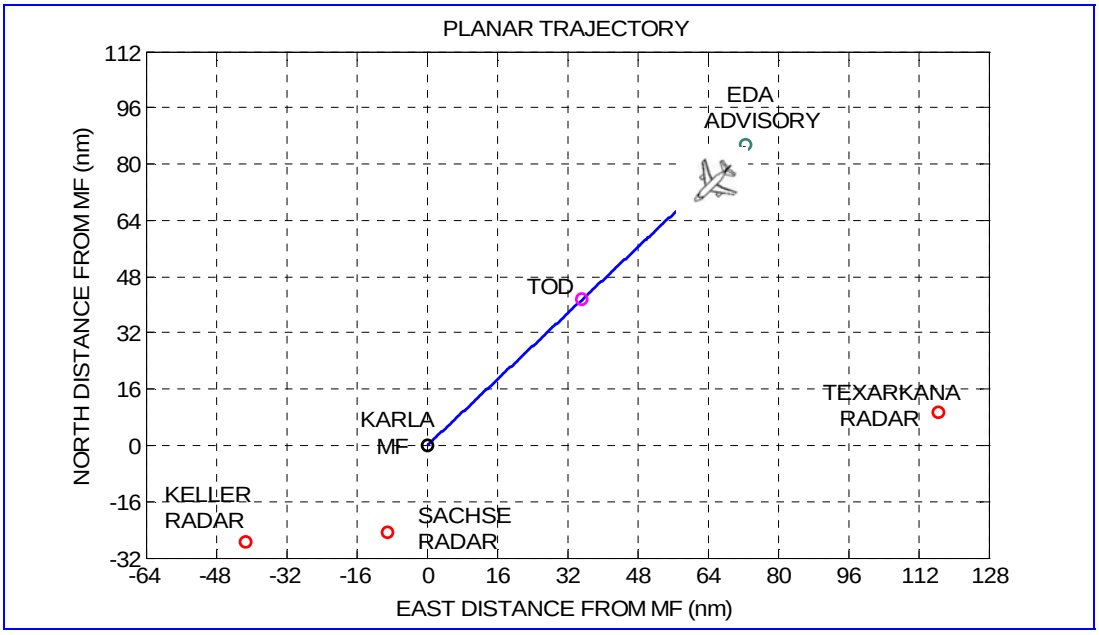

Figure 4 Nominal Straight-In Planar Trajectory with Surveillance Radars

IV Nominal Trajectories

Five different trajectories were used to explore the impact of EDA planning and execution errors on MF crossing time delays. A straight-in (un-delayed) nominal trajectory served as the baseline trajectory. This trajectory is illustrated in Figure 4 for an arrival into the Dallas Ft. Worth at the northeast meter fix (MF), KARLA.

Shown in a local north-east distance coordinate system centered on KARLA MF, this figure also includes the locations of the surveillance radars in the vicinity. In particular, the Texarkana radar, with a sweep of 12 sec, is used to track the initial approach of the aircraft to the top of descent (TOD). The Sachse radar, with a 10 sec sweep in turn, is used to track the aircraft during the descent to the meter fix.

Figure 5, in turn presents an altitude profile of this same trajectory. It shows that the aircraft arrives at a cruise altitude of 30,000 ft and descents to 11,000 ft before crossing the KARLA MF.

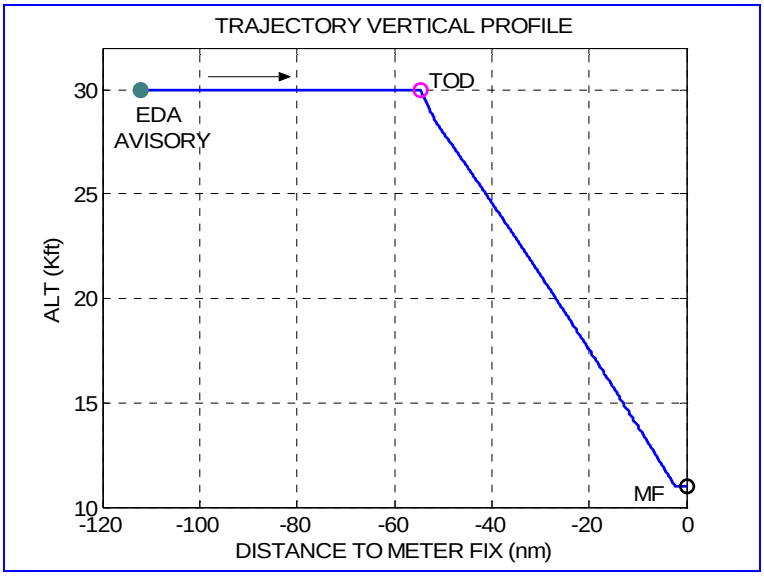

Figure 5 Nominal Straight-In Vertical Trajectory

Two of the delayed trajectories follow the same straight-in trajectory as in Figure 5. They differ in that one trajectory introduces a 45 sec delay during cruise while the second trajectory introduces a 60 sec delay during descent.

The last two trajectories incorporate a path stretch maneuver, as illustrated in Figure 6. The path stretch maneuver is made using a 60 deg track angle turnout maneuver. The path stretch turn back maneuver is used to control the length of the delay. Hence for one trajectory, a 3 min delay is obtained while for the second trajectory a 6 min delay is obtained.

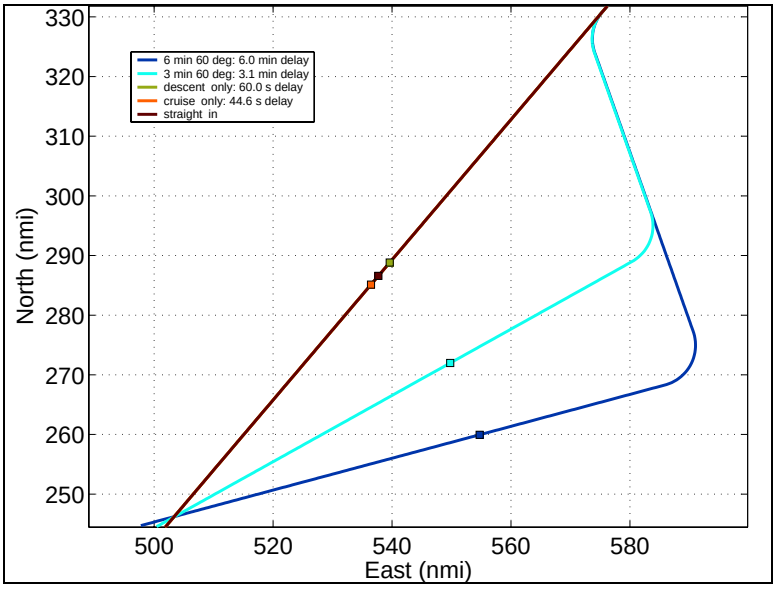

Figure 6 Cruise-Only, Descent-Only, and Path Stretch Delayed Trajectories

V Monte Carlo Input Error Statistics

A reference search was made to find the input parameter error statistics required for the Monte Carlo error analysis. In general, the references fell into three categories based on what they provided:

1. Input parameter statistics: Initial position and velocity, weight, thrust, drag, winds, etc.
2. Intermediate parameter statistics: Radar position and velocity, turn distance, flight path angle, track angle, altitude and altitude rate, etc.
3. Output parameter statistics: ETA, MF altitude, MF speed, etc.

References [1 - 13] provide the desired input error statistics based on field test measurements or database analyses. The second category consisted of analytic and simulation analyses that provided intermediate parameters that will be obtained directly from the Monte Carlo simulation. Finally, the third category included output parameter statistics obtained either through simulation or field data measurements. This last category would only help to validate the results obtained with the Monte Carlo performance analysis. References [14 - 15] fell into the last two categories.

The key input errors that correspond to the input parameters of Table 1 are summarized in Table 2. The errors are grouped into Planning and Execution error mean (μ) and standard deviation (σ). The errors are also grouped into similar categories using the shaded rows. No statistics are provided for some of the EDA/TMA planning errors since

Table 2 EDA Trajectory Key Prediction Parameter Errors

DATA			INPUT DATA STATISTICS			
			TMA/EDA Planning		Aircraft Execution	
Parameter	Symbol	Units	μ	σ	μ	σ
Initial Position	S	nm	0	0.2	0	0.25
Initial Velocity	V_G	kts	-8	15	0.6	2.1
TOD Location • **FTE** • **Advisory**	s_{TOD}	nm	Not Applicable		0 0.5	0.25 0.29
Mach • **FTE*** • **Advisory**	M		Not Applicable		0.001/0.0025 0.005	0.0035/0.0085 0.003
CAS • **FTE*** • **Advisory**	V_{CAS}	kts	Not Applicable		0.32/0.05 0.5	1.3/4.0 0.29
Path Stretch Angle • **FTE** • **Advisory**	$\theta_{PS,S}$	deg	Not Applicable		0 0.5	0.15 0.29
Path Stretch Return • **FTE** • **Advisory**	 $s_{PS,F}$ $t_{PS,F}$	 nm sec	Not Applicable		 0 0	 0.25 17
Aircraft Weight	W	%	5.5	5.6	0	5.6
Aircraft Thrust-Drag*	$(T\text{-}D)$	%	8.4/5.5	1.4/2.1	0/0	1.4/2.1
Winds*	v_W,	kts	0	10.3/9.6	0	11.2/10.2
Air Temperature*	τ_0	deg C	0	1.1/1.0	0	1.2/1.1

*Cruise/descent statistics

these are derived based on the remaining EDA/TMA planning errors. Their statistics, however, are shown in the aircraft execution columns when they are sent as advisory parameters to the aircraft.

All the errors in Table 2 are assumed to have Gaussian statistics except for the advisory errors. Since the advisory errors arise from truncating or rounding the exact advisory parameter to the nearest integer value (e.g., nearest 0.01 for Mach), the advisory errors are uniformly distributed. Also, all of the error statistics are assumed to be random biases except for the flight technical errors (FTE). The FTE are probably best modeled as a time-varying (Gauss-Markov process) bias. However, to simplify the Monte Carlo simulation, these were treated as random (white noise) errors.

In considering the EDA performance for the various metrics, the performance of the nominal advisory can be measured assuming the aircraft will fly this advisory perfectly (perfect execution). Alternately, the overall EDA performance can be determined for the various metrics based on the combined performance of the EDA advisory and the advisory execution by the aircraft.

VI Aircraft Model

The Boeing 737-300 was selected for this study because it is a common class of aircraft in commercial service in the NAS. Figure 7 shows the number of flights per class of aircraft during Sunday, 26 May 2002, in the NAS.

The figure shows that B-737 flights occur over twice as frequently as any other identified class of aircraft. The B-737-300 model selected for this study represents over one-third (2474) of the B73X flights in Figure 7.

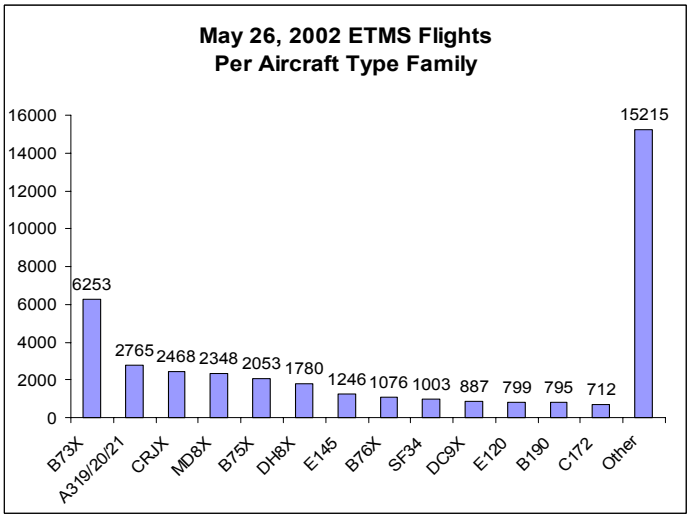

Figure 7 Number of Flights per Aircraft Class in the NAS for May 26th, 2002

VII Trajectory Synthesizer Model

A CTAS Trajectory Synthesizer

The trajectory simulation model uses the same aircraft equations as the CTAS and EDA trajectory synthesizer[17,18]. The equations are as follows with variables defined at the beginning of this paper:

True airspeed:

$$\dot{V}_t = \frac{T(h,M;\kappa) - D(h,V_t,L)}{m} - g\gamma_a + \frac{d}{dt}(V_w \cos\theta_{rw}) \tag{1}$$

Altitude: $$\dot{h} = V_t\gamma_a = V_g\gamma_i \tag{2}$$

Heading: $$\dot{\Psi}_i = \frac{L\sin\phi}{mV_g} \tag{3}$$

Lift; $$L = \frac{mg}{\cos\phi} \tag{4}$$

Path distance: $$\dot{s} = \sqrt{V_t^2 - (V_w\cos\theta_{rw})^2} - V_w\cos\theta_{rw} = V_g \tag{5}$$

East position: $$\dot{x} = V_g \sin\Psi_i \tag{6}$$

North position: $$\dot{y} = V_g \cos\Psi_i \tag{7}$$

Aircraft mass: $$\dot{m} = 0 \tag{8}$$

Computing the drag requires a few more equations because drag is expressed in terms of lift coefficient and drag coefficient. The equations are

Drag coefficient: $$C_D = \frac{D}{\bar{q}S} \tag{9}$$

Lift coefficient: $$C_L = \frac{L}{\bar{q}S} \tag{10}$$

Dynamic pressure: $$\bar{q} = \frac{\rho \cdot V_t^2}{2} = 0.7PM^2 \tag{11}$$

For the idle thrust, constant Mach, constant CAS descents, the following equations were used to compute the flight path angle[18]:

Constant Mach flight path angle: $$\gamma_a^*(h;M_0) = \left(\frac{1}{m}\right)\left[\frac{T(h,M_0;\kappa_0) - D(h,V_t;L)}{V_T\left[\left.\frac{dV_T}{dh}\right|_{M_0} - \frac{d}{dh}\left(V_w\cos\theta_{rw}\right)\right] + g}\right] \tag{12}$$

Constant CAS flight path angle: $\gamma_a^*(h;CAS_0) = \left(\frac{1}{m}\right)\left[\frac{T(h,M_{CAS};\kappa_0) - D(h,V_t;L)}{V_T\left[\frac{dV_T}{dh}\Big|_{CAS_0} - \frac{d}{dh}\left(V_w \cos\theta_{rw}\right)\right] + g}\right]$ (13)

The partial derivatives of air speed with respect to altitude are evaluated at the descent Mach number or the descent CAS, as appropriate.

B Matlab Trajectory Synthesizer

To simulate such trajectories, the Matlab TS uses 4 modules for

1) Level turns
2) Level decelerations
3) Cruise
4) Mach/CAS descent

These modules are augmented with trajectory parameters that specify where decelerations start and the target speeds. They also specify where turns start and the target headings and where the Mach/CAS descents start. Finally, they also include the descent Mach number and CAS as well as the meter fix location. Figure 8 is a block diagram of the Matlab TS. The figure shows the input variables in italics and the output variables in a normal font.

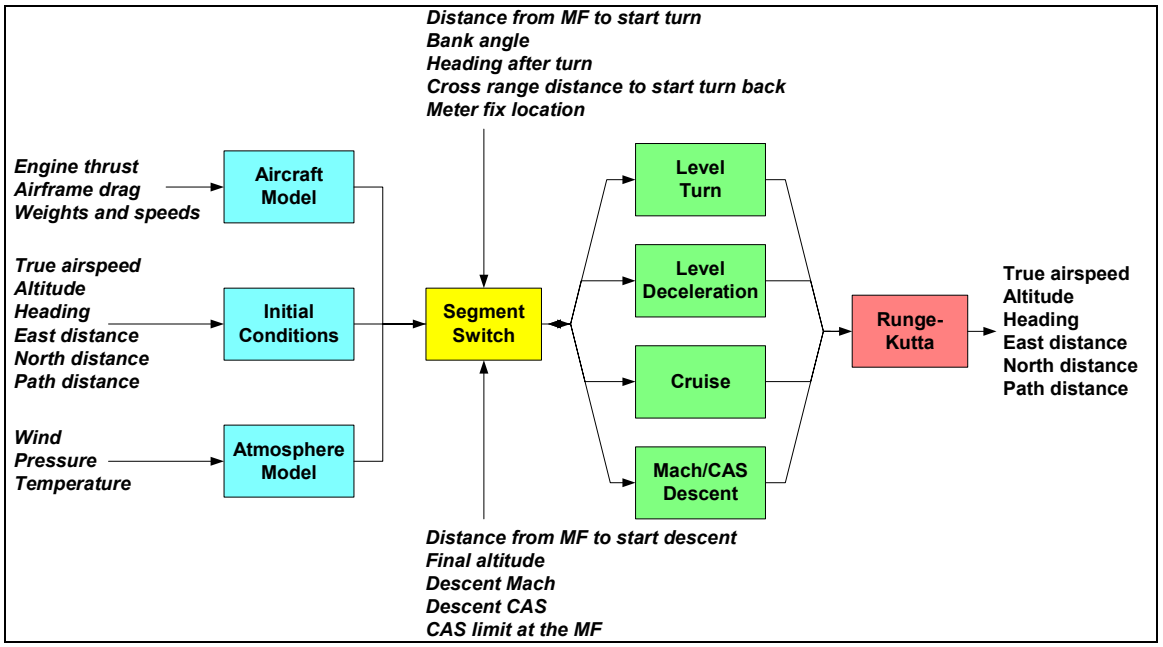

Figure 8 Matlab trajectory simulation (TS) block diagram

A Matlab Runge-Kutta function is used to perform the integrations. This function uses two approximations, one of order 2 and one of order 3, and their difference to estimate the error in each step of the integration process. Matlab interpolation functions are used to produce values for the drag coefficient from the aerodynamic data and the idle thrust from the engine data.

Selected Matlab TS trajectories were compared to trajectories generated by the CTAS TS. The process consisted of first selecting 2 representative approach trajectories. We used a straight-in approach to KARLA and a 6 minute delay, 60 degree turn-out approach to KARLA. Next, TS dump files were generated for these trajectories. A Matlab function was developed to read TS dump files and use the data to produce the necessary trajectory parameters for the Matlab TS. Finally, the Matlab TS was run with these trajectory parameters and these results were compared to the TS dump results.

Table 3 Differences between the Matlab TS and the TS used by CTAS and EDA for the straight-in approach to KARLA in ZFW (No winds)

	Matlab TS Minus Real TS Difference
Meter Fix Crossing Time (s)	-0.20
Bottom of Descent Time (s)	0.70
East Position (ft)	6.08
North Position (ft)	-18.23
Meter Fix Altitude (ft)	-0.71
Meter Fix CAS (kt)	0.74

Table 3 shows the results of the comparisons for the straight-in approach to KARLA. All of the differences are insignificant and may be due to integration differences, interpolation differences, or truncation or rounding differences. We did pinpoint one sign error in the CTAS TS where the flight path angle computations are corrected for nonstandard day conditions.

Table 4 shows results of the comparisons for the 6 minute delay, 60 degree turn-out approach to KARLA. These differences are much larger than the straight-in differences and are due to some non-standard behavior in the CTAS TS turn-out trajectories. Hence, most of the differences in Table 4 can be attributed to this non-standard behavior.

Table 4 Differences between the Matlab TS and the TS used by CTAS and EDA for the 6 minute delay, 60 degree turn-out approach to KARLA in ZFW (No winds)

Variable	Event		
	Top of descent	Bottom of descent	Meter fix crossing
Time (s)	-2.663	-1.875	-6.082
Speed (kt)	0.044	-0.062	5.083
Altitude (ft)	-2.020	0.0	-1.479
Distance (nm)	0.311	0.110	0.276

The final verification and validation case was a straight-in approach to KARLA with a uniform and constant wind field. Since the CTAS TS uses RUC data files to create a realistic wind field, comparison results were unavailable. Instead, expected and simulated results for compared for head wind and no wind cases for three trajectory segments, as summarized in Table 5. The three trajectory segments were cruise, a short cruise segment after meter fix crossing, and descent.

Based on the results shown in Table 3 and 4, it was concluded that the Matlab TS matches the CTAS TS adequately for the purposes of performing an EDA performance analysis. Based on the results shown in Table 5, it was concluded that the uniform and constant wind field is implemented correctly in the Matlab TS.

Table 5 Differences between expected path lengths and Matlab TS simulated path lengths for wind and no-wind trajectories and selected trajectory segments

Segment	Case	Expected (nm)	Simulated (nm)	Difference (nm)
Cruise to TOD	**Wind**	59.84	59.84	3.19e-005
	No wind	59.84	59.84	3.38e-004
Cruise after MF	**Wind**	2.49	2.49	0
	No wind	2.49	2.49	8.88e-016
Descent	**Wind - no wind**	-9.754	-9.759	5.75e-003

VIII Monte Carlo Simulation Results

A Number of Monte Carlo Trial Selection

To determine the number of Monte Carlo trials that are required to achieve reasonable Monte Carlo output statistics, the finite sample size statistics algorithms were reviewed[19]. Hence for N sample output errors (ε_n), the sample mean (μ_N) and its uncertainty ($\sigma_{\mu N}$) are obtained as follows:

Sample mean:
$$\mu_N = \frac{1}{N}\sum_{n=1}^{N}\varepsilon_n \tag{14}$$

Uncertainty in sample mean:
$$\sigma_{\mu_N} = \frac{\sigma}{\sqrt{N}} \tag{15}$$

The sample standard deviation (σ_N) and its uncertainty ($\sigma_{\sigma N}$) are obtained as follows:

Sample standard deviation: $$\sigma_N = \sqrt{\frac{1}{(N-1)}\sum_{n=1}^{N}(\varepsilon_n - \mu_N)^2} \quad (16)$$

Uncertainty in sample standard deviation $$\sigma_{\sigma_N} = \frac{\sigma}{\sqrt{2N}} \quad (17)$$

In (15) and (17), σ represents the unknown true (infinite sample) standard deviation. Then by focusing on the desired accuracy of the sample standard deviation, (17) can be inverted and expressed in terms of a sample standard deviation uncertainty percentage ($k_{\%}$) that is user selected:

Monte Carlo uncertainty tolerance: $$k_{\%} \equiv 100\left(\frac{\sigma_{\sigma_N}}{\sigma}\right) = \frac{100}{\sqrt{2N}} \quad (18)$$

Finally, inverting (18), leads to an expression for the required number of trials N if the sample standard deviation is no greater than $k_{\%}$:

Number of Monte Carlo trials: $$N = \frac{10^4}{2k_{\%}^2} \quad (19)$$

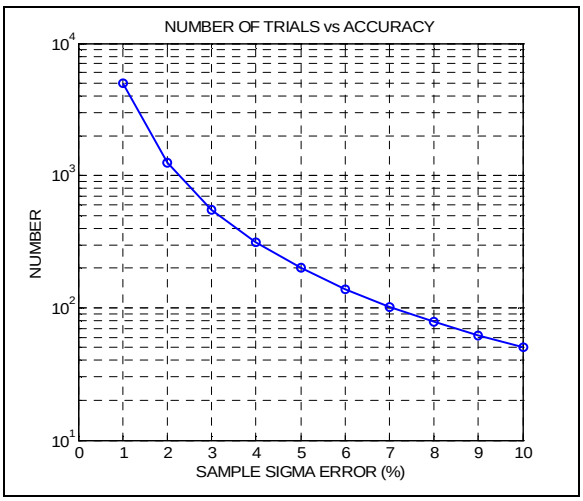

Figure 9 Number of Monte Carlo Trials vs Sample Standard Deviation Accuracy

This equation is illustrated in Figure 9. It shows that if an output sample uncertainty of 5% is acceptable, only 200 trials are required. However, if the uncertainty must not exceed 2%, then 1250 trials are required.

Since the Matlab TS simulation requires about 33 min to generate 200 trials, 200 trials was selected as a reasonable and practical standard for the cases presented in this section.

At the end of this section, the final results will be presented as well as their 95% confidence limits. These confidence limits are obtained using plus and minus (±) twice the sample statistics uncertainty values as computed by (15) and (17). This is justified since (15) is very nearly Gaussian for N > 30 while (17) is nearly Gaussian for N > 100 based on finite sample statistics[19].

B TOD Planning Weight Sensitivity

Figure 10 shows the input sample frequency function when a weight error of mean 5.5% and standard deviation of 5.6% is sampled 200 times for the B737 with a nominal TOD weight of 100,000 lb. The sample mean is 5.3 % (5279 lb) while the sample standard deviation is 5.1% (5092 lb). Also shown is the analytic Gaussian density function centered on the sample mean with a spread corresponding to the sample standard deviation. Figure 10

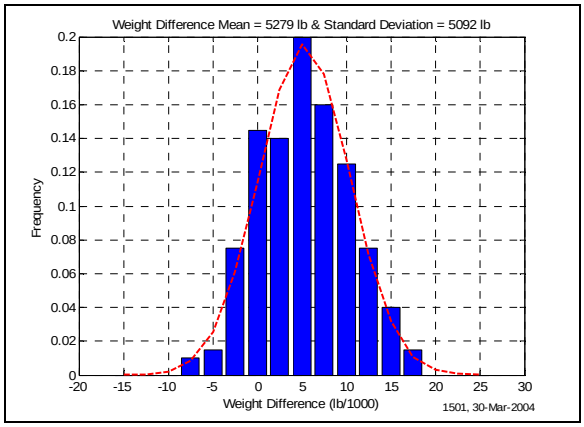

Figure 10 Planning Trajectory TOD Weight Uncertainty Sample Frequency Function

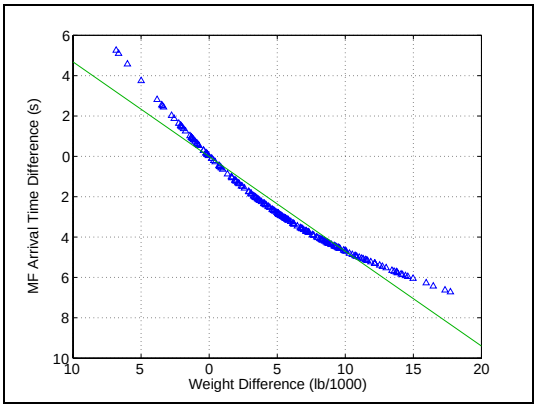

Figure 11 Planning Trajectory TOD Weight-MF Crossing Time Delay Scatter Diagram

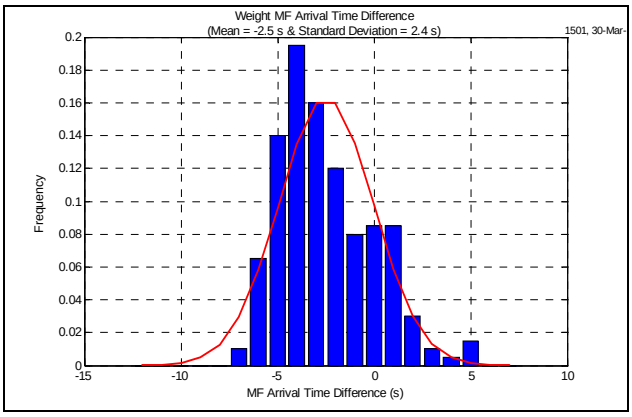

Figure 12 Planning Trajectory MF Arrival Time Delay Sample Frequency Function (Weight Uncertainty)

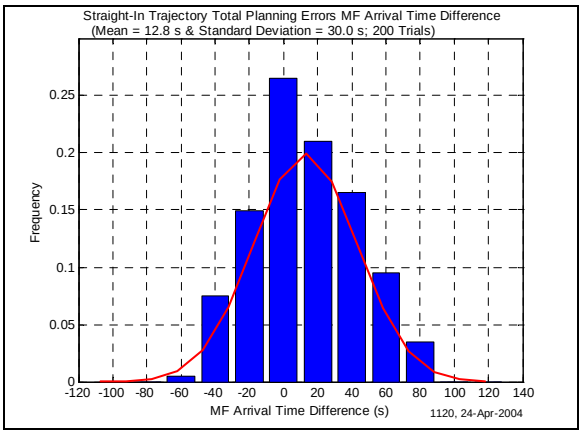

Figure 13 Planning Trajectory MF Arrival Time Delay Sample Frequency Function

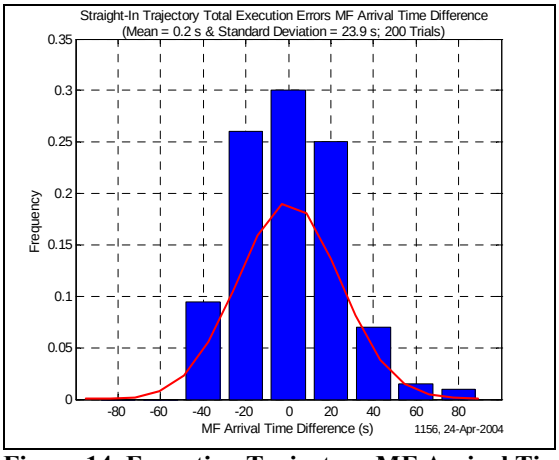

Figure 14 Execution Trajectory MF Arrival Time Sample Frequency Function

verifies that the sample and actual statistics are reasonably close and nearly Gaussian.

Figure 11 is a scatter diagram that shows the dependence of the MF crossing time delay on the sample TOD weight error. The scatter function is seen to be slightly concave up rather than perfectly linear.

Based on Figures 10 and 11, the MF arrival time sample frequency function due to TOD weight uncertainty is obtained as shown in Figure 12. This figure shows that the MF arrival time frequency density function is slightly skewed to the left. This is consistent with the slightly concave scatter function of Figure 11.

C Straight-In Trajectory Results

When all the planning errors are present, the MF delay sample frequency function of Figure 13 is obtained. This figure shows that the planning trajectory MF delay error is skewed slightly to the right.

Table 6 Planning Trajectory MF Crossing Time Delay Statistics

Input Parameters	Parameter Statistics		MF Cross Time	
	Mean	Sigma	Mean	Sigma
Initial Position	0	0.2 nm	0.1 sec	1.7 sec
Initial Speed	-8 kts	15 kts	14.9 sec	24.8 sec
Weight	5.5 %	5.6 %	-2.5 sec	2.4 sec
(Thrust – Drag)	8.4/5.5 %	1.4/2.1 %	0.12 sec	0.02 sec
Winds	0	10.3/9.6 kts	-1.8 sec	17.9 sec
Temperature (Descent)	0	1.1/1.0 ^{0}C	0	0.4 sec
Total Planning			12.8 sec	30.0 sec

A summary of the individual planning error MF delay sensitivities as well as the total planning error MF delay statistics is presented in Table 6. From this table it can be seen that the initial speed and wind speed error are the leading contributors to the total planning MF delay error.

Table 7 Execution Trajectory MF Crossing Time Statistics

Input Parameters	Parameter Statistics		MF Cross Time	
	Mean	Sigma	Mean	Sigma
Initial Position	0	0.25 nm		
Initial Speed	0.6 kts	2.1 kts	-0.8 sec	2.9 sec
Top of Descent*	0/0.5 nm	0.25/0.29 nm	1.9 sec	1.1 sec
Weight	0	5.6 %		
(Thrust – Drag)	0	1.4/2.1 %		
Winds	0	11.2/10.2 kts	0.1 sec	23.1 sec
Temperature (Descent)	0	1.1 ^{0}C		
Total Execution			0.2 sec	23.9 sec

When all the execution errors are present, the MF arrival time sample frequency function of Figure 14 is obtained. While the total execution error is not as large as the planning error, it is still a significant source of MF arrival time variation. The execution error statistics are summarized in Table 7. As can be seen, the wind speed uncertainty is the major contributor to the total execution MF arrival time error.

D Monte Carlo Summary

Table 8 Trajectory MF Crossing Time Delay Statistics (200 Trials, 95% Confidence Limits)

TRAJECTORY	PLANNING (sec)		EXECUTION (sec)		COMBINED (sec)	
	Mean	Sigma	Mean	Sigma	Mean	Sigma
Straight-In (No Delay)	12.8 ±4.2	30.0 ±3.0	0.2 ±3.8	23.9 ±2.4	12.4 ±5.4	38.1 ±3.8
Cruise-Only 45 sec Delay	5.6 ±3.1	22.2 ±2.2	2.0 ±3.3	23.1 ±2.3	7.6 ±4.5	31.7 ±3.2
Descent-Only 60 sec Delay	15.3 ±4.0	28.4 ±2.8	3.8 ±3.1	21.9 ±2.2	18.1 ±5.2	36.5 ±3.6
Path Stretch 3 min Delay	23.0 ±4.9	35.0 ±3.5	-1.4 ±3.7	25.8 ±2.6	21.4 ±6.4	45.1 ±4.5
Path Stretch 6 min Delay	20.5 ±5.4	38.2 ±3.8	1.6 ±4.0	28.2 ±2.8	22.1 ±6.2	46.8 ±4.4

The MF delay planning and execution error statistics for the five trajectories are summarized in Table 8. Also included are the combined planning and execution errors as well as the 95 % confidence limits for these sample statistics.

Examining Table 8, it can be seen that the planning errors are larger than the execution errors for all trajectories with the exception of the Cruise-Only 45 sec delay trajectory. For this trajectory, the errors are nearly the same. Also, while the planning errors are generally larger than the execution errors, both are significant. contributors to the combined errors.

Overall, the planning and execution errors in the path stretch trajectories produce larger planning and execution MF delays than the three straight-in trajectories. This result is not unexpected since the path stretch trajectories have longer flight times.

IX Analysis of EDA Conflict Detection Performance

The main objective of air traffic management and control is the safe separation of all aircraft in the NAS during all stages of flight. Decision support tools designed to make this objective easier to achieve rely on aircraft trajectory predictions to uncover future aircraft conflicts and to test alternative routes intended to eliminate the future conflicts.

The last section has shown that errors in input data significantly degrade the precision of the trajectory predictions. This section explains some of the consequences of the trajectory prediction errors on conflict detection.

A Conflict Detection

The basic concept in conflict detection is that a decision-maker uses input data and some rules to conclude that a conflict either exists or does not exist. The decision-maker can be a human or a computer application. The input data can be instrument readings or trajectory predictions. The rules can be simple or complex. Because the input data have errors, the probability that the decision will be correct is less than unity and the probability that the decision is incorrect will be greater than zero.

Conflict detection has four outcomes and associated probabilities:

1. *Probability of correct detection:* the likelihood that the decision that a conflict exists is correct.
2. *Probability of false alarm:* the likelihood that the decision that a conflict exists is incorrect.
3. *Probability of missed alert:* the likelihood that the decision that a conflict *not* exist is incorrect
4. *Probability of no conflict:* the likelihood that the decision that a conflict *not* exist is correct

Hence, the goal of conflict detection is to maximize categories (1) and (4) and to minimize categories (2) and (3).

For the conflict analysis, two sets of truth trajectories were defined. The first set has two aircraft flying at the same cruise speed with constant in-track spacing during cruise of 7 nm. This scenario was analyzed by using frequency distributions at different times from the same sample population.

The second set has two aircraft flying at different cruise speeds with converging in-track spacing during cruise. Four speed differences were chosen such that the true conflict would occur during the cruise phase at 200 s, 300 s,

400 s and 500 s into the flight. The initial separation is 7 nm and a conflict is defined a separation that is less than or equal to 5 nm. In both cases, trajectory prediction errors produce false alarms and missed alerts.

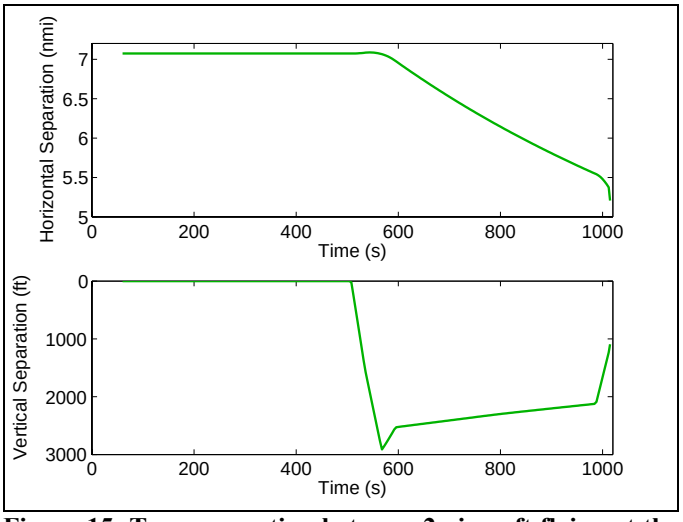

Figure 15 True separation between 2 aircraft flying at the same cruise speed with constant in-track spacing during cruise of 7 nm

B Same Cruise Speed Scenario

Figure 15 shows the true horizontal and vertical separation between two aircraft flying at the same cruise speed with constant in-track spacing during cruise of 7 nm. The true separation does not start decreasing until after the lead aircraft completes the constant Mach segment after top of descent. In fact, during the constant Mach segment, separation actually increases slightly because the true air speed is increasing to change the CAS from the cruise value of 271 kt to the descent value of 280 kt. The true separation always exceeds 5 nm in this scenario.

Adding input errors to the trajectory prediction produces an uncertainty in the aircraft position. Figure 16 shows how the dispersion in in-track position grows with time due to errors in initial speed for a single aircraft. The time interval is 60 s and the first time is 80 s. Top of descent is at 504 s. The Monte Carlo population size is 200. The dispersion stops growing significantly after top of descent because the speed changes to the descent value.

Figure 17 shows the relative separation (in-track) dispersions for two aircraft where the leading aircraft is 500 s past the Center boundary and the trailing aircraft is 440 s past the Center boundary. Note that the aircraft are actually still separated by 7 nm. However, due to trajectory prediction uncertainties, a small fraction (30% of the cases) are perceived to have violated the minimum safe separation limit of 5 nm. Hence there is a probability of a false alarm of 0.30.

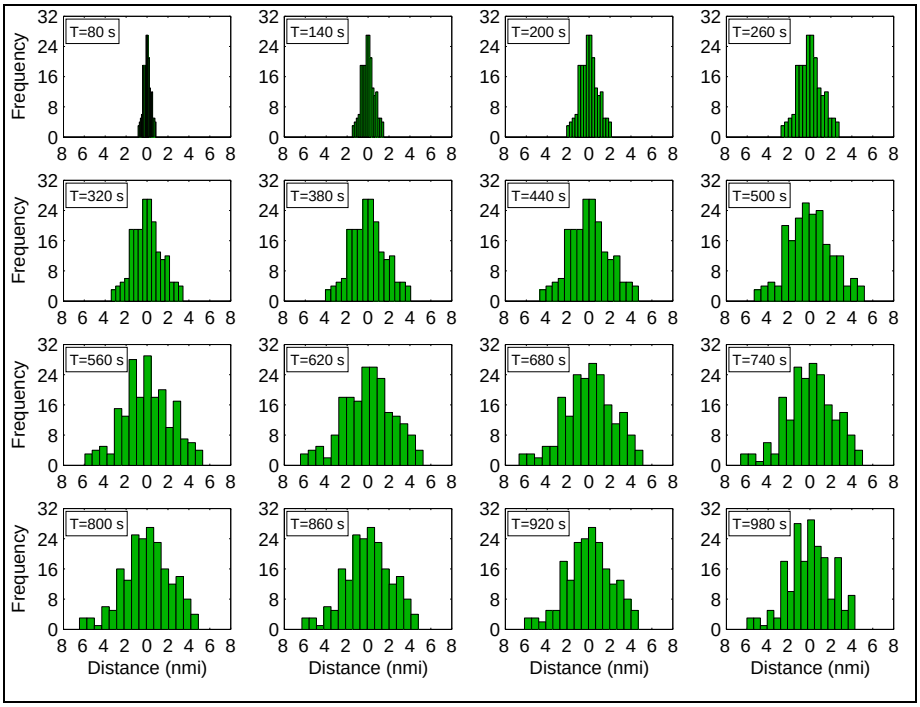

Figure 16 In-track dispersion as a function of time due to errors in initial speed

Extending these results to additional flight times during the cruise segment, yields Figure 18. This figure shows that the probability of false alarm exceeds 10% after the leading aircraft is 200 s past the Center boundary when the true in-track separation is 7 nm.

The effect of varying the actual (true) separation on the probability of false alarm is examined in Figure 19. This figure shows that probability of false alarm exceeds 10% when the true separation distance is less than 9 nm for the case whee the lead aircraft is 500 s past the Center boundary.

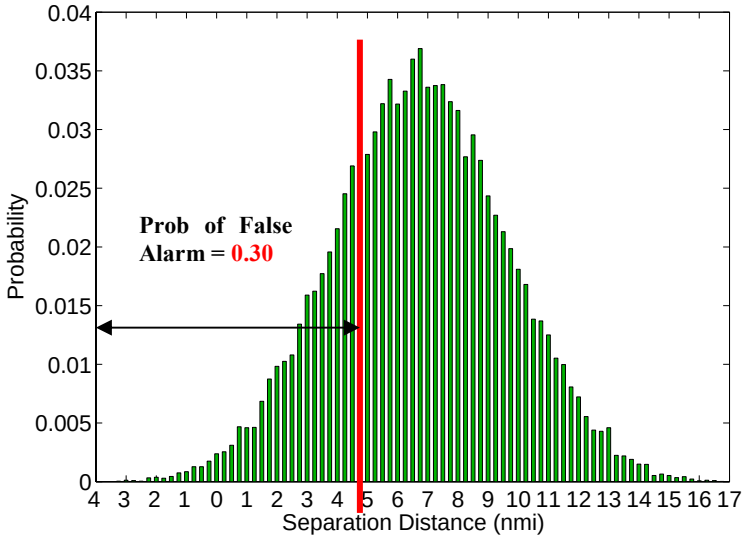

Figure 17 Separation frequency function for 2 cruise aircraft with a true separation of 7 nm when the leading aircraft is 500 s past the Center boundary and the trailing aircraft is 440 s past the Center

C Different Cruise Speed Scenario

The second set of truth trajectories has two aircraft flying at different cruise speeds with converging in-track spacing during cruise. Four speed differences were chosen to create conflicts at 200 sec, 300 sec, 400 sec and 500 sec into the flight. The initial separation is 7 nm and a conflict is defined a separation that is less than or equal to 5 nm.

Figure 20 shows the in-track and vertical separation dispersion relative to the leading aircraft for the case where the trailing aircraft true cruise speed is 24 kt greater than the leading aircraft leading to an actual loss of safe separation at 300 sec. While the dispersion bands include all 200 trials, it may be loosely associated with a ±3 sigma (99.7%) error bound. This figure

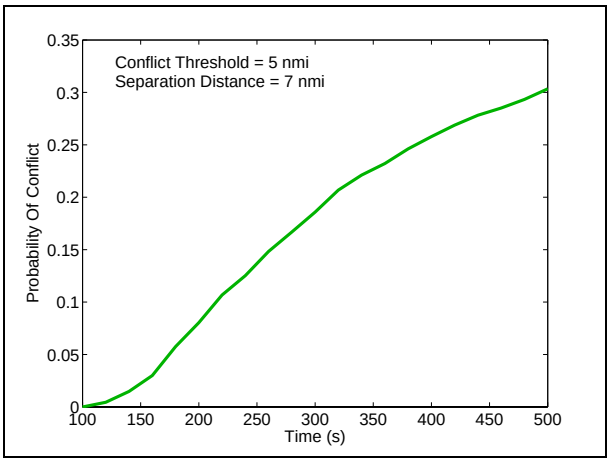

Figure 18 Probability of false alarm as a function of time for 2 aircraft with a true in track separation of 7 nm.

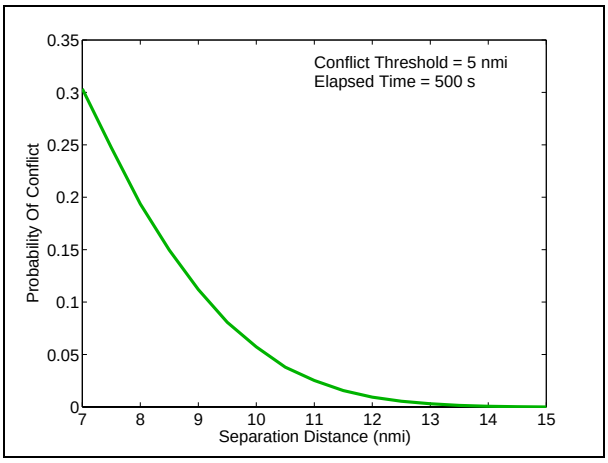

Figure 19 Probability of false alarm as a function of true in track separation distance when the lead aircraft is 500 s past the Center boundary.

also shows a simple rule for detecting conflicts with a horizontal green dashed line. The rule declares that a conflict occurs when the separation is less than or equal to 5 nm, *i.e.*, above the line.

The dark red line in the middle of the dispersion band is the true separation. The vertical green dashed line denotes when the true conflict first occurs. The missed alert and false alarm regions are labeled in the figure, as are the regions of true conflict and true absence of conflict.

The probability that the trailing aircraft is less than or equal to 5 nm behind the leading aircraft is shown in Figure 21 for four different closing speeds: 14 (turquoise), 18 (red), 24 (green), and 36 kts (blue). Figure 22, in turn presents the probability that the trailing aircraft is greater than 5 nm behind the leading aircraft for the same four closing speeds. Both of these probabilities are functions of prediction look ahead time and they also depend on the true cruise speed differences. The five-pointed stars in each of the figures indicate where the true conflicts first occur. The probability values in Figure 21 at earlier times (prior to the true conflicts indicated by the stars) are probabilities of false alarms. Similarly, the probability values in Figure 22 at later times are probabilities (after the true conflicts indicated by the stars) are missed alerts.

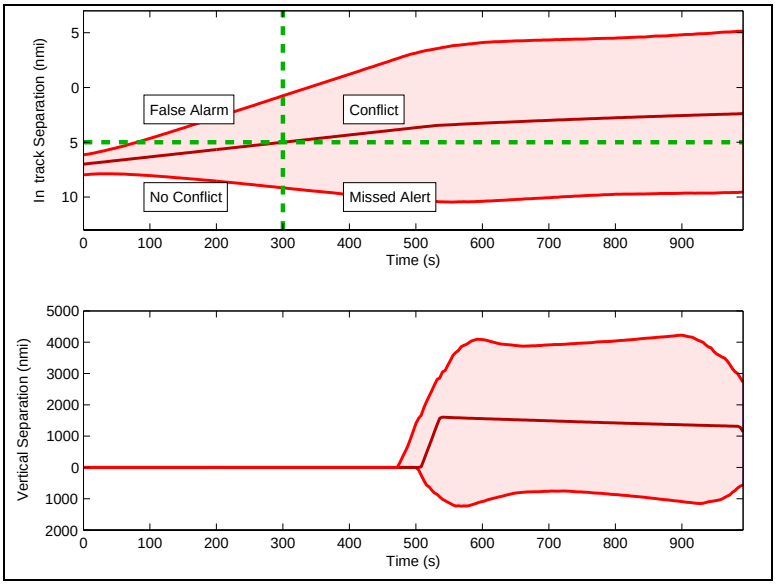

Figure 20 In-track and vertical separation dispersion relative to the leading aircraft when the trailing aircraft true cruise speed 24 kt greater than the leading aircraft true cruise speed

At the times that the true conflict occurs, both the probabilities of false alarms and the probabilities of missed alerts are near 0.5. This is a consequence of the symmetry of the input error distributions and of the fact that this analysis is focused on the cruise segment before descent. Figure 21 shows that a modest false alarm probability requirement of 10% constrains the prediction look-ahead time to significantly less than the true conflict time. Figure 22 shows that achieving a modest missed alert probability of 10% may be impossible for these cases.

These results show that errors in the inputs to the trajectory prediction process have a significant effect on the prediction of conflicts. Probabilities of false alarm and missed alerts are high due to these errors. Consequently, the prediction look-ahead times must be kept small.

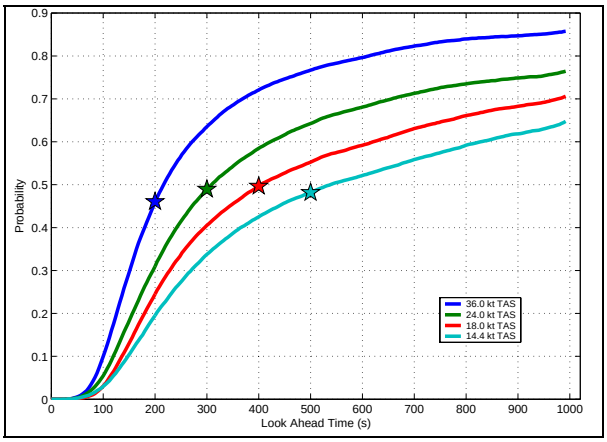

Figure 21 Probability that the trailing aircraft is ≤ 5 nm behind the leading aircraft for different closing speeds (14, 18, 24, and 36 kts)

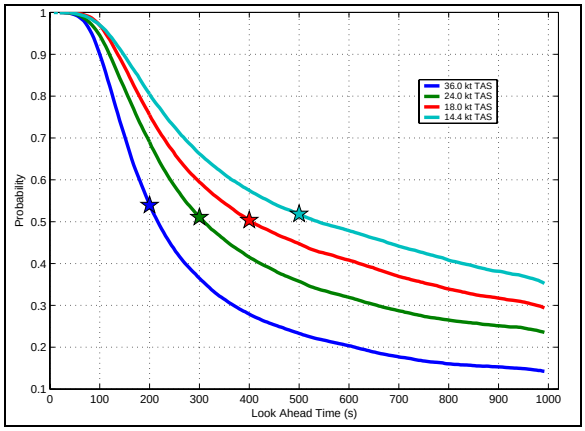

Figure 22 Probability that the trailing aircraft is > 5 nm behind the leading aircraft for various closing speeds (14, 18, 24, and 36 kts)

X Conclusion

From an EDA perspective, it is clear that the various trajectory errors identified and quantified in this study will affect the overall accuracy of the EDA advisories when applied to the real-world environment. In the current operational environment, the EDA design and operational paradigm must be adaptable to these errors.

One simple example of this is the sensitivity of meter fix crossing time errors as a result of errors in aircraft weight. From this analysis, it is clear that low estimates of aircraft weight can cause the aircraft to miss the speed and altitude constraints at the meter fix. High estimates of weight will not cause this effect. Therefore, it is better to err on the side of higher aircraft estimates of weight if one wants to minimize the risk of nonconformance with the meter fix crossing constraints. Of course, these errors have other implications to the overall performance of the aircraft model (e.g., the greater the assumed weight is higher than the actual weight, the lower the resulting

trajectory efficiency of the resulting EDA advisory will be). EDA should use the best possible estimate but it does give insight into the effect of weight on EDA and guidance on how to minimize its impact.

The other clear implication for EDA from this study is the identification and quantification of the impact of the parameters that have the most significant impact to the EDA system, namely the aircraft speed estimates and the wind speed estimates. New technologies that address these specific parameters can have a dramatic impact on EDA accuracies. For example, ADS-B has the potential for improving aircraft state data; FIS-B has the potential for improving the accuracy of weather data onboard the aircraft; and data link has the potential of supporting the provision of higher quality aircraft and weather data.

This study has also clearly identified the problems associated with false alarms and missed alerts resulting from the propagation of trajectory errors. Given the typical 20 minute trajectory look-ahead time frame that EDA examines (from entry into the center to arrival at the meter fix), this study has demonstrated a significant probability that the EDA conflict detection can produce both missed alerts and false alarms. Future development of EDA conflict detection algorithms need to look at incorporating the ramifications of the missed alert and false alarm probabilities into the conflict detection algorithm. This would then provide the controller a sense of how "real" the detected conflict is or, conversely, how far out in time will a "conflict-free" EDA advisory realistically be valid. This is an area that is in need of further study to determine appropriate EDA conflict detection look-ahead timeframes and alerting buffers based on desired false alarm and missed alert rates. It should be noted that CTAS, of which EDA is one of the decision support tools, is implementing conflict detection algorithms that consider conflict probabilities. However, this capability is still under development at this time.

Another important outcome of this study is the valuable development of the Matlab TS simulation. With this tool, there now exists a proven tool and methodology for performing stochastic sensitivity studies on en route cruise and descent trajectory parameters.

In addition, the Matlab TS simulation is a tool that is useful for independent CTAS TS verification and validation. During the development of the Matlab TS simulation, a number of unexpected TS behaviors were identified and brought to the attention of the CTAS TS developers. Some of these issues have resulted in CTAS TS improvements.

Acknowledgments

The authors wish to thank the En Route Operations branch of NASA Ames Research Center for funding this research. The authors appreciate the extensive discussions, technical direction and program management of Rich Coppenbarger of the En Route Operations branch of NASA Ames Research Center and the support and guidance received from Richard Lanier of the Ames FAA Liaison Office.. In addition, the authors also wish to thank the CTAS software group for their help with the CTAS code with a special thanks to William Chan for his help and guidance regarding the CTAS standalone TS. We would also like to acknowledge the support of the EDA software development group, Susan Dorsky and Jeff Gateley specifically, for their help in generating the EDA baseline trajectories used for this study.

References

[1]Coppenbarger, R.A., et al, " Real-Time Data Link of Aircraft Parameters to CTAS," 4TH USA/Europe ATM R&D Seminar, Santa Fe, NM, Dec 3-7, 2001

[2]Mueller, K.T.,& Weidner, T., " Air-Ground Data Link Study Plan," SeagullTechnology, Inc., 22 March 2000

[3]Williams, D.H., & Green, S.M., "Flight Evaluation of CTAS Trajectory Prediction Process, "NASA/TP-1998-208439, July 1998

[4]Weidner, T., Davidson, T.G., "Preliminary En Route Data Exchange Potential Benefits Assessment," Seagull Technology, TR98175.9-01, Dec 1998

[5]Hunter, G., Weidner, T., Couluris, G., Sorensen, J., Bortins, R., " CTAS Error Sensitivity, Fuel Efficiency, and Throughput Benefits Analysis," Seagull Technology TR96150-02, July 1996

[6]Mueller, K.T., "Final Approach Spacing Tool (FAST) Velocity Accuracy Performance Analysis," Seagull Technology, Inc., 98148-3-01, April 1998.

[7]Anon, *Approval of Area Navigation Systems for Use in the US National Airspace System*, Department of Transportation, Advisory Circular 90-45A, 21 February 1975 (Reprinted 1980)

[8]Weidner, T., Davidson, T.G., "Preliminary En Route Data Exchange Potential Benefits Assessment," Seagull Technology, TR98175.9-01. (Dec 1998)

[9]Williams, David H., and Green, Steven M., "Flight Evaluation of Center-TRACON Automation System Trajectory Prediction Process," NASA TP-1998-208439. (July 1998)

[10]Kayton, M., and Fried, W.R., "Avionics Navigation Systems," Second Edition, John Wiley & Sons. (1997)

[11]Green, S., Vivona, R., and Grace, M., "Field Evaluation of Descent Advisor Trajectory Prediction Accuracy for En route Clearance Advisories," AIAA GN&C Conference, AIAA-98-4479. (Aug 1998)
[12]Green, S., and Vivona, R., "Field Evaluation of Descent Advisor Trajectory Prediction Accuracy," AIAA GN&C Conference, AIAA96-3764. (July 1996)
[13]Benjamin, Stanley G., et al, "Aviation Forecasts From the RUC-2," 8th Conference on Aviation, Range, and Aerospace Meteorology. (Jan 1999)
[14]Cole, Rodney, E., et al, "An Assessment of the Errors in the 60 Km Rapid Update Cycle Wind Forecasts Relative to the Needs of the Center TRACON Advisory System, and an Assessment of the Increase in Accuracy due to Augmentation of the Wind Forecasts with near Real-time MDCRS data via the Integrated Terminal Weather System," Draft, MIT Lincoln Laboratory. (Jan 1998)
[15]Hunter, G., "Aircraft Trajectory Prediction Uncertainties and Design of Challenging DST Test Scenarios using Intelligent Challengers (Part I)," Seagull Technology, Inc., September 2003
[16]Mondoloni, S., Paglione, M., & Green, S., "Trajectory Modeling Accuracy for Air Traffic Management Decision Support Tools," ICAS 2002 Congress, 2002
[17]Slattery, R. and Zhao, Y., 1997, "Trajectory Synthesis for Air Traffic Automation," AIAA J. of Guidance, Control and Dynamics, v. 20, n. 2, pp 232-238
[18]Zhao, Yiyuan and Slattery, R.A., 1996, "Capture Conditions for Merging Trajectory Segments to Model Realistic Aircraft Descents," AIAA J. of Guidance, Control and Dynamics, v. 19, n. 2, pp 453-460
[19]Spiegel, M.R., "Statistics," Schaum Publishing Co., 1961
[20]Coppenbarger, R., Lanier, R, Sweet, D., Dorsky, S., "Design and Development of the En Route Descent Advisor (EDA) for Conflict-Free Arrival Metering," AIAA GN&C Conference, AIAA-4875 (August 2004)