JOURNAL OF SPACECRAFT AND ROCKETS
Vol. 31, No. 4, September–October 1994

Functional Dependence of Trajectory Dispersion on Initial Condition Errors

Robert A. LaFarge* and Roy S. Baty†
Sandia National Laboratories, Albuquerque, New Mexico 87185

This article proposes numerical techniques to approximate dispersion bounds and burst patterns for Monte Carlo trajectory simulations. The algorithms developed approximate trajectory dispersion bounds and burst patterns caused by the errors in initial conditions in 1/100th of the computational expense of full Monte Carlo analyses. The proposed techniques are based on the properties that the six-degree-of-freedom equations of motion produce solutions that vary continuously with initial conditions and preserve the statistical distribution of the initial conditions. The continuity of solutions in initial conditions is studied numerically by performing a stability analysis. Numerical experiments simulating a fuse effectiveness study for two generic re-entry bodies are exhibited. The dispersion bounds and burst patterns predicted using the proposed algorithms are compared to the dispersion bounds and burst patterns predicted using full Monte Carlo simulations. The agreement is excellent.

Nomenclature

B = $N \times N$ transformation
$C_{\delta X}$ = Position error covariance matrix
$C_{\bar{\nu}}$ = Covariance matrix
Df = Jacobian of f
e_i = Unit basis vector
E = Expected value function
f = Function defining the ordinary differential equation
g = Probability density
h = Coordinate transformation
K, M = Real constants
P_I = Probability of impact
P_K = Probability of kill
t = Time variable
w = N-dimensional independent random variables
X_i = Eigenvalue axis
x = Solution of the ordinary differential equation
x_0 = Initial condition for the ordinary differential equation
α = Angle of attack
Δ = Reference to the nominal
ε = Small perturbation
λ = Random vector
λ_i = Eigenvalue of B
μ = Random vector
μ_x, μ_y = Mean values
ν = N-dimensional vector of random variables
σ_{xy} = Covariance
σ_i = Standard deviation
ρ = Spectral radius
Φ = Fundamental matrix for the linearized ordinary differential equation

Introduction

THE solution of the six-degree-of-freedom (6DOF) equations of motion is an essential part of investigating the flight behavior of vehicles. Many trajectory codes[1–3] have been written over the years that produce reasonable trajectory simulations for ballistic flight vehicles. For most applications, only a limited number of trajectories are computed. Examples using a limited number of solutions include investigations of resonance conditions, predictions of the nominal downrange and crossrange values, and simulations of water impact conditions. There are applications, however, that rely on multiple trajectory solutions to answer the investigator's questions. Two examples of these applications are determining the probability (P_I) of explosively formed debris impacting a predefined exclusion area in a range safety study and determining the probability of kill (P_k) for a particular fuse type in a fuse effectiveness study. Trajectories for these applications are usually studied using Monte Carlo techniques, which use statistical information about the vehicle and about its flight conditions to produce a set of initial conditions for a trajectory code. The statistical information can be either uncorrelated, as in the case of vehicle mass properties and the fuse error models, or correlated, as in the case of the booster-induced position and velocity errors at re-entry defined by the pre-re-entry covariance. At Sandia National Laboratories (SNL), the preprocessing code MCPRAM[4] produces a set of input files for the AMEER[1] trajectory code to use in fuse effectiveness studies. For these studies, the trajectories are computed to the fuse burst point. Figure 1 illustrates the typical MCPRAM results of a study on a timer fuse. The particular weapon yield and target hardness would produce a peak over pressure curve, as illustrated by the dashed curves. The symbols inside the curves represent burst points that eliminate the target. Computing P_k for this combination of booster, weapon, and target simply requires counting the number of burst points inside the curves.

For these studies to be significant statistically, approximately 1000 trajectories must be computed. This is very time consuming and quite expensive computationally, even if cost-cutting techniques, such as incorporating different fuses on the same trajectory, are used. If an investigation is in its early stages, the investigator could be satisfied with approximate results if these can be obtained quicker and at a lower cost than full Monte Carlo results. It is in this category of estimation that the techniques proposed in this paper are to be applied.

In the present article, techniques for estimating dispersion bounds and burst patterns based on computing a few trajectories are proposed. These approaches are developed to approximate trajectory dispersion bounds and burst patterns caused by initial condition errors without the computational expense of Monte Carlo simulations. The proposed methods are based on the properties that the equations of motion produce solutions that vary continuously with initial conditions and preserve the statistical distribution of the initial conditions. To study the continuity of solutions in initial conditions, a perturbation equation, linearized about the nominal trajectory, is derived. Numerical experiments that simulate a fuse effectiveness study for a timer fuse are used to investigate the algorithms and the perturbation equation for two generic re-entry bodies (RBs). The

Received March 27, 1993; revision received Oct 18, 1993; accepted for publication Nov. 24, 1993.

*Senior Member of Technical Staff, Intelligent System Principles Department. Senior Member AIAA.
†Senior Member of Technical Staff, Aero Sciences and Fluid Dynamics Department.

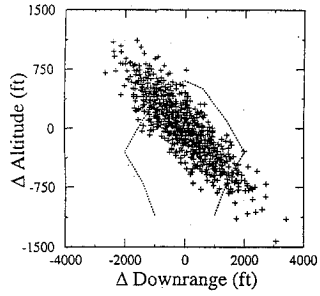

Fig. 1 A downrange/altitude probability of kill fuse study.

dispersion bounds and burst patterns predicted by these experiments are then compared to the results of full Monte Carlo simulations.

Trajectory Dispersion Algorithm

This section outlines the fundamental mathematical ideas needed to develop the trajectory dispersion algorithm. The basic notion applied in the dispersion algorithm is that ordinary differential equations (ODEs) yield solutions that are continuous functions of their initial conditions.

Continuity and Initial Conditions

To examine how the solutions of ODEs depend on initial values, consider the arbitrary initial value problem:

$$\frac{dx}{dt} = f(t, x) \tag{1}$$

and

$$x(0) = x_0 \tag{2}$$

Here, Eqs. (1) and (2) represent an n-dimensional system of ordinary differential equations with the initial condition vector x_0.

A simple, one-dimensional example of the initial value problem defined by Eqs. (1) and (2) is given by

$$\frac{dx}{dt} = x \tag{3}$$

and

$$x(0) = 2.0 \tag{4}$$

By integrating Eq. (3) and applying the condition of Eq. (4), the solution of this example problem can be shown to be

$$x(t) = 2.0e^t \tag{5}$$

Now, to see how the solution of the model problem is a continuous function of its initial value, suppose the initial value, Eq. (4), is perturbed by a small number say, ε; yielding a perturbed initial value:

$$x(0) = 2.0 + \varepsilon \tag{6}$$

Then the solution of the perturbed initial value problem becomes

$$x(t) = (2.0 + \varepsilon)e^t \tag{7}$$

Equation (7) shows that a small perturbation of the initial value causes a small perturbation of the solution. This observation is exactly what is meant when an initial value problem is said to be a continuous function of its initial conditions.

The arbitrary initial value problem defined by Eqs. (1) and (2) will also have solutions that are continuous functions of their initial conditions if certain theoretical conditions are met. The main assumption needed to ensure that solutions of Eqs. (1) and (2) vary continuously with the initial conditions is that the function f defining the system of ODEs satisfy a Lipschitz condition in x for all (t, x) in some rectangular domain of interest. Here, a function is said to satisfy a Lipschitz condition if

$$|f(t, x) - f(t, y)| \leq K|x - y| \tag{8}$$

where K is a constant, and x, y are vectors in the domain of interest. A standard theorem[5] in the theory of ODEs states that the solution of the initial value problem defined by Eqs. (1) and (2) is a continuous function of the initial conditions if the system defined by f satisfies the Lipschitz condition given in Eq. (8).

Assuming that the system defining the differential equation satisfies the Lipschitz condition does not restrict the class of trajectory problems that can be considered with the algorithm outlined in this report. The condition in Eq. (8) guarantees that the initial value problem defined by Eqs. (1) and (2) has a unique solution. Therefore, without an assumption similar to that expressed in Eq. (8), it is futile to try to simulate trajectories with initial value problems.

Stability of the Equations of Motion

Because solutions of the initial value problem defined by Eqs. (1) and (2) are continuous in initial conditions, the problem of developing a trajectory dispersion scheme reduces to determining the long-time solution behavior with perturbed initial conditions. To this end, the variation of the solutions of Eq. (1) as a function of small perturbations ε of the initial condition

$$x(0) = x_0 + \varepsilon \tag{9}$$

will be studied. This amounts to studying the stability of the system of Eq. (1).

To analyze the stability of the system of Eq. (1), a linear perturbation equation is developed. Let the function f on the right-hand side of Eq. (1) be analytic, and let x_0 be an initial condition of interest. Then the equation for a small perturbation of x_0 becomes

$$\frac{d}{dt}(x_0 + \varepsilon) = f(t, x_0 + \varepsilon) \tag{10}$$

Expanding f in a Taylor series about x_0 and combining the result with Eq. (10) yields

$$\frac{d}{dt}(x_0 + \varepsilon) = f(t, x_0) + Df(t, x_0)\varepsilon + N(t, x_0, \varepsilon) \tag{11}$$

where Df is the Jacobian of f. Then, subtracting Eq. (1) from Eq. (11) produces a nonlinear perturbation equation:

$$\frac{d\varepsilon}{dt} = Df(t, x_0)\varepsilon + N(t, x_0, \varepsilon) \tag{12}$$

where N represents the nonlinear terms in the Taylor expansion. Finally, neglecting the nonlinear terms in Eq. (12) produces a linear perturbation equation for the system given by Eq. (1):

$$\frac{d\varepsilon}{dt} = Df(t, x_0)\varepsilon \tag{13}$$

Typical mathematical assumptions necessary to neglect N and produce Eq. (13) are discussed in Ref. 6.

Equation (13) is a linear system approximating the behavior of small perturbations of Eq. (1). The problem of studying the continuity of solutions of Eq. (1) in initial conditions is then reduced to studying the stability of the linear system, given in Eq. (13). In this case, the stability of the linear system [Eq. (13)] about the equilibrium point

$$\varepsilon = 0 \tag{14}$$

is analyzed. The stability problem defined by Eqs. (13) and (14) is used to describe the time history of perturbations from the nominal trajectory, with the initial value x_0.

For the present discussion, a solution ε of Eq. (13) is defined to be stable if a small perturbation of the initial value associated with ε causes a small perturbation of the solution. Now, to show that the linear system [Eq. (13)] is stable about the zero equilibrium point, let Φ denote the fundamental matrix[7] of the linear system. The solution of Eq. (13) for an initial value ε_0 is then given by

$$\varepsilon(t) = \Phi(t, x_0)\varepsilon_0 \tag{15}$$

From a standard result in the theory of ordinary differential equations,[6] it follows that solutions are stable when there is a number M such that

$$\|\Phi(t, x_0)\| \leq M \tag{16}$$

for $t \geq t_0$, where t_0 is the initial time, and $\| \|$ is a matrix norm.

Dispersion Algorithm

The basic idea of the proposed trajectory dispersion algorithm is to estimate dispersion bounds by computing a few trajectories for extreme points bounding the sets of initial conditions used in Monte Carlo simulations. This algorithm has three steps: 1) given the initial conditions for a Monte Carlo simulation, determine a small set of initial conditions that bound the Monte Carlo initial conditions; 2) compute trajectory simulations for the nominal initial condition and for the small set of initial conditions constructed in step 1; and 3) at the burst location, construct a ball centered at the nominal burst point with the minimum diameter necessary to contain all the burst positions of the trajectories calculated with the small set of initial conditions from step 1. The diameter of this ball will be the maximum value of the distances from the nominal burst point to burst positions computed in step 2. This diameter provides an approximation of the dispersion bound for the given set of Monte Carlo initial conditions.

The values of the initial conditions used to bound the set Monte Carlo initial conditions are referred to as the "corners" of the initial conditions. In the "Numerical Experiments" section the dispersion bounds for two RBs are computed using this algorithm. In both of these examples, eight corners are used to estimate the dispersion bounds. Therefore, including the nominal trajectory, each numerical example required a total of nine trajectory simulations.

It should be noted that this algorithm only provides an estimate of the dispersion bound, because it is theoretically possible for the Monte Carlo initial conditions to yield burst points that lie outside of the ball containing the "corner" burst points. However, the trajectory dispersion algorithm produces a very good approximation of the dispersion bound, because the system of differential equations governing the motion of vehicles yield solutions that are continuous in initial conditions. The example RB calculations produced dispersion bounds that contained all the Monte Carlo burst positions.

The trajectory dispersion algorithm is applicable to any aerovehicle that has analytic aerodynamic coefficients. However, the algorithm has only been applied to high and medium performance re-entry vehicles. If another class of vehicles is considered, it is recommended that the stability of the equations of motion be studied. If the new class of vehicles yield trajectories in which solutions of the linear stability equation do not seem to be bounded in time, the trajectory dispersion algorithm may not produce useful results.

Flight Statistics

This section discusses the pre-re-entry covariance used to model the initial condition errors in Monte Carlo fuse effectiveness studies. This covariance defines errors for the velocity and position vector components at re-entry. The six variables (three velocity components and three position components) described by the pre-re-entry covariance are correlated. Correlated variables are simply variables whose values depend on each other. Using correlated variables in a Monte Carlo analysis requires that a transformation (and its inverse) of the correlated variables exist, so that the variables in the new space are independent of each other. A random number generator can be used to generate independent random variables. The inverse transformation is used on the independent random variables to produce the random correlated variables. Fortunately, the existence of such a transformation can be proved.[4] Before the transformation can be discussed, correlated variables and a covariance matrix must be defined statistically. All random variables are assumed to have a normal distribution.

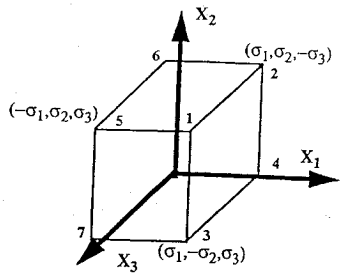

Fig. 2 Corners of the eigenspace.

Covariance Definition

In statistical terms, if x and y are random variables and $E[x]$ is the expected value function,[8] the covariance σ_{xy} is defined by

$$\sigma_{xy} = E[(x - E[x])\,(y - E[y])] \tag{17}$$

If μ_x and μ_y represent the expected values (means) of the random variables x and y, the covariance is simplified to

$$\sigma_{xy} = E[xy] - \mu_x \mu_y \tag{18}$$

If x and y are independent (uncorrelated), Eq. (18) is exactly zero, and if x and y are correlated, then σ_{xy} is nonzero. The variance σ_x^2 of the random variable x, which is the square of the standard deviation, is defined by

$$\sigma_x^2 = E[(x - \mu_x)^2] \tag{19}$$

If ν represents a vector of n random variables $(x_1, \ldots, x_n)^T$, a covariance matrix C_ν on ν can be defined by

$$C_\nu = E[\nu, \nu^T] = \begin{bmatrix} \sigma_{x_1}^2 & \cdots & \sigma_{x_1 x_n} \\ \vdots & \vdots & \vdots \\ \sigma_{x_n x_1} & \cdots & \sigma_{x_n}^2 \end{bmatrix} \tag{20}$$

Generating the correlated random numbers $x_1, \ldots, x_n$ for a Monte Carlo analysis requires that a transformation exist so that

$$\nu = Bw \tag{21}$$

where w represents a set of n independent variables $y_1, \ldots, y_n$. The eigenvector matrix of C_ν represents such a transformation, and the square roots of the eigenvalues of this transformation λ_i are the standard deviations σ_i of the independent variables $y_1, \ldots, y_n^4$, where $i = 1, \ldots, n$:

$$\sigma_i = \sqrt{\lambda_i} \tag{22}$$

Because these variables are independent, each can be produced with a normally distributed random number generator.[9] After w is completed, the correlated random numbers are formed using Eq. (21).

Position Error Covariance

The pre-re-entry covariance is a six-by-six matrix that defines the booster induced errors to the velocity and position vector components at re-entry in a downrange and crossrange coordinate system.[4] These errors are referenced to the nominal re-entry time, which corresponds to the initial time for the trajectory computations in this article. Consequently, the random errors produced by this covariance will be used as the source of the initial condition errors. So that the results may be better isolated, a covariance that produces only the position errors is used, in conjunction with only a timer fuse. This position error covariance $C_{\delta X}$ corresponds to the lower right-hand corner of the pre-re-entry covariance. The elements of the covariance have been generated randomly and represent no particular booster system.

Initial Condition Corners

The eigenvectors and eigenvalues of $C_{\delta X}$ are also used to generate the "corners" of the eigenspace. These corners are to be used to define the burst point dispersion instead of a Monte Carlo simulation. They are defined by the one standard deviation, Eq. (22), lengths of the eigenvectors of $C_{\delta X}$. Some of the values of the corners along with the numbering scheme used in this paper are shown in Fig. 2. The axes (X_1, X_2, X_3) represent the three eigenvectors of $C_{\delta X}$.

The position error covariance models the initial condition errors for the Monte Carlo experiments and, as such, describes the initial conditions for the corner trajectories that estimate the Monte Carlo dispersion. Before the results on the Monte Carlo experiments are discussed, the stability of the 6DOF equations applied to specific vehicles is investigated. It is the stability of the equations of motion that will allow the bound of the dispersion of the burst points to be approximated by only a few trajectories.

Numerical Experiments

To observe the effects of aerodynamic stability and nonlinearity, two RBs are investigated. One is a medium weight, 900 lb, vehicle with moderate aerodynamic stability, and the other is a heavier, 1300 lb, less aerodynamically stable vehicle. The predicted angle of attack history α for the nominal trajectory for each vehicle is shown in Figs. 3 and 4. Verification of the procedure outlined in this article requires that the stability of the 6DOF equations of motion applied to the two RBs studied be determined. To prove that the corners of the eigenspace encompass the 1-σ burst points, a two-part Monte Carlo experiment is computed for each vehicle. The first part is an unbiased experiment where no attempt is made to control the random numbers. This shows a typical burst point distribution. The second

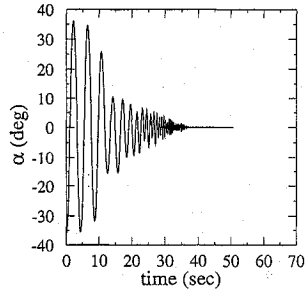

Fig. 3 Angle of attack time history for the first RB.

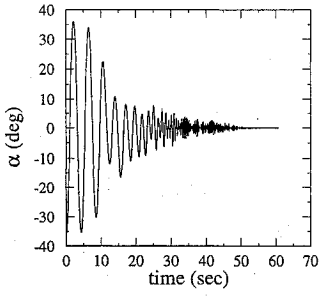

Fig. 4 Angle of attack time history for the second RB.

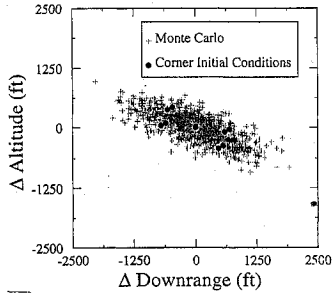

Fig. 5 Initial conditions, downrange/altitude.

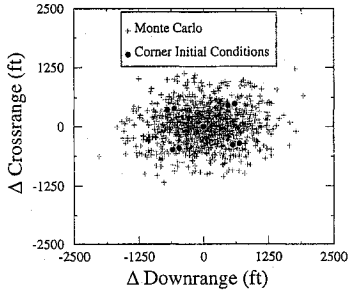

Fig. 6 Initial conditions, downrange/crossrange.

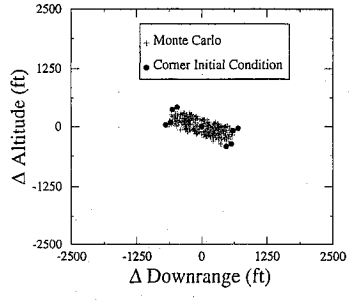

Fig. 7 Initial conditions (1-σ) downrange/altitude.

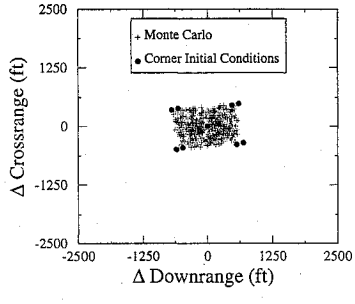

Fig. 8 Initial conditions (1-σ) downrange/crossrange.

part of the experiment only looks at those initial condition errors from the first part of the experiment produced by random numbers one standard deviation or less. Because for each eigenvector this restriction would allow theoretically 68% of the random numbers, only 32% $[(0.68)^3]$ of the total 1000 initial condition errors should be expected to fall in this category. In fact, 357 initial condition errors of length 1-σ or less were produced. The initial condition errors produced for each part are shown in Figs. 5–8.

Stability Experiments

To analyze the stability of small perturbations about the nominal trajectory, Eq. (13) is integrated for arbitrary initial conditions. This requires the computation of the Jacobian at each time step, $t + \Delta t$, along the trajectory. The elements of the Jacobian, a 13-by-13 matrix, are computed analytically where possible; otherwise, a second order leapfrog[9] method is used to compute the partial derivatives.

The general solution of Eq. (13) is obtained by computing the fundamental matrix. Then, for arbitrary initial conditions, the solution of Eq. (13) is given by Eq. (15). Because all solutions of Eq. (13) are necessarily linear combinations of the columns of the fundamental matrix,[7] the fundamental matrix may be evaluated numerically by integrating Eq. (13) for each of the following initial conditions:

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \cdots, \quad e_{13} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{23}$$

These integrals are then used to construct the fundamental matrix

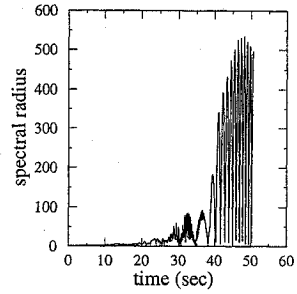

Fig. 9 Spectral radius time history, first RB.

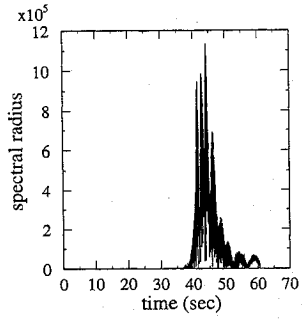

Fig. 10 Spectral radius time history, second RB.

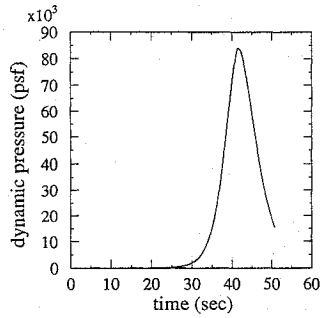

Fig. 11 Dynamic pressure time history, first RB.

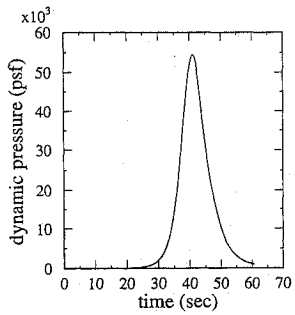

Fig. 12 Dynamic pressure time history, second RB.

at each time step along the trajectory.

Once the fundamental matrix is computed, it must be demonstrated to be bounded; that is, Eq. (16) must be satisfied for all $t \geq t_0$. A theorem in matrix analysis[10] states that for a matrix Φ and a real $\varepsilon > 0$, there exists a matrix norm denoted by $\| \ \|$ so that

$$\rho(\Phi) \leq \|\Phi\| \leq \rho(\Phi) + \varepsilon \tag{24}$$

where $\rho(\Phi)$ is the spectral radius of Φ. Here, the spectral radius is defined by

$$\rho(\Phi) \equiv \max\{|\lambda|: \ \lambda \text{ is an eigenvalue of } \Phi\} \tag{25}$$

To show that the fundamental matrix is bounded, the spectral radius is used to approximate the value of a matrix norm.

The spectral radius histories for each vehicle are shown in Figs. 9 and 10. The less stable vehicle, Fig. 10, produced a much larger, though still bounded, spectral radius. The associated dynamic pressures q, which drive the fluctuations in the spectral radius, are illustrated by Figs. 11 and 12. The difference in the maximum magnitude of the spectral radius is a function of the difference in the aerodynamics of the two RBs. The results of this investigation suggest that the 6DOF equations of motion are well behaved in initial conditions for these two vehicles. Furthermore, the results suggest that the proposed algorithm should provide a good estimate of trajectory dispersion bounds for vehicles similar to re-entry bodies. The comparison of the Monte Carlo simulations with the proposed method follows in the next section.

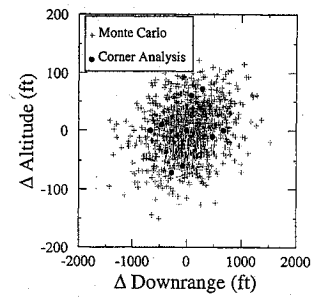

Fig. 13 Full Monte Carlo results for the first RB, downrange/altitude.

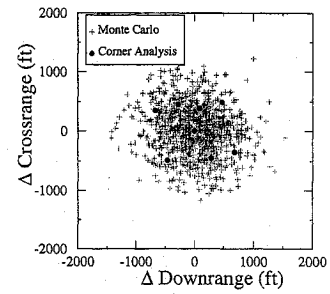

Fig. 14 Full Monte Carlo results for the first RB, downrange/crossrange.

Monte Carlo Experiments

Because the 6DOF equations of motion are stable about the nominal trajectory for both RBs, a set of trajectories based on the initial condition errors, illustrated by Figs. 5 and 6, are computed for each RB. These are then used to produce the typical burst point distribution for a Monte Carlo experiment for a timer fuse with no mechanization errors. For each RB, the nominal trajectory is computed to a reference altitude, and the time of flight is used as the timer fuse setting for the Monte Carlo experiment. Because the second vehicle is less stable, it flew with a coning angle and took 10 s longer to reach the reference altitude. The results of the experiments, which are referenced to the nominal trajectory, are illustrated in Figs. 13 through 16, together with the results from computing the burst points of the corners. As expected from the random number distribution, the burst points computed from the corners are contained within the distribution of the full Monte Carlo experiment.

The second half of each experiment investigates those trajectories with initial condition errors generated by random numbers less than 1-σ magnitude in the eigenspace, Figs. 7 and 8. The burst points are compared to the maximum burst dispersion predicted by the scheme proposed in this paper. As can be seen in Figs. 17–20, the burst points from each modified Monte Carlo experiment are contained within the dispersion predicted by the proposed scheme, which is represented by the circle. Although the maximum spectral radius for the second vehicle was orders of magnitude greater than the maximum of the first vehicle, the maximum dispersion bounds predicted by the present method still contained the Monte Carlo data. Furthermore, the results suggest that a more refined fit to the Monte Carlo data using the corner burst points can be made.

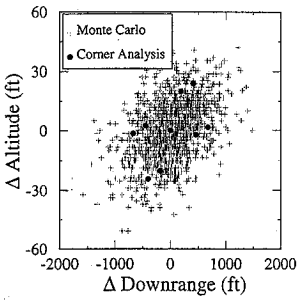

Fig. 15 Full Monte Carlo results for the second RB, downrange/altitude.

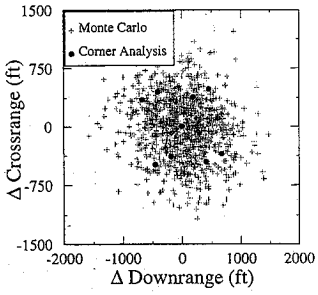

Fig. 16 Full Monte Carlo results for the second RB, downrange/crossrange.

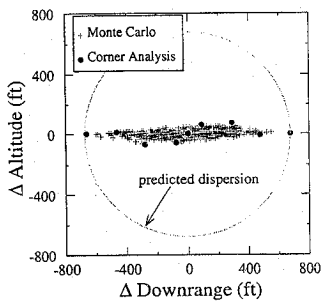

Fig. 17 One-σ results for the first RB, downrange/altitude.

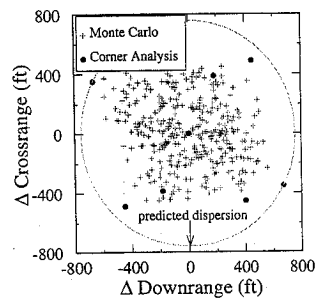

Fig. 18 One-σ results for the first RB, downrange/crossrange.

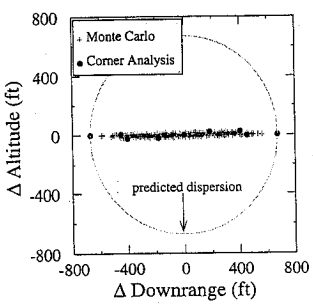

Fig. 19 One-σ results for the second RB, downrange/altitude.

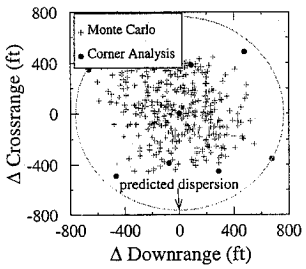

Fig. 20 One-σ results for the second RB, downrange/crossrange.

Discussion of Numerical Experiments

The numerical experiments show that the proposed algorithm estimated the bounds for trajectory dispersion caused by initial condition errors extremely well. Note that the dispersion of the corner burst locations, when compared with their initial locations, suggests that the trajectories are converging to the nominal trajectory. This result could be analyzed by applying theorems on the asymptotic stability of ordinary differential equations. However, the observed convergence of the perturbed trajectories to the nominal trajectories was beyond the scope of this study.

The numerical experiments in the present paper only suggest that the algorithm will provide reasonable estimates of the dispersion bounds. These numerical experiments *do not prove* that this technique will work for an arbitrary vehicle. For example, if a vehicle exhibits a nonanalytic hysteresis behavior, this method will fail to provide useful dispersion bounds. For suspect vehicles, an investigation of the stability of the equations of motion is highly recommended. However, for vehicles with well-behaved aerodynamics, such as high-performance RBs, the results from the experiments suggest that the stability analysis is not necessary.

Burst Covariance Algorithm

The numerical experiments outlined in the previous section show that the corner initial conditions can be used to approximate dispersion bounds accurately for statistical trajectory studies without running full Monte Carlo simulations. Moreover, further study of the 1-σ dispersion plots suggests that the corners may also preserve information about the statistics at the burst location.

As observed in Figs. 21 and 22, rescaled burst locations for the two RBs, the correlated nature of the dispersion pattern with respect to the corner positions at the burst location suggests that the statistical properties of the initial conditions are conserved. Fundamental theorems in the theory of stochastic differential equations exist that give conditions for the statistical properties of the initial conditions to be conserved at the burst points. A theorem[11] relating random initial conditions with solutions of deterministic ordinary differential equations states that the probability density associated with the initial conditions of an initial value problem is conserved in time and varies as a function of solutions of the underlying ordinary differential equation. This is illustrated with the initial value problem defined by Eqs. (1) and (2), which model the flight of re-entry bodies. Let

$$g(\boldsymbol{\lambda}) = g(\lambda^1, \lambda^2, \lambda^3, \ldots, \lambda^{13}) \tag{26}$$

denote a probability density of the initial conditions given by Eq. (2). Here the vector $\boldsymbol{\lambda}$ represents a value of the random variable x_0, Eq. (2), defining the initial conditions. Furthermore, let μ denote a vector that the solution of the initial value problem could obtain at the burst point as a function of initial conditions. Then the probability at the burst point is related to the probability of the initial conditions by the equation

$$g(\mu^1, \mu^2, \mu^3, \ldots, \mu^{13}, t) = g(\lambda^1, \lambda^2, \lambda^3, \ldots, \lambda^{13}) \times \left| \frac{\partial(h^1, \ldots, h^{13})}{\partial(\mu^1, \ldots, \mu^{13})} \right| \tag{27}$$

where

$$h^i = (x^i)^{-1} \text{ and } \lambda^i = h^i(\mu, t) \text{ for } i = 1, \ldots, 13$$

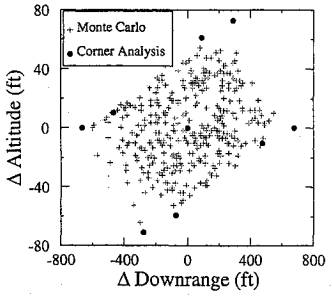

Fig. 21 Rescaled 1-σ burst locations for the first RB.

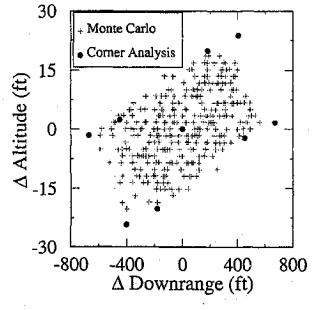

Fig. 22 Rescaled 1-σ burst locations for the second RB.

In Eq. 27,

$$\left|\frac{\partial(h^1,\ldots,h^{13})}{\partial(\mu^1,\ldots,\mu^{13})}\right| \tag{28}$$

is the Jacobian of the inverse function, denoted by the vector $\boldsymbol{h}$, of the solution vector $\boldsymbol{x}$ to the ordinary differential equation given by Eq. (1).

Equation (27) indicates that the basic statistical properties associated with the initial value problem describing the motion of vehicles with smooth aerodynamic coefficients are conserved along trajectories in time. For example, the expected value function—a generalization of the expected value used in the flight statistics section—is defined mathematically by

$$E[y_{\text{Burst}}] = \int_{S(t)} \mu g(\mu, t)\,\mathrm{d}\mu = \int_{S(0)} x(\lambda, t) g(\lambda)\,\mathrm{d}\lambda \tag{29}$$

where t is the burst time and $S(t)$ is the set integrated over as a function of burst time. Here $S(0)$ represents the initial condition set, and $S(t)$ represents the burst position set. Therefore, because the probability density is conserved by solutions to the equations of motion, the covariance properties used to model the initial conditions are conserved at the burst location in the sense of Eq. (29).

Burst Covariance Prediction

Assuming the statistical properties of the initial conditions for a vehicle are preserved along trajectories, the corner and nominal trajectories used in the dispersion algorithm can be applied to approximate position covariance matrices and burst dispersion patterns. This section outlines a numerical algorithm that estimates these quantities based on the corner and nominal trajectories. The dispersion patterns at the burst locations are then computed for the two re-entry vehicles previously studied and compared with the numerical results computed from the full Monte Carlo simulations.

The trajectory burst pattern algorithm has five steps: 1) given the nominal and corner burst positions, vectors are computed from the nominal burst point to each burst corner; 2) observations are created and stored along the vectors generated in step 1 using a Gaussian distribution; 3) the stored observations from step 2 are used to compute a position covariance matrix applying Eqs. (18) and (19); 4) eigenvalues and eigenvectors are computed using this position covariance matrix; and 5) the positions of 1000 burst points are calculated using the results of step 4 in Eqs. (21) and (26). Furthermore, step 3 of this algorithm is performed with the assumption that the expected value function is approximated by the sample mean.

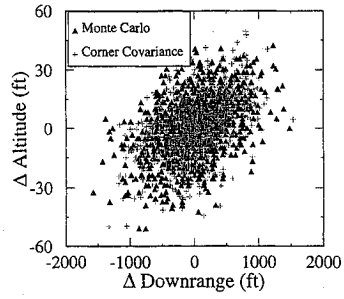

Fig. 23 Comparison of algorithm and MC burst patterns for the first RB.

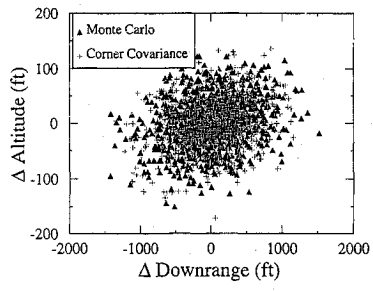

Fig. 24 Comparison of algorithm and MC burst patterns for the second RB.

The dispersion patterns at the burst location computed with the trajectory burst pattern algorithm are compared with the direct output of the full Monte Carlo simulations. Figure 23, a downrange/altitude plot, shows the approximated dispersion pattern and the Monte Carlo (MC) dispersion pattern for the first RB. Figure 24 compares these two dispersion patterns for the second RB.

The agreement of the burst dispersion patterns in Figs. 23 and 24 is quite good, because the approximated patterns cluster directly on top of the Monte Carlo simulations for both re-entry bodies. Recall that the approximated burst patterns required the computation of nine trajectories, whereas the Monte Carlo simulations required 1000. Further detail on these numerical results can be found in Ref. 13.

Conclusions and Recommendations

This article has developed numerical techniques to approximate dispersion bounds and burst patterns for statistical trajectory simulations without the time and expense of full Monte Carlo analyses. These techniques use a few initial conditions to estimate properties at the burst location and require 1/100th of the computational expense of a Monte Carlo analysis. Numerical experiments performed with these schemes for two generic re-entry bodies show that these methods accurately approximated the dispersion bounds and burst patterns predicted by full Monte Carlo simulations. Moreover, the numerical experiments demonstrated that the 6DOF equations of motion for these re-entry bodies yielded solutions that were continuous in initial conditions.

Although the algorithms developed in this paper are based on general mathematical results and worked very well for the RBs studied, this paper *did not prove* that these algorithms will work for an arbitrary vehicle. It is, therefore, recommended that stability analyses and full Monte Carlo simulations be applied to vehicles for which the flight performance is not well understood or exhibits a nonanalytic hysteresis behavior. If a large number of full Monte Carlo simulations are planned for a specific class of vehicles, the trajectory dispersion and burst pattern algorithms could be applied in conjunction with example Monte Carlo simulations to determine average flight performance.

Acknowledgments

This work was performed at Sandia National Laboratories under the U. S. Department of Energy contract DE-AC04-94AL85000. The authors thank Don H. Tucker of the University of Utah for his technical advice on the theory of ordinary differential equations. The

authors also thank Albert E. Hodapp, Jr. and Harold R. Vaughn (retired), of Sandia National Laboratories, for their helpful discussions on the flight characteristics of re-entry bodies. Michael E. Senglaub of Sandia National Laboratories requested this study.

References

[1]Meyer, E. J., "A User's Manual for the AMEER Flight Path-Trajectory Simulation Code," Sandia National Lab., SANDO80-2056, Albuquerque, NM, Dec. 1984.

[2]LaFarge, R. A., "A Description of the Reference Frames and the Coordinate Transformations Used in AMEER," Sandia National Lab., SAND88-0243, Albuquerque, NM, Reprinted Oct. 1992.

[3]Brown, R. C., Brulle, R. V., Combs, A. E., and Griffin, G. D., "Six-Degree-of-Freedom Flight-Path Study Generalized Computer Program," Air Force Flight Dynamics Lab., FDL-TDR-64-1, Wright-Patterson AFB, OH, Oct., 1964.

[4]LaFarge, R. A., "A Users' Manual for MCPRAM and for the Fuze Options in AMEER," Sandia National Lab., SAND90-0483, Albuquerque, NM, May 1990.

[5]Hirsch, M. W., and Smale, S., *Differential Equations, Dynamical Systems, and Linear Algebra*, Academic Press, New York, 1974.

[6]Piccinini, L. C., Stampacchia, G., and Vidossich, G., *Ordinary Differential Equations in R^n*, Springer–Verlag, New York, 1984, pp. 346–351.

[7]Sanchez, D. A., *Ordinary Differential Equations and Stability Theory: An Introduction*, Dover Publications, New York, 1968.

[8]Bowker, A. H., and Lieberman, G. J., *Engineering Statistics*, Prentice-Hall, Englewood Cliffs, NJ, 1972.

[9]Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., *Numerical Recipes*, Cambridge Univ. Press, Cambridge, England, UK, 1986.

[10]Horn, R. A., and Johnson, C. R., *Matrix Analysis*, Cambridge Univ. Press, Cambridge, England, UK, 1985.

[11]Saaty, T. L., *Modern Nonlinear Equations*, Dover Publications, New York, 1981.

[12]Verhulst, F., *Nonlinear Differential Equations and Dynamical Systems*, Springer–Verlag, New York, 1990.

[13]LaFarge, R. A., and Baty, R. S., "The Functional Dependence of Trajectory Dispersion on Initial Condition Errors," Sandia National Lab., SAND92-0515, Albuquerque, NM, Feb. 1993.