SANDIA REPORT
SAND2000-0824
Unlimited Release
Printed April 2000

Estimation of Total Uncertainty in Modeling and Simulation

William L. Oberkampf, Sharon M. DeLand, Brian M. Rutherford, Kathleen V. Diegert, and Kenneth F. Alvin

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Prices available from (703) 605-6000
Web site: http://www.ntis.gov/ordering.htm

Available to the public from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

SAND2000 - 0824
Unlimited Release
Printed April 2000

Estimation of Total Uncertainty in Modeling and Simulation

William L. Oberkampf
Validation and Uncertainty Estimation Dept.

Sharon M. DeLand
Mission Analysis and Simulation Dept.

Brian M. Rutherford
Statistics and Human Factors Dept.

Kathleen V. Diegert
Reliability Assessment Dept.

Kenneth F. Alvin
Structural Dynamics and Smart Systems Dept

Sandia National Laboratories
P. O. Box 5800
Albuquerque, New Mexico 87185-0828

Abstract

This report develops a general methodology for estimating the total uncertainty in computational simulations that deal with the numerical solution of a system of partial differential equations. A comprehensive, new view of the general phases of modeling and simulation is proposed, consisting of the following phases: conceptual modeling of the physical system, mathematical modeling of the conceptual model, discretization and algorithm selection for the mathematical model, computer programming of the discrete model, numerical solution of the computer program model, and representation of the numerical solution. Our view incorporates the modeling and simulation phases that are recognized in the operations research community, but it adds phases that are specific to the numerical solution of partial differential equations. In each of these phases, general sources of variability, uncertainty, and error are identified. Our general methodology is applicable to any discretization procedure for solving ordinary or partial differential equations. To demonstrate this methodology, we describe two system-level examples: a weapon involved in an aircraft crash-and-burn accident, and an unguided, rocket-boosted, aircraft-launched missile. The weapon in a crash and fire is discussed conceptually, but no computational simulations are performed. The missile flight example is discussed in more detail and computational results are presented.

Acknowledgements

We thank David Salguero of Sandia National Laboratories for providing generous assistance and advice on the computer code TAOS and on missile flight dynamics. Larry Rollstin, also of Sandia, provided the Improved Hawk missile characteristics along with helpful advice on rocket systems. We also thank Jon Helton, Rob Easterling, Tim Trucano, and Vicente Romero of Sandia for their comments and suggestions in reviewing an earlier version of this report.

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the U. S. Department of Energy under Contract DE-AC04-94AL85000.

Contents

Figures

Tables

1. Introduction

Historically, the primary method of evaluating the performance of an engineered system has been to build the design and then test it in the use environment. This testing process is commonly iterative, as design weaknesses and flaws are sequentially discovered and corrected. The number of design-test iterations has been reduced with the advent of computer simulation through numerical solution of the mathematical equations describing the system behavior. Computational results can identify some flaws and they avoid the difficulties, expense, or safety issues involved in conducting certain types of physical tests. Examples include the atmospheric entry of a space probe into another planet, structural failure of a full-scale containment vessel of a nuclear power plant, failure of a bridge during an earthquake, and exposure of a nuclear weapon to certain types of accident environments.

Modeling and simulation are valuable tools in assessing the survivability and vulnerability of complex systems to natural, abnormal, and hostile events. However, there still remains the need to assess the accuracy of simulations by comparing computational predictions with experimental test data through the process known as validation of computational simulations. Physical experimentation, however, is continually increasing in cost and time required to conduct the test. For this reason, modeling and simulation must take increasing responsibility for the safety, performance, and reliability of many high consequence systems.

Realistic modeling and simulation of complex systems must include the nondeterministic features of the system and the environment. By "nondeterministic" we mean that the response of the system is not precisely predictable because of the existence of variability or uncertainty in the system or the environment. Nondeterminism is thoroughly ingrained in the experimental culture, but it is only dealt with in certain modeling and simulation disciplines. Examples of these disciplines are nuclear reactor safety,[1-9] civil and marine engineering,[10-13] and environmental impact.[14-20] The emphasis in these fields has been directed toward representing and propagating parameter uncertainties in mathematical models of the physical event. The vast majority of this work has used probabilistic methods to represent sources of variability or uncertainty and then sampling methods, such as Monte Carlo sampling, to propagate the sources.

Our focus in this report is on a framework for estimating the total modeling and simulation uncertainty in computational predictions. We consider nondeterministic physical behavior originating from a very broad range of variabilities and uncertainties, in addition to inaccuracy due to modeling and simulation errors. Variability is also referred to in the literature as stochastic uncertainty, aleatory uncertainty, inherent uncertainty, and irreducible uncertainty. (For reasons discussed in Section 2, we use the term "variability" and we provide our definition.) The mathematical representation most commonly used for variabilities is a probability or frequency distribution. Propagation of these distributions through a modeling and simulation process has been well developed in the disciplines mentioned above.

Uncertainty as a source of nondeterministic behavior derives from lack of knowledge of the system or the environment. This restrictive use of the term "uncertainty" has been debated and developed during the last decade in the risk assessment community.[2, 15, 16, 19, 21-27] In the literature it is also referred to as epistemic uncertainty and reducible uncertainty. Once you accept

this segregation of variability and uncertainty, one is immediately faced with the question: Are probability (or frequency) distributions appropriate mathematical representations of uncertainty? Since this debate is raging in the literature, we feel compelled to register our opinion in Section 2. Whichever side one chooses in this debate, however, does not affect our proposed framework for modeling and simulation.

The issue of numerical solution error is generally ignored in risk assessment analyses and nondeterministic simulations. Neglecting numerical solution error can be particularly detrimental to total uncertainty estimation when the mathematical models of interest are cast in terms of partial differential equations (PDEs). Types of numerical error that are of concern in the numerical solution of PDEs are: spatial discretization error in finite element and finite difference methods, temporal discretization error in time dependent simulations, and error due to discrete representation of strongly nonlinear features. It is fair to say that the field of numerical error estimation is considered to be completely separate from uncertainty estimation.[28-30] Although many authors in the field of numerical error estimation refer to solution error as "numerical uncertainty," we believe this confuses the issue. Since we concentrate on systems described by the numerical solution of PDEs, we directly include possible sources of error in our framework.

This report proposes a comprehensive, new framework, or structure, of the general phases of modeling and simulation. This structure is composed of six phases, which represent a synthesis of the tasks recognized in the operations research community, the risk assessment community, and the computational mathematics community. The phases are 1) conceptual modeling of the physical system, 2) mathematical modeling of the conceptual model, 3) discretization and algorithm selection for the mathematical model, 4) computer programming of the discrete model, 5) numerical solution of the computer program model, and 6) representation of the numerical solution. Characteristics and activities of each of the phases are discussed as they relate to a variety of disciplines in computational mechanics and thermal sciences. We also discuss the distinction between variability, uncertainty, and error that might occur in any of the phases of modeling and simulation. The distinction between these terms is important not only in assessing how each contributes to an estimate of total modeling and simulation uncertainty, but also how each should be mathematically represented and propagated.

To demonstrate this methodology, we describe two system-level examples: a weapon involved in an aircraft crash-and-burn accident, and a rocket-boosted, aircraft-launched missile. In the weapon in a fire example, we discuss a coupled-physics simulation that addresses the conceptual problem of a weapon damaged in an aircraft crash and exposed to a fuel-fire environment. This simulation considers the widest possible range of a fully coupled thermal-material response analysis regarding the detonation safety of the weapon. The weapon in a crash and fire is discussed conceptually, but no computational simulations are performed. In the missile flight example, we consider the missile to be a relatively short range, i.e., 20 nautical miles, unguided, air-to-ground rocket. In the conceptual modeling phase for this example, we discuss alternative system/environment specifications, scenario abstractions, physics coupling specifications, and nondeterministic specifications. After discussing varying conceptual models, only one branch of the analysis is pursued: rigid body flight dynamics. Of the large number of possible nondeterministic phenomena, we consider only two: variability of the initial mass of the missile and uncertainty in the thrust of the rocket motor because of unknown initial motor temperature. To illustrate mathematical modeling uncertainty, we pursue two models with differing levels of

physics: a six-degree-of-freedom and a three-degree-of-freedom model. In each case we include the effect of error due to numerical solution of the equations of motion for each model.

2. Modeling and Simulation

2.1 Review of the Literature

The operations research (OR) community has developed many of the general principles and procedures for modeling and simulation. Researchers in this field have made significant progress in defining and categorizing the various phases of modeling and simulation. (For recent texts in this field, see Refs. 31-34.) The areas of emphasis in OR include definition of the problem entity, definition of the conceptual model, assessment of data and information quality, validation methodology, and usage of simulation results as an aid in decision making. From a computational sciences perspective, many feel this work is extraneous because it does not deal explicitly with solving PDEs. However, we have found that the OR work is very helpful in providing a constructive philosophical approach for identifying sources of variability, uncertainty, and error, as well as developing some of the basic terminology.

In 1979 the Technical Committee on Model Credibility of the Society for Computer Simulation developed a diagram identifying the primary phases and activities of modeling and simulation.[35] Included as Fig. 1, the diagram shows that analysis is used to construct a conceptual model of reality. Programming converts the conceptual/mathematical model into a computerized model. Then computer simulation is used to simulate reality. Although simple and direct, the diagram clearly captures the relationship of two key phases of modeling and simulation to each other, and to reality. The diagram also includes the activities of model qualification, model verification, and model validation. However, the diagram does not address the detailed activities required for the solution of PDEs describing the system nor the activities necessary for uncertainty estimation.

Jacoby and Kowalik developed a more detailed view for the phases of modeling and simulation in 1980 (Fig. 2).[36] Their view not only better defined the phases of modeling and simulation, they also emphasized the mathematical modeling aspects of the process. After the purpose of the modeling effort is clarified and refined, a prototype modeling effort is conducted. The activities they describe in this effort are similar to those activities the present literature refers to as the conceptual modeling phase. In the preliminary modeling and mathematical modeling phases, various alternate mathematical models are constructed and their feasibility evaluated. In the solution technique phase the numerical methods for solving the mathematical model, or models, are specified. In the computer program phase the actual coding of all the numerical methods is conducted, as well as verification testing of the code. In the model phase, they describe activities that are all related to model validation, i.e., comparisons with experimental data, and checks on the reasonableness of predicted results. In the modeling result phase the interpretation of results is conducted and an attempt is made to satisfy the original purpose of the modeling and simulation effort. The feedback and iterative nature of the entire process is represented by the dashed-loop circling the modeling and simulation effort.

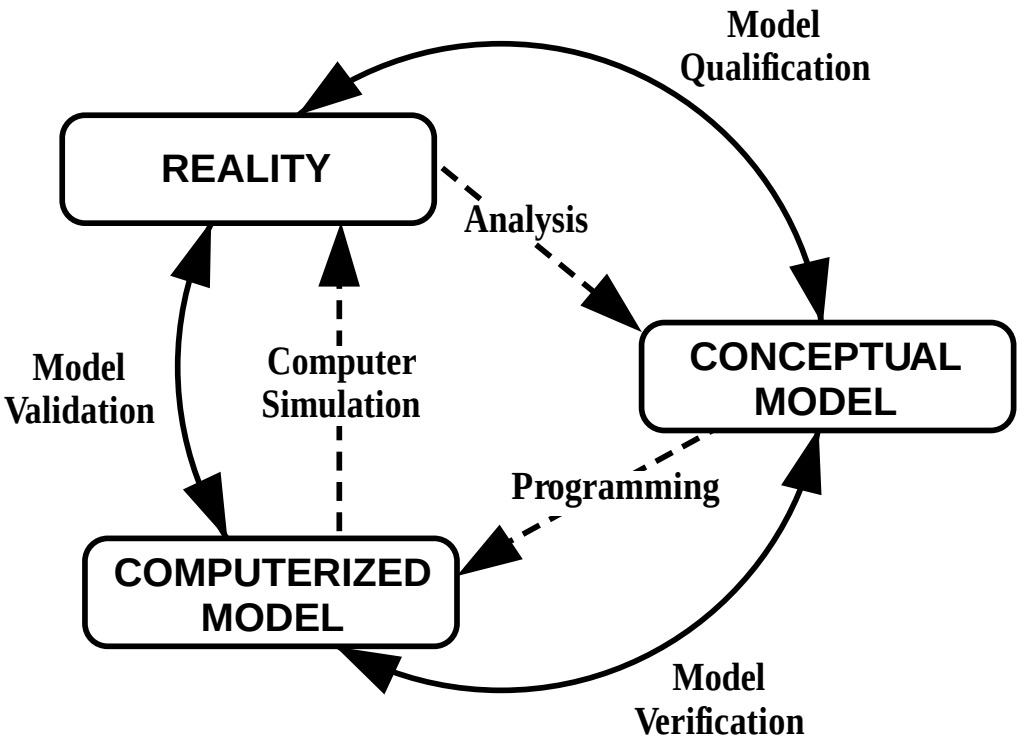

Figure 1
View of Modeling and Simulation by the Society for Computer Simulation[35]

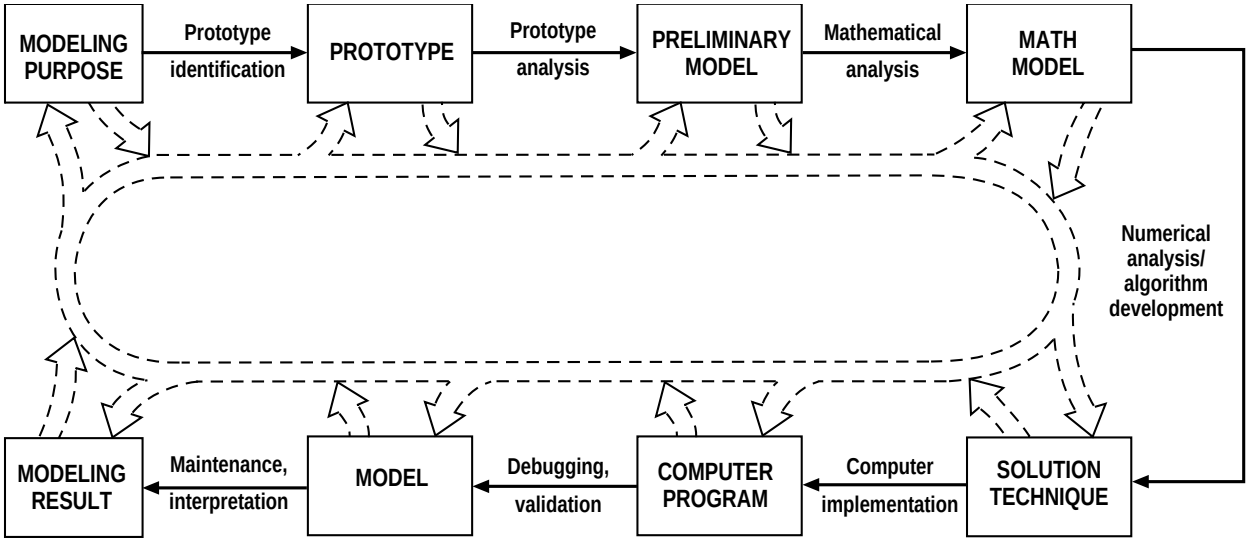

Figure 2
View of Modeling and Simulation by Jacoby and Kowalik[36]

Throughout the 1980s, Sargent[37, 38] made improvements toward generalizing the concepts of modeling and simulation shown in Fig. 1. His most important contribution was the development of general procedures for verification and validation of models and simulations. An extension of the phases of modeling and simulation was made by Nance[39] and Balci[40] to include the concept of the life cycle of a simulation (Fig. 3). Major phases added by Nance and Balci to the earlier description were System and Objectives Definition, Communicative Models, and Simulation Results. Even though the Objectives Definition and Simulation Results phases were specifically identified by Jacoby and Kowalik,[36] there is no indication this work was recognized. Communicative Models were described by Nance and Balci as "a model representation which can be communicated to other humans, can be judged or compared against the system and the study objective by more than one human."[39]

Work in the risk assessment community, specifically, nuclear reactor safety and environmental impact of radionuclides, has not directly addressed the phases of modeling and simulation. They have concentrated on the possible sources that could contribute to total uncertainty in risk assessment predictions. Reactor safety analyses have developed extensive methods for constructing possible failure and event tree scenarios that aid in risk assessment.[2, 5, 7, 9, 19, 41] Analyses of the risk of geologic repositories for the disposal of low-level and high-level nuclear waste have used scenario analyses, and they have identified sources of indeterminacy and inaccuracy occurring in other phases of the risk analysis. Specifically, they have identified different types of sources occurring in conceptual modeling, mathematical modeling, computer code implementation, and experimentally measured or derived model input data.[42, 43]

The development of the present framework for the phases of modeling and simulation builds on much of this previous work. Some of this work, however, we were not aware of until very recently. Our framework could be viewed as a synthesis of this reviewed literature, and the addition of two elements. First, a more formal treatment of the nondeterministic elements of the system and its environment, and second, a dominant element incorporating the numerical solution of partial differential equations. Our unification of these perspectives will be presented and discussed in Section 2.3.

2.2 Sources of Variability, Uncertainty, and Error

Sources of variability, uncertainty, and error are associated with each phase of modeling and simulation. Examining the literature in many fields that deal with nondeterministic systems (e.g., operations research, structural dynamics, and solid mechanics) one finds that most authors do not carefully distinguish between what they mean by variability, uncertainty, and error, or worse, their definitions contradict one another. Even when these terms have been defined, their definitions are typically couched in the restricted context of the particular subject, or they do not address the issue of error.[2, 7, 9, 10, 41] During the last ten years some authors in the risk assessment field have begun to clearly distinguish between some of these sources; particularly the distinction between variability and uncertainty.[2, 15, 16, 21-27, 44-54] This field is the first to use the separate notion and treatment of variability (aleatory uncertainty) and uncertainty (epistemic uncertainty) in practical applications. We are convinced of the constructive value of this distinction and we will adopt

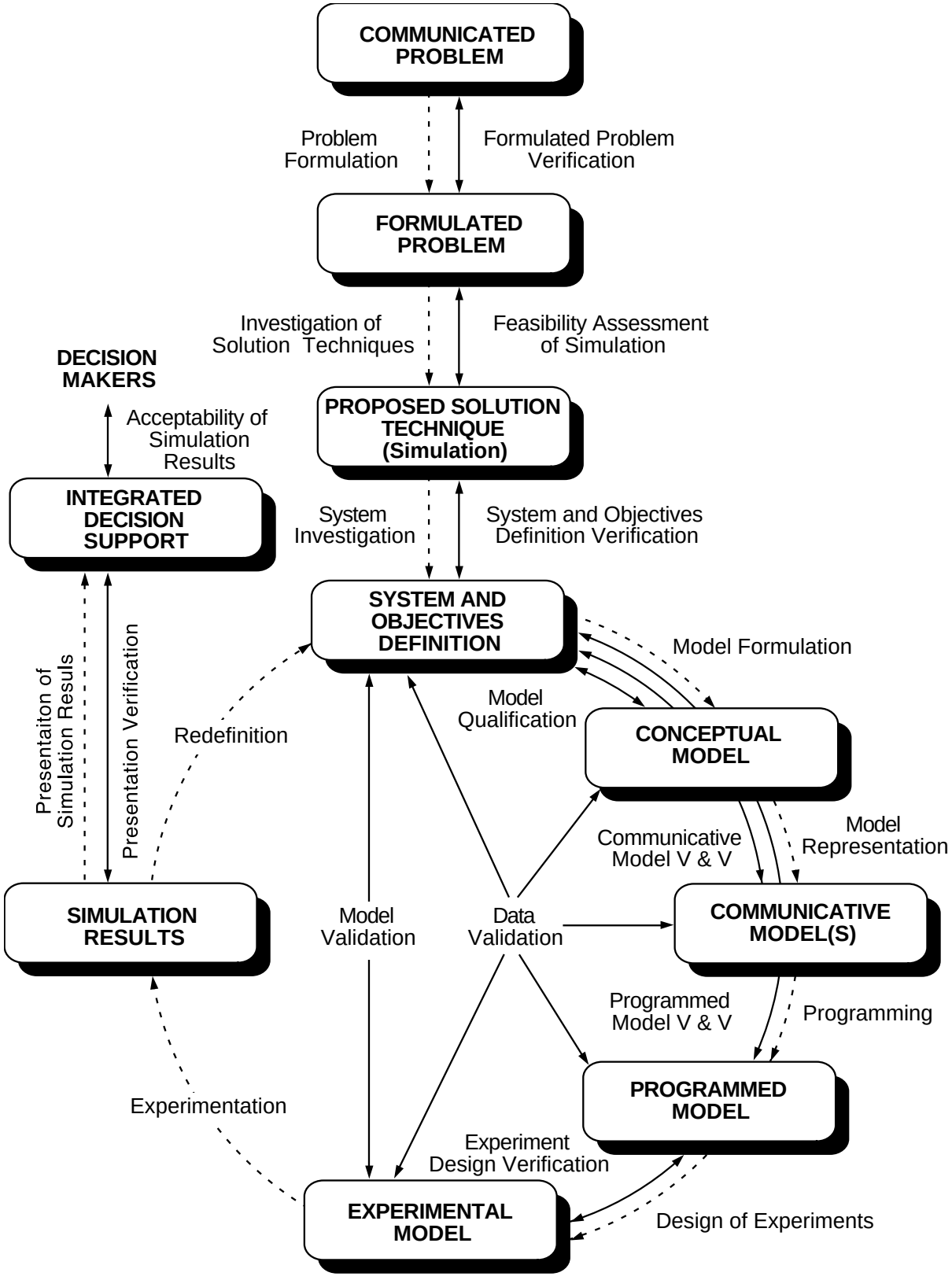

Figure 3
Life Cycle of a Simulation Study[39, 40]

essentially the same definitions used by these authors. Recommended texts which emphasize the mathematical representation aspects of variability and uncertainty are Refs. 55-62.

We use the term *variability* to describe the *inherent* variation associated with the physical system or the environment under consideration. Sources of variability can commonly be singled out from other contributors to total modeling and simulation uncertainty by their representation as distributed quantities that can take on values in an established or known range, but for which the exact value will vary by chance from unit to unit or from time to time. As mentioned earlier, variability is also referred to in the literature as stochastic uncertainty, aleatory uncertainty, inherent uncertainty, and irreducible uncertainty. An example of a distributed quantity is the exact dimension of a manufactured part, where the manufacturing process is well understood but variable and the part has yet to be produced. Variability is generally quantified by a probability or frequency distribution when sufficient information is available to estimate the distribution.

We define *uncertainty* as a *potential* deficiency in any phase or activity of the modeling process that is due to *lack of knowledge*. The first feature that our definition stresses is "potential," meaning that the deficiency may or may not exist. In other words, there may be no deficiency, say in the prediction of some event, even though there is a lack of knowledge if we happen to model the phenomena correctly. The second key feature of uncertainty is that its fundamental cause is incomplete information. Incomplete information can be caused by vagueness, nonspecificity, or dissonance.[55, 63] Vagueness characterizes information that is imprecisely defined, unclear, or indistinct. Vagueness is characteristic of communication by language. Nonspecificity refers to the variety of alternatives in a given situation that are all possible, i.e., not specified. The larger the number of possibilities, the larger the degree of nonspecificity. Dissonance refers to the existence of totally or partially conflicting evidence. Dissonance exists when there is evidence that an entity or elements belong to multiple sets that either do not overlap or overlap slightly. Mathematical theories available for representation of uncertainty are, for example, evidence (Dempster/Shafer) theory,[58, 64] possibility theory,[65, 66] fuzzy set theory,[55, 62] and imprecise probability theory.[57, 67]

Since the cause of uncertainty is partial knowledge, increasing the knowledge base can reduce the uncertainty. As mentioned earlier, in the literature our definition of uncertainty is also referred to as epistemic uncertainty and reducible uncertainty. When uncertainty is reduced by an action, such as observing, performing an experiment, or receiving a message, that action is a source of information. The amount of information obtained by the action is measured by the resulting reduction in uncertainty. This concept of information is called "uncertainty-based information." Examples of this concept include: improving the accuracy of prediction of heat flux in a steel bar by learning more about the thermal conductivity of the bar in the predictive model, improving the prediction of the convective heat-transfer rate in turbulent flow by refining the turbulence model, and improving the prediction accuracy for melting a component in a fuel fire by gaining more knowledge of the typical atmospheric wind conditions.

We define *error* as a *recognizable* deficiency in any phase or activity of modeling and simulation that is *not* due to lack of knowledge. Our definition stresses the feature that the deficiency is identifiable or knowable upon examination; that is, the deficiency is not caused by

lack of knowledge. Essentially there is an agreed-upon approach or ideal condition that is considered to be more accurate. If divergence from the correct or more accurate approach is pointed out, the divergence is either corrected or allowed to remain. It may be allowed to remain because of practical constraints, such as the error is acceptable given the requirements, or the cost to correct it is excessive. This implies a segregation of error types: an error can be either *acknowledged* or *unacknowledged*. Acknowledged errors are those deficiencies that *are* recognized by the analysts. When acknowledged errors are introduced by the analyst into the modeling or simulation process, the analyst typically has some idea of the magnitude or impact of such errors. Examples of acknowledged errors are finite precision arithmetic in a computer, approximations made to simplify the modeling of a physical process, and conversion of PDEs into discrete equations. Unacknowledged errors are those deficiencies that are *not* recognized by the analyst, but they are recognizable. Examples of unacknowledged errors are blunders or mistakes, that is, the analyst intended to do one thing in the modeling and simulation but, for example, as a result of human error, did another. There are no straightforward methods for estimating, bounding, or ordering the contribution of unacknowledged errors. Sometimes an unacknowledged error can be detected by the person who committed it; e.g., a double-check of coding reveals that two digits have been reversed. Sometimes blunders are caused by inadequate human interactions and can only be resolved by better communication. Redundant procedures and protocols for operations depending on a high degree of human intervention can also be effective in reducing unacknowledged errors.

Our definitions of uncertainty and error may seem strange, or even inappropriate, to those familiar with experimental measurements, or the science of physical measurements: metrology. In experimental measurements, error is defined as "the difference between the measured value and the true value."[68] Experimentalists define uncertainty as "the estimate of error."[68] We do not believe these definitions are sufficient for modeling and simulation for two reasons. First, the experimentalists definition of error depends on two factors; the measured value and the true value. The measured value is well defined and perfectly clear. The true value is *not* known, except in the special case of comparison with a defined standard, that is, an accepted true value. For the general case then, the true value and the error are not known and they can only be subjectively estimated.[2] Our definitions of error and uncertainty precisely segregate the meaning of the two terms with knowledge, i.e., what is "known" (or can be ordered) and what is "unknown." Second, by defining uncertainty as an estimate of error the experimentalists are saying that, from the view of knowledge theory, uncertainty and error are the same type entity. For example, if uncertainty were to be zero, then either the error is zero, or the uncertainty is erroneous.

From our definitions variability and uncertainty are somewhat related, but error clearly has different characteristics. Variability and uncertainty are normally thought to produce stochastic, or non-deterministic, effects, whereas errors commonly yield a reproducible, or deterministic, bias in the simulation. In some applications we expect that there will be sources that do not fall precisely into either the variability category or the uncertainty category. Consider, for example, a newly designed solid fuel gas generator that closely resembles previous designs and manufacturing processes. Assume that very limited test results are available on the performance of this new design. If modeling and simulation are used to predict the performance of the new gas generator, the total predicted uncertainty will contain inherent variability similar to that associated with previous designs, but it will also contain an uncertainty component based on the lack of knowledge related to the effect of the design changes.

2.3 Proposed Phases of Modeling and Simulation

Figure 4 depicts our representation of the phases of modeling and simulation appropriate to systems analyzed by the numerical solution of PDEs. The phases represent collections of tasks required in a large scale simulation analysis. The ordering of the phases implies an information and data flow indicating which tasks are likely to impact decisions and methodology occurring in later phases. However, there is significant feedback and interaction between the phases, as is shown in Fig. 4. These phases follow the recent work of Refs. 69, 70. The paragraphs below provide brief descriptions of each of these phases. The modeling and simulation process is initiated by a set of questions, posed by a designer or decision maker, for which the information to address the questions can be provided (at least in part) through a computer simulation analysis.

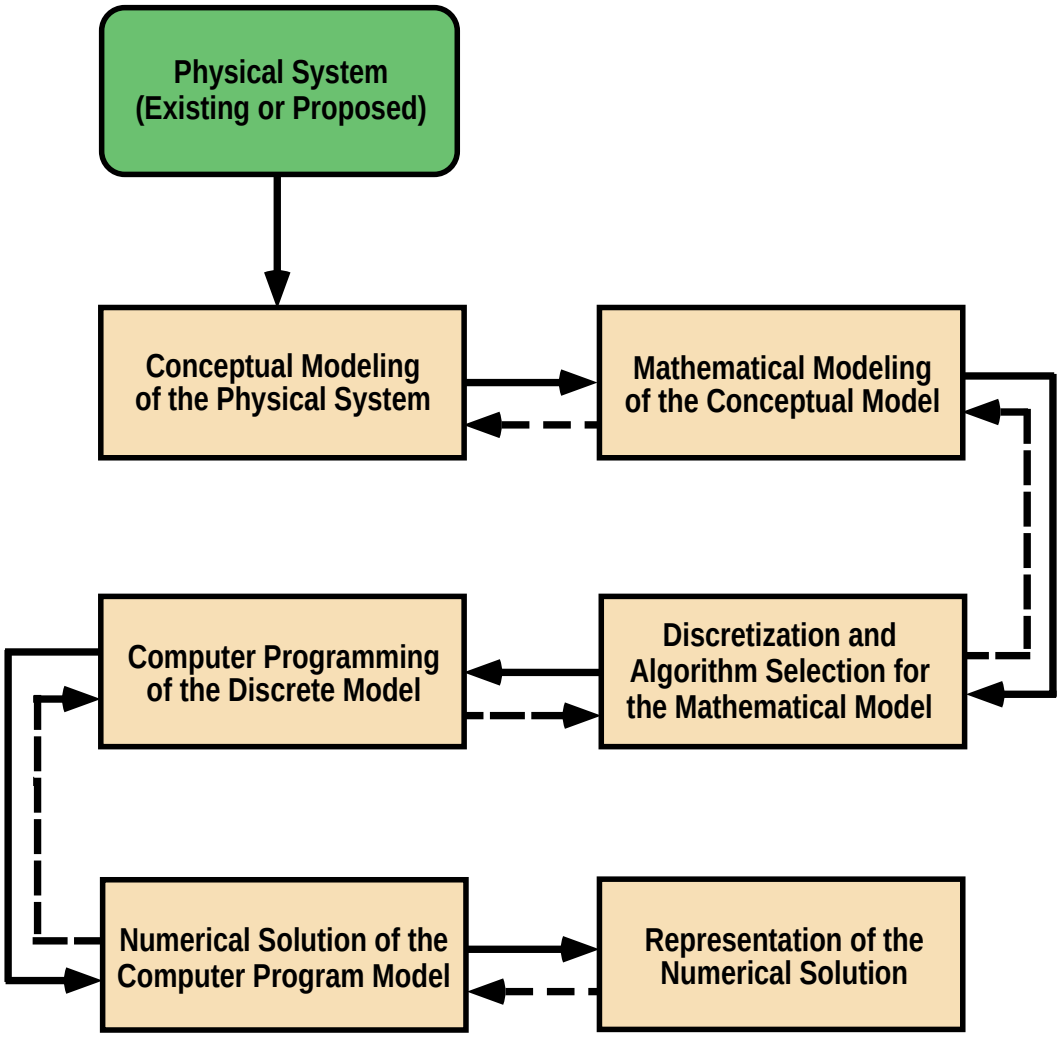

Figure 4
Proposed Phases for Computational Modeling and Simulation

Conceptual Modeling of the Physical System. Our initial phase encompasses developing a

specification of the physical system and the environment. This includes determining which physical events, or sequence of events, and which types of coupling of different physical processes will be considered. It also includes identifying elements of the system and environment that will be treated as nondeterministic. These determinations must be based on the general requirements for the modeling and simulation effort. The physical system can be an existing system or process, or it can be a system or process that is being proposed. During the conceptual modeling phase, no mathematical equations are written, but the fundamental assumptions of the possible events and physical processes are made. Only conceptual issues are considered, with heavy emphasis placed on determining all possible factors, such as physical and human intervention, that could possibly affect the requirements set for the modeling and simulation. Identifying possible event sequences, or scenarios, is similar to developing a fault-tree structure in the probabilistic risk assessment of high consequence systems, such as in nuclear reactor safety analyses. Even if a certain sequence of events is considered extremely remote, it should still be included as a possible event sequence in the fault tree. Whether or not the event sequence will eventually be analyzed is not a factor that impacts its inclusion in the conceptual modeling phase. After the system and environment are specified, options for various levels of possible physics couplings should be identified, even if it is considered unlikely that all such couplings will be considered subsequently in the analysis. If a physics coupling is not considered in this phase, it cannot be resurrected later in the process. Another task conducted in this phase of the analysis is the identification of all of the system and environment characteristics that might be treated nondeterministically. Consideration is given as to whether these characteristics are to be treated as fixed, stochastic, or unknown. However, details concerning their representation and propagation are deferred until later phases.

Mathematical Modeling of the Conceptual Model. The primary task in this phase is to develop precise mathematical models, i.e., analytical, statements of the problem (or series of event-tree-driven problems) to be solved. Any complex mathematical model of a problem, or physical system, is actually composed of many mathematical submodels. The complexity of the models depends on the physical complexity of each phenomenon being considered, the number of physical phenomena considered, and the level of coupling of different types of physics. The mathematical models formulated in this phase include the complete specification of all PDEs, auxiliary conditions, boundary conditions, and initial conditions for the system. For example, if the problem being addressed is a fluid-structure interaction, then all of the coupled fluid-structures PDEs must be specified, along with any fluid or material-property changes that might occur as a result of their interaction. The integral form of the equations could also be considered, but this type of formulation is not addressed in the present discussion.

Another function addressed during this phase of analysis is selecting appropriate representations and models for the nondeterministic elements of the problem. Several considerations might drive these selections. Restrictions set forth in the conceptual modeling phase of the analyses may put constraints on the range of values or types of models that might be used further in the analysis. Within these constraints the quantity and/or limitations of available or obtainable data will play an important role. A probabilistic treatment of nondeterministic variables generally requires that probability distributions can be established, either through data analysis or through subjective judgments. In the absence of data, qualified “expert opinion” or similar type information from other sources regarding the relative likelihoods may be incorporated. If there is a significant lack of information, it is possible that only bounding or set representations may be

appropriate for uncertainties.

Our emphasis on comprehensiveness in the mathematical model should not be confused with a model's attempt to represent physical complexity. The predictive power of a model depends on its ability to correctly identify the dominant controlling factors and their influences, *not* upon its completeness. A model of limited, but known, applicability is often more useful than a more complete model. This dictum of engineering seems to be forgotten today with the advent of rapidly increasing computing power. The clear tendency, observable in all fields of engineering, is to use more complex models and then "beat them to death" with the computer. Some of the more flagrant examples of this tendency are found in the use of Navier-Stokes equations to compute the lift on a streamlined body at low angle of attack, the use of time-iterative Navier-Stokes equations to compute attached supersonic flow over a vehicle, and the use of finite elements through the thickness of a thin shell (rather than shell elements) to compute the stress.

An additional point concerning the incompleteness of models should be made. Any mathematical model, regardless of its physical level of detail, is *by definition* a simplification of reality. Any complex engineering system, or even individual physical processes, contain phenomena that are not represented in the model. Statements such as "full physics simulations" can only be considered as marketing jargon. Our point was succinctly stated nearly twenty years ago by Box[71]: "All models are wrong, some are useful."

Discretization and Algorithm Selection for the Mathematical Model. This phase accomplishes two tasks related to converting the mathematical models into a form that can be addressed through computational analysis. The first task involves conversion of the PDE form of the mathematical model into a discrete, or numerical, model. Simply stated, the mathematics are translated from a calculus problem to an arithmetic problem. In the discretization phase all of the spatial and temporal differencing methods, discretized boundary conditions, discretized geometric boundaries, and grid generation methods are specified in analytical form. In other words, algorithms and methods are prescribed in mathematically discrete form, but the spatial and temporal step sizes are not specified. The discretization phase focuses on the conversion from continuum mathematics to discrete mathematics, not on numerical solution issues. We strongly believe that the continuum mathematical model and the discrete model should be separately represented in the phases of modeling and simulation.[72] The discretization phase deals with questions such as consistency of the discrete equations with the PDEs, stability of the numerical method, approximations of mathematical singularities, and differences in zones of influence between the continuum and discrete systems.

The second task of this phase of the analysis is the specification of the methodology that will dictate computer runs to be performed in a later phase of the analysis in order to accommodate the nondeterministic aspects of the problem. For example, a Monte Carlo method or response surface method could be chosen for propagating variabilities. Nondeterministic sources include system parameters, boundary conditions, and initial conditions that may vary randomly from component-to-component and/or system-to-system. Modeling too can be nondeterministic in nature when alternative models are constructed to address the same aspects of the problem. Presumably only one model is correct (or more correct) for the task, but this is not generally known beforehand, i.e., in a prediction. In addition, a preferred model may be too expensive to be used exclusively in the analysis and, as a result, less accurate models would be used for portions of the analysis.

Computer Programming of the Discrete Model. This phase is common to all computer modeling: algorithms and solution procedures defined in the previous phase are converted into a computer code. The computer programming phase has probably achieved the highest level of maturity because of decades of programming development and software quality assurance efforts.[73, 74] These efforts have made a significant impact in areas such as commercial graphics, mathematics, and accounting software, telephone circuit-switching software, and flight control systems. On the other hand, these efforts have had little impact on corporate and university-developed software for computational fluid dynamics, solid dynamics, and heat transfer simulations, as well as most applications written for massively parallel computers.

Numerical Solution of the Computer Program Model. In this phase the individual numerical solutions are actually computed. No quantities are left arithmetically undefined or continuous; only discrete parameters and discrete solutions exist with finite precision. For example, a spatial grid distribution and a time step is specified; space and time exist only at discrete points, although these points may be altered during subsequent computer runs.

Multiple computational solutions are usually required for nondeterministic analyses. These multiple solutions are dictated by the propagation methods and input settings determined in the discretization and algorithm selection phase. Multiple solutions can also be required from the mathematical modeling phase if alternative models are to be investigated. For some propagation methods the number and complete specification of subsequent runs is dependent on the computed results. When this is the case, these determinations are made as part of this phase of the analysis.

Representation of the Numerical Solution. The final phase of the modeling and simulation process concerns the representation and interpretation of both the individual and collective computational solutions. The collective results are ultimately used by decision makers or policy makers, whereas the individual results are typically used by engineers, physicists, and numerical analysts. Each of these audiences have very different interests and requirements. The individual solutions provide detailed information on deterministic issues such as the physics occurring in the system, the adequacy of the numerical methods to compute an accurate solution to the PDEs, and the system's response to the deterministic boundary and initial conditions. For the individual solutions the primary task is the construction of continuous functions based on the discrete solutions obtained in the previous phase. Here the continuum mathematics formulated in the mathematical modeling phase is approximately reconstructed based on the discrete solution.

We have specifically included representation of the numerical solution as a phase in the modeling and simulation process because of the sophisticated software that is being developed to comprehend modern complex simulations. This area includes three-dimensional graphical visualization of solution, animation of solution, use of sound for improved interpretation, and use of virtual reality which allows analysts to "go into the solution space." Some may argue that this final phase is simply "post-processing" of the computational data. We believe, however, this description does not do justice to the rapidly growing importance of this area and the possibility that it introduces unique types of errors. In addition, by referring to this phase as representation of the numerical solution, we are able to include types of errors that are not simply due to the modeling and simulation of the system, but also to the processing of the computed solution and to the conclusions drawn therefrom.

The collective solutions provide information on the nondeterministic response of the system. For the collective solutions the primary task is the assimilation of individual results to produce summary data, statistics, and graphics portraying the nondeterministic features of the system. These results are utilized to assess the simulation results from a high-level perspective and compare them to requirements of the analysis.

Summary. The phases of modeling and simulation described above illustrate the major components involved in planning and conducting a large-scale simulation analysis. When viewed from the planning aspect, the issues confronted in each phase may be addressed simultaneously. For example, in most large-scale system simulations the activities will be performed by different groups of people with different areas of expertise, such as professional planners, physicists, engineers, computer programmers, and numerical analysts. A “feedback” aspect indicated in Fig. 4, but not explicitly discussed here, is the use of sensitivity analyses in a large-scale analysis. Sensitivity analyses and scoping studies are critical when there are hundreds of variabilities and uncertainties in an analysis. Sensitivity analyses and scoping studies are clear examples of how feedback from the solution representation phase occurs in a large-scale analysis. There is, however, a clear sequential aspect to the phases as shown Fig. 4. Two key sequential features of this illustration are that decisions must be made at each phase and that continuous parameters and model specification information propagate through the phases. In most cases, the decisions made at one phase will impact the models formulated or activities conducted in later phases. A single simulation run will be characterized by assumptions "assigned" from choices set forth in each of the phases. When simulations are actually performed and the simulation results are analyzed, total modeling and simulation uncertainty can be attributed to the various assumptions and inputs and ultimately (where it is of interest to do so) to the phases themselves.

3. Weapon in a Fire Example

3.1 Description of the Problem

This example problem, and the one given in Section 4, are used to expand upon the activities conducted in the phases of modeling and simulation and provide different examples for sources of variability, uncertainty, and error that occur in each of the phases. We consider the coupled thermal-material analysis of a weapon in an open-pool fuel-fire environment. Assume that the weapon may be damaged, but that the level of damage is unknown. This example would be characteristic of a weapon carried by an aircraft that crashed during take-off or landing. Assume that the type of weapon is known, but no other information about the weapon before the accident is known. The weapon contains high explosive that is normally a solid, and the weapon has an integrated electrical-mechanical arming, fusing, and firing system. For this example problem, the purpose of the analysis is to compute a probabilistic estimate of whether the high explosive will detonate. Stated somewhat differently, the goal of the analysis is to compute a risk assessment of the detonation safety of the weapon in a crash-and-burn scenario.

The purpose of our example problem is to point out the myriad of factors and possibilities that enter into a complex, real-world, engineering simulation. In our example problem we only discuss variabilities, uncertainties, and errors in the conceptual modeling and mathematical modeling phases. These are the only phases discussed in this example because they determine the scope and

complexity of the analysis. Although no computations are made here, the potential magnitude of the required computing effort should become clear.

3.2 Conceptual Modeling Activities

Figure 5 illustrates the activities conducted in each of our phases of modeling and simulation. In Fig. 5 we identify four activities for the initial phase of the modeling and simulation process: system/environment specification, scenario abstraction, coupled physics specification, and nondeterministic specification. Although nondeterministic specification could be considered as a subset of system/environment specification, we have separated these two activities to place emphasis on nondeterministic solutions in modeling and simulation.

The system/environment specification activity involves the careful delineation between what is considered part of the system and what is considered part of the environment. Elements of the system can be influenced by the environment, but the system cannot influence the environment. It is obvious that multiple system/environment specifications can be employed, depending on the requirements of the modeling and simulation effort. The system/environment specification activity primarily introduces uncertainties that arise in defining the physical modeling scope of the problem. The wider the scope, the more possibilities there are for uncertainties due to lack of knowledge about aspects of the modeled system and environment. Note that errors can also arise in the activity of defining the physical system, but these are less of a concern.

Scenario abstraction consists of the determination of all possible physical events, or sequences of events, that may affect the goals of the analysis. According to our definitions given in Section 2, primarily uncertainties will populate this activity. For relatively simple systems, such as fluid flow with no interaction with any structures or materials, scenario abstraction can be straightforward. For complex engineered systems exposed to a variety of interacting factors, scenario abstraction is a mammoth undertaking. The best example we can give for how this should be accomplished for complex systems is the probabilistic safety assessment of nuclear power plants. As the many-branched event tree is constructed for complex scenarios, the probability of occurrence of certain events becomes extremely low. Typically little analysis effort is expended on these extraordinarily rare possibilities. If one is dealing with very high consequence systems, however, these extremely improbable scenarios must be examined. Guidance concerning whether these events should be included is usually determined by conceptually estimating the risk, i.e., the product of the expected frequency of occurrence and the magnitude of the consequence of the event.

Coupled physics specification consists of identifying and clarifying what physical and chemical processes could be considered in the modeling and also what level of coupling could be considered between them. Computational analysts tend to immediately focus on the “practical” or “affordable” levels of coupled physics analyses. This is an efficient approach in many instances in the sense that little time is spent with higher levels of coupling that may not be allowable within the scope (schedule and budget) of the analysis. However, the danger of this approach is to eliminate from possible consideration those analyses that may be required for assessment of certain types of indeterminacy or risk. By identifying alternate levels of coupling, acknowledged errors are clearly the source that will result in inaccuracy in predictions from the analysis. The ordering of the accuracy, or physical fidelity, of the alternate models can be difficult or impossible for complex

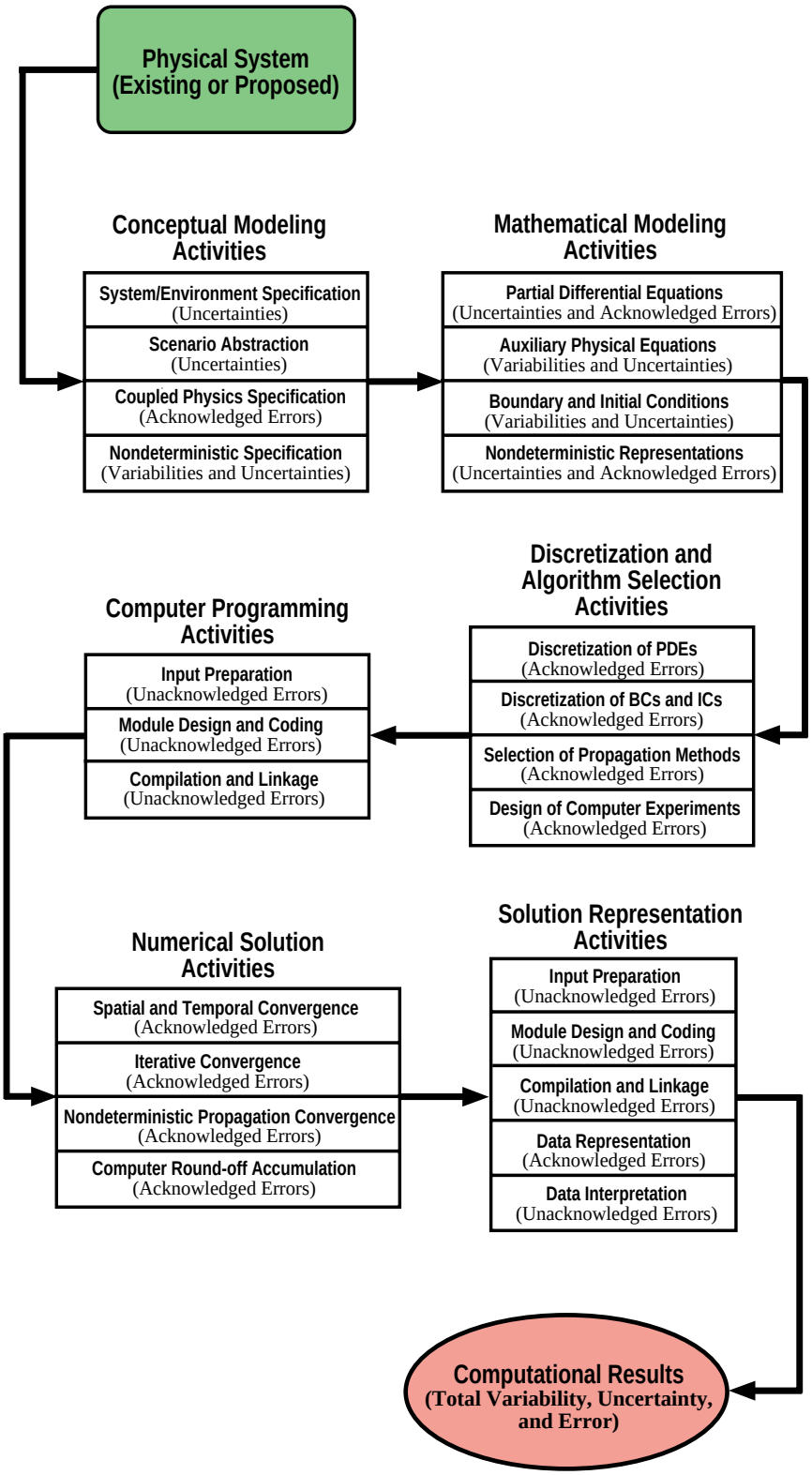

Figure 5
Activities Conducted in the Phases of Computational Modeling and Simulation

multiphysics/chemistry systems.

In the nondeterministic specification activity, decisions are made concerning what aspects of the system and environment will be considered deterministic or nondeterministic. Variabilities and uncertainties are the dominant sources in the nondeterministic specification activity. Variabilities arise because of inherent randomness in parameters or conditions of the process or the event. For a complex engineered system, uncertainties occur because of lack of knowledge about initial factors, such as the following, that might have impacted the system: Was the system incorrectly manufactured or assembled? How well was the system maintained? Was the system damaged in the past and not recorded? These are examples where it may not be possible to reduce the lack of knowledge, and reduce the uncertainty, by improved sampling of past events. However, the uncertainty can sometimes be reduced by certain actions taken with respect to the system that limits or further defines the state of key elements of the system. Often these are policy or procedural decisions.

For our example problem of a weapon in a fire, the list below provides a number of possible sources of variabilities, uncertainties, and errors applicable to the particular activities in the conceptual modeling phase. Rather than attempt to list all of the possibilities, we give examples of possible choices and sources of variabilities, uncertainties, and errors that could be included in each of the activities:

<u>System/Environment Specification</u>
- Specification of aircraft as part of the system; everything else is part of the environment
- Inclusion of aircraft and fire as part of the system; everything else is part of the environment
- Inclusion of aircraft, fire, and emergency response activities as part of the system; everything else is part of the environment
- Specification of what elements of physics, chemistry, and electronics are to be included in the simulation

<u>Scenario Abstraction</u>
- Consideration of structural and electrical damage to the arming, fusing and firing system before the start of the fire
- Uncertainty in structural damage due to the crash before the start of the fire
- Number of similar weapons and other weapons carried on-board the aircraft
- Effect of an adjacent weapon detonating during the crash and/or fire
- Uncertainty in aircraft crash area characteristics, e.g., water, trees, city
- Possible atmospheric source of electrical energy to the arming, fusing and firing system during the crash and fire, e.g., lightning
- Possible effects of emergency response teams to the crash and fire, e.g., use of water to extinguish the fire

<u>Coupled Physics Specifications</u>
- Consideration of fully coupled thermal/material/electrical system of one weapon, fuel pool, aircraft, local structures and ground material, and local atmosphere
- Consideration of fully coupled thermal/material response of one weapon
- Consideration of fully coupled thermal/material response of two weapons, with detonation of one weapon

Nondeterministic Specifications

- Manufacturing and assembly variability of components and the complete system
- Variability in material properties of components and subsystems before the crash
- Lack of information concerning maintenance, storage history, and possible damage of the weapon before the accident
- Uncertainty in aircraft crash fuel sources and quantity
- Uncertainty in wind speed and temperature and other meteorological conditions during the fire
- Uncertainty in thermal emissivity of surfaces before and during the fire

3.3 Mathematical Modeling Activities

As shown in Fig. 5, we have identified four general activities in the mathematical modeling phase: 1) formulation of all the continuum equations for conservation equations of mass, momentum, and energy; 2) formulation of all the auxiliary equations that supplement the conservation equations, such as expressions for thermal conductivity, fluid dynamic turbulence models, and chemical reaction equations; 3) formulation of all the initial and boundary conditions required to solve the PDEs, and 4) the selection of a mathematical representation for the nondeterministic elements of the system. In this phase, all three contributors are possible sources: variabilities, uncertainties, and errors. (Note that for the remainder of the paper when we refer to "errors" we will only be referring to *acknowledged* errors, unless otherwise stated.) The most common variabilities and uncertainties are, respectively, those due to inherent randomness of parameters in known physics, and those due to limited, or inadequate, knowledge of the physics involved. Note that parameter variability is by far the most commonly analyzed in nondeterministic analyses. The most common errors introduced are those due to mathematically representing the physics in a more simplified or approximate form. Together, the mathematical modeling uncertainties and errors are sometimes referred to as "model form errors" or "model structural errors" in the literature.

Examples of uncertainties that occur in the formulation of the conservation equations are limited knowledge of the physics of multiphase flow, limited knowledge of turbulent reacting flow, and uncertainty in the modeling of fluid/structure/chemical interactions. Examples of variabilities and uncertainties that occur in formulation of the auxiliary physical equations are, respectively, poorly known material properties resulting from manufacturing variability, and unreliable fluid flow turbulence models. It may be argued that deficiencies in turbulence models should be considered as errors rather than uncertainties. This is based on the argument that the accuracy of turbulence models could be ordered, e.g., algebraic models, one-equation models, and two-equation models. In a general sense, this ordering could be accepted; but for individual flow fields there is no guarantee that any one model will be better than any other model. Examples of variabilities in initial and boundary conditions are dimensional tolerances in component geometry due to manufacturing and assembly variances, initial temperature distribution in a solid, and turbulence levels in the approaching wind. Examples of uncertainties in initial and boundary conditions are unknown damaged weapon and component geometry, unknown damaged aircraft geometry, and unknown location of the weapon relative to the fire, aircraft, and wind direction.

Acknowledged errors in the nondeterministic representation are often the result of inferring a

model (probabilistic or otherwise) that is based on incomplete understanding of the variable or phenomenon. Only in those rare cases where the models can be derived from first principles, given the assumptions of the model, are the representations known and exact. Uncertainties are introduced through this activity in the specification of, or fitting of, parameters associated with these representations. For example, if the dimension of a manufactured component is represented through a normal probability distribution and there are limited production data on the dimension, then there is uncertainty associated with the mean and variance estimates for that distribution.

Some examples of acknowledged errors in mathematical modeling are the assumption that a flow field can be modeled as a two-dimensional flow when three-dimensional effects may be important, the assumption of a steady flow when the flow is actually unsteady, the assumption of continuum fluid mechanics when noncontinuum effects may be a factor, and the assumption of a rigid boundary when the boundary is flexible. These examples of acknowledged errors are all characteristic of situations in which physical modeling approximations were made to simplify the mathematical model and the subsequent solution.

For the sample problem of a weapon in a fire, the list below provides examples of mathematical modeling choices and sources of variabilities, uncertainties, and errors for each of the mathematical modeling activities:

Conservation Equations
- Use of unsteady two-dimensional analysis, or unsteady three-dimensional analysis
- Statement of the fluid, structural, and electrical circuit conservation equations
- Statement of the detailed coupling of the fluid, structural, and electrical-circuit conservation equations

Auxiliary Physical Equations
- Use of algebraic, one-equation, two-equation, or large-eddy-simulation turbulence models
- Specification of reacting flow materials; aircraft fuel, weapon components, aircraft structure and subsystems, crash site surroundings, etc.
- Use of equilibrium or nonequilibrium chemical reaction models
- Specification of the number of gas species used in chemical reaction models
- Specification of which materials will be considered to change phase (melting, solidification, vaporization, and condensation)
- Variability or uncertainty in thermodynamic and transport properties of all materials
- Uncertainty due to use of transport and thermodynamic properties outside their range of validity
- Inappropriate or inaccurate statistical models to represent variability in continuous parameters

Nondeterministic Representations
- The assignment of a normal distribution to characteristics of various components of the system based on considerations in their manufacturing process
- Assigning a uniform distribution to the aircraft fuel quantity
- Using a set of bounding values to represent meteorological considerations during the fire
- Making a set of assumptions concerning the maintenance and storage history of the weapon prior to the accident but not conveying to users that the analysis was conditional on these assumptions

Boundary and Initial Conditions
- Uncertainty in damaged weapon geometry before the fire
- Uncertainty in damaged aircraft and surroundings geometry before and during the fire
- Variability of thermal contact resistance in all solid-solid interfaces before and during the fire
- Change in geometry of systems and components due to melting and vaporization
- Variability and uncertainty of wind and temperature conditions near the crash site

3.4 Discretization and Algorithm Selection Activities

The discretization and algorithm selection phase consists of determining the approaches to be used for converting the continuum model of the physics into a discrete mathematics problem and converting the continuous representation of the nondeterministic elements to a discrete set of analyses. Converting the continuum model is fundamentally a mathematics-approximations topic, errors and not uncertainties are the dominant issue in this phase. Some may question why this conversion process should be separated from the solution process. We argue that this conversion process is the root cause of more difficulties in the numerical solution of nonlinear PDEs than is generally realized. Taking a historical perspective, early numerical methods and solutions were developed for linear PDEs, such as simple heat conduction, Stokes flow, and linear structural dynamics. Modern numerical solutions have attacked nonlinearities such as high-Reynolds-number laminar flow and shock waves and, in hindsight, these have proven more difficult than anticipated. Additional nonlinear physics such as turbulent flow, combustion, detonation, multiphase flow, phase changes of gases, liquids and solids, fracture dynamics, and chaotic phenomena are also being attacked with limited success. When strongly nonlinear features are coupled, the mathematical underpinnings become very thin and the successes become fewer. Recent investigators[75] have clearly shown that the numerical solution of nonlinear ordinary and PDEs can be quite different from exact analytical solutions even when using established methods that are well within the numerical stability limits. This phenomena has been referred to as the "dynamics of numerics" as opposed to the "numerics of dynamics."[76] It is becoming increasingly clear that the mathematical features of strongly nonlinear and chaotic systems can be fundamentally different between the continuous and discrete form, regardless of the grid size.[77] It has been pointed out that the zones of influence between the continuum and numerical counterparts are commonly different, even in the limit as the mesh size approaches zero.[72]

Determining an appropriate approach to selecting representative nondeterministic elements and values and then implementing this approach is a second set of activities associated with this phase. The nondeterministic elements of the system generally take on values over a continuous range. System responses and performance criterion, however, are generally calculated and analyzed for a discrete set of sub-problems where values for these nondeterministic elements are completely specified. Inferences are then based on the discrete results, hence it is important that these sub-problems be selected to be representative (or that their results be reweighted in the analysis to achieve this representative aspect). Furthermore, all assumptions and all selections made to bound the nondeterministic elements of the problem should be well documented so that inferences based on the resulting analyses are understood to be conditional on these assumptions.

As shown in Fig. 5, we identify four activities in this phase: discretization of the conservation laws (PDEs), discretization of the boundary and initial conditions, selection of propagation methods, and the design of the computer experiments. Errors that occur in the discretization

processes can be very difficult to isolate for a complex physical process or a sophisticated numerical method. In finite differencing, one method of identifying these errors is to analytically prove whether the method is consistent: that is, does the finite difference method approach the continuum equations as the step size approaches zero? For simple differencing methods, this is straightforward. For complex differencing methods such as essentially non-oscillatory schemes and second-order, multidimensional upwind schemes, determining the consistency of the algorithms for a wide range of flow conditions and geometries is difficult.

Several related issues are also treated as part of the discretization activities: Are the conservation laws satisfied for finite grid sizes? Does the numerical damping approach zero as the mesh size approaches zero? Do aliasing errors exist for zero mesh size? Note that discretization of PDEs are also involved in the conversion of Neumann and Robin's, i.e., derivative, boundary conditions to difference equations. We have included the conversion of continuum initial conditions to discrete initial conditions not because there are derivatives involved, but because spatial singularities may be part of the initial conditions. An example is the decay of vortex for which the initial condition is given as a singularity. Our point is also valid, indeed much more common, when singularities or discontinuities are specified as boundary conditions. Some may argue that because these discontinuities and boundary singularities do not actually occur in nature, it is superfluous to be concerned about whether they are accurately represented. This argument misses the point completely. If these nonlinear features exist *in the continuum mathematical model of the physics*, the issue is whether the discrete model represents them accurately, not whether they exist in nature. In other words, the focus should be on verification (solving the problem right), as opposed to validation (solving the right problem).

The two activities in this phase that address nondeterministic elements and their values are selection of propagation methods and design of computer experiments. Both address the conversion of the nondeterministic elements of the analysis into multiple runs, or solutions, of a deterministic computational simulation code. Selection of a propagation method involves the determination of an approach, or approaches, to propagating variabilities and uncertainties through the computational phases of the analysis. Examples of methods for propagating variabilities include: reliability methods;[41] sampling methods such as Monte Carlo or Latin Hypercube;[78, 79] or statistical design approaches.[80] Methods for the propagation of uncertainties defined using nonprobabilistic representations, e.g., possibility theory and fuzzy sets, are a subject of current research.[64, 81-83] The design of computer experiments that is performed as a part of this phase is driven to a large extent by the availability of resources and by the requirements of the analysis. Establishing an experimental design often involves more than just an implementation of the propagation method specified above. The problems associated with large analyses can often be decomposed in a way that permits some variables and parameters to be investigated using only portions of the code or, perhaps, simpler models than are required for others. This decomposition of the problem and selection of appropriate models, together with the formal determination of inputs for the computer runs, can have a major effect on the uncertainty introduced into the analysis in this phase.

3.5 Computer Programming Activities

The correctness of the computer programming phase is most influenced by unacknowledged errors, i.e., mistakes. In Fig. 5 we have identified three basic activities: input preparation, module

design and coding, and compilation and linkage. The topic of reducing unacknowledged errors, i.e., mistakes, in this phase is thoroughly covered in many software quality assurance texts.[73, 74] This does not mean, however, this phase is a trivial element of modeling and simulation. Some computational researchers experienced only with model problems, even large-scale model problems, do not appreciate the magnitude of the issue. They feel it is simply a matter of carelessness that can easily be remedied by quality assurance practices. The high number of inconsistencies, static errors, and dynamic, i.e., run-time, errors in well tested commercial computer codes was recently investigated by Hatton.[84] He conducted two major studies of the reliability and consistency of commercial science and engineering software. One set of tests evaluated coding defects without running the code; i.e., static tests. The other set of tests evaluated the agreement of several different codes which used different implementations of the same algorithms, acting on the same input data. Note that all of these tests were *verification* tests; none used experimental data. He has concluded: "All the evidence ... suggest that the current state of software implementations of scientific activity is rather worse that we would ever dare to fear, but at least we are forewarned."

The capturing and elimination of programming errors, while not generating much excitement with computational researchers, remains a major cost factor in producing highly verified software. Even with the maturity of the software quality assurance methods, assessing software quality is becoming more difficult because of massively parallel computers. In our opinion, the complexities of optimizing compilers for these machines, of message passing, and of memory sharing are increasing faster than the capabilities of software quality assessment tools. As a case in point, debugging computer codes on massively parallel computers is moving toward becoming a nondeterministic process. That is, the code does not execute identically from one run to another because of other jobs executing on the massively parallel machine. It is still a fundamental theorem of programming that the correctness of a computer code and its input cannot be proven, except for trivial problems.

3.6 Numerical Solution Activities

As shown in Fig. 5, we have identified four activities occurring in the numerical solution phase: spatial grid and time-step convergence, iterative convergence, nondeterministic propagation convergence, and computer round-off. The primary deficiency that occurs in this phase is the occurrence of acknowledged errors. Numerical solution errors have been investigated longer and in more depth than any other errors discussed previously. Indeed, they have been investigated since the beginning of computational solutions.[85] These deficiencies in the solution of the discrete equations are properly called errors because they are approximations to the solutions of the original PDEs.

Of the four activities listed in the numerical solution phase, perhaps the one that requires most explanation is iterative convergence. By this we mean the finite accuracy to which nonlinear algebraic, or transcendental, discrete equations are solved. Iterative convergence error normally occurs in two different procedures of the numerical solution: 1) during the iterative convergence that must be achieved within a time step and 2) during the global iterative convergence of an elliptic PDE. Examples of the iterative convergence that must be achieved during a time step are intra-time-step iteration to solve the unsteady heat conduction equation when the thermal conductivity depends on temperature, intra-time-step iteration to determine the liquid-solid boundary in a

melting or solidification problem, and the iterative solution for nonlinear analytic expressions for transport or thermodynamic properties. In finite volume schemes, for example, conservation of mass, momentum, and energy can be violated with inadequate iterative convergence at each time step.

Nondeterministic propagation convergence refers to activities related to adjustments in, or further specification of, inputs determining the multiple deterministic computer runs. Some methods for uncertainty propagation and experimental design rely on run-time results to help direct further computer experimentation. Reliability methods, for example, focus on finding a specific point (for functional expansion) that provides a 'best approximation' to system performance.

Although we identified four activities in the numerical solution phase, these activities are seen to produce three types of acknowledged errors. The first error type is due to the discretized solution of the PDEs; spatial grid error and time-step error are of this type. The second type is a consequence of reducing the initial nondeterministic problem with infinitely many solutions to a manageable problem with a finite set of representative analyses and responses. Any finite number of deterministic solutions produces an error of this type. The third type is due to the approximate solution of the discrete equations, that is, those errors made in the solution to the given discrete equations. Iterative convergence and round-off errors are of this type and they account for the difference between the exact solution to the discrete equations and the computer solution obtained.

All texts dealing with the numerical solution of PDEs address the topic of estimating the magnitude of the spatial grid convergence error. Some of these texts also deal with the errors associated with temporal convergence, iterative convergence, and round-off. Even though grid convergence error is fairly well understood, it is our view that it is commonly the largest contributor to error in numerical simulations. The reason for this paradox is simple: cost. The grid size used for a numerical solution is commonly at the limit of computer time or budgetary constraints. For large-scale engineering simulations, such as a fire in a building or the thermal-hydraulic safety analysis in a nuclear reactor, the grid is far from spatially resolved. The grid is simply considered "acceptable" for the simulation at hand and the solution is used. Our point is not that these type simulations are "useless," but that improved understanding is needed about how these solutions should be used.

3.7 Solution Representation Activities

In the solution representation phase shown in Fig. 5, we have identified five activities: post-processor input, preparation, module design and coding, compilation and linkage, data representation, and data interpretation. Input and programming and compilation activities are dominated by unacknowledged errors, as pointed out previously in our discussion of the computer programming phase.

Data representation errors originate as a result of the inaccurate or inappropriate construction of continuous functions from the discrete solution in the post-processor. Examples are oscillations of the continuous function in between discrete solution points due to the use of a high-order polynomial function in the post-processor, extrapolation of solution variables outside the discrete solution domain of independent variables, and inappropriate interpolation of the discrete solution between multiblock grids. We believe that these are acknowledged errors based on the question:

"What is the mathematically correct reconstruction of the continuum functions from the PDEs using the discrete solution points?" When viewed from this perspective, one becomes concerned about the issue because this is not the perspective taken in modern data visualization packages. The view of these general purpose packages is that there is no connection between the two. Reconstruction is done based on speed, convenience, and robustness of the package.

Data interpretation errors are made by the interpreter, i.e., the user, based on observation of the representation of the numerical solution. In other words, data interpretation errors occur when a user incorrectly interprets the numerical solution. Consequently, these errors should be viewed as unacknowledged errors. An example of an interpretation error is concluding that a predicted solution is chaotic when it is not (and vice versa). Importantly, our definition of data interpretation errors does not include inappropriate decisions made by the user based on the simulation, such as incorrect design choices or inept policy decisions.

3.8 Summary Comments

In this example we have identified the activities conducted in each phase of modeling and simulation and we have discussed a number of variabilities, uncertainties, and errors that occur in different phases. Drawing distinctions among variability, uncertainty, and error is, we believe, crucial for the correct representation and propagation of sources through the modeling and simulation process. In addition, such distinctions provide a natural path for considering the reduction or possible elimination of sources. If a source is a variability, then the magnitude of variability might be reduced by a procedural change, for example, a change in quality control requirements or restriction of activities during certain weather conditions. If a source is an uncertainty, its magnitude could be reduced by gathering more information, for example, obtaining additional experimental data or improving the modeling of a physical phenomenon. If a source is an acknowledged error, its magnitude could be reduced, for example, by adding more computations with higher grid resolution. If a source is an unacknowledged error, it could be found and eliminated, for example, by independently checking the correctness of input data in a large-scale simulation.

Once a large-scale simulation has been completed and the results are applied to their intended purposes, then a more global perspective of total variability, uncertainty, and error should be adopted. For example, suppose that a large-scale simulation was conducted that involved thousands of individual computations. Many of these computations may be part of a Monte Carlo simulation to predict the effects of multiple sources of variability and uncertainty. Once all of the sources are aggregated, or convolved, into a global modeling and simulation result, then the users of the results may not be concerned with the contribution of each source. At this point, it is not generally useful to continue to distinguish them. Consequently, we recommend that the abbreviated term "total uncertainty" or “prediction uncertainty” be used to mean "Total variability, uncertainty, and error in modeling and simulation predictions."

4. Missile Flight Example

4.1 Description of the Problem

In this example we consider an analysis of the flight of a rocket-boosted, aircraft-launched missile. In our analysis we make the following assumptions about the missile:

1) The missile is unguided during its entire flight, i.e., only ballistic flight is considered.
2) The missile is propelled by a solid fuel rocket motor for the initial portion of its flight, and it is unpowered during the remainder of the flight.
3) The missile is fired from a launch rail attached to the aircraft in flight.
4) The only aerodynamic surfaces on the missile are fins to provide flight stability.

The analysis considers the missile flight to be in the unspecified future, i.e., the analysis is an attempt to predict future plausible events, not analyze an event in the past.

Figure 5 illustrates the activities that are conducted in each of the six phases of modeling and simulation. Also shown for each activity are the dominant sources of variability, uncertainty, and error that typically occur in each activity. We now discuss in detail the activities that are conducted in each of the phases and explain how these activities are applied to the missile flight example.

4.2 Conceptual Modeling Activities

As seen in Fig. 5, we have identified four major activities that are conducted in the conceptual modeling phase: system/environment specification, scenario abstraction, coupled physics specification, and nondeterministic specifications. The system/environment specification activity consists primarily of careful specification of what physical or conceptual elements are considered as part of the system and what are considered part of the environment. When we say physical or conceptual elements are part of the system we mean that it is possible that any of the elements can interact with one another. This concept is similar to a system as defined in thermodynamics. The state of a system is influenced by processes internal to the system, i. e., endogenous processes, and also processes or activities external to the system, i. e., exogenous effects. Exogenous processes or activities are considered to be part of the environment. A system is influenced by the environment, but the environment *cannot be* influenced by the system.[32] In other words, the system and the environment do not *interact*; the system can respond to the environment, but the environment cannot respond to the system. System/environment specifications are a matter of engineering judgement and are not unique. As a result, these specifications pose one of the most difficult conceptual issues in modeling and simulation.

Figure 6 shows three possible system/environment specifications for the example problem. They are listed from the most inclusive to least inclusive. System/Environment Specification 1 considers the missile and the atmosphere near the missile to be part of the system, whereas the launching aircraft and target are considered part of the environment. An example of an analysis that would be allowed with this specification is the flow field of the missile and the rocket exhaust is coupled to flow field of the launching aircraft, i.e., the missile and rocket exhaust could interact

with the aircraft flow field, but the aircraft structure could not change its deformation due to the rocket exhaust. Another example allowed by this system/environment specification would be the analysis of the missile flight inside an enclosure or tunnel, e.g., near the target. System/ Environment Specification 2 considers the missile and the aerothermal processes occurring on the missile to be part of the system, whereas the atmosphere near the missile, the launching aircraft, and the target are considered part of the environment. This specification allows analyses that couple the missile and its aerothermal effects. For example, one could consider the deformation of the missile due to aerodynamic loading and thermal heating, and then couple these deformations into recomputing the aerodynamic loading and thermal heating. System/Environment Specification 3 considers the missile to be the system, whereas the aerothermal processes, the atmosphere near the missile, the launching aircraft, and the target are considered part of the environment. Even though this is the simplest specification considered, it still allows for significant complexities in the analysis.

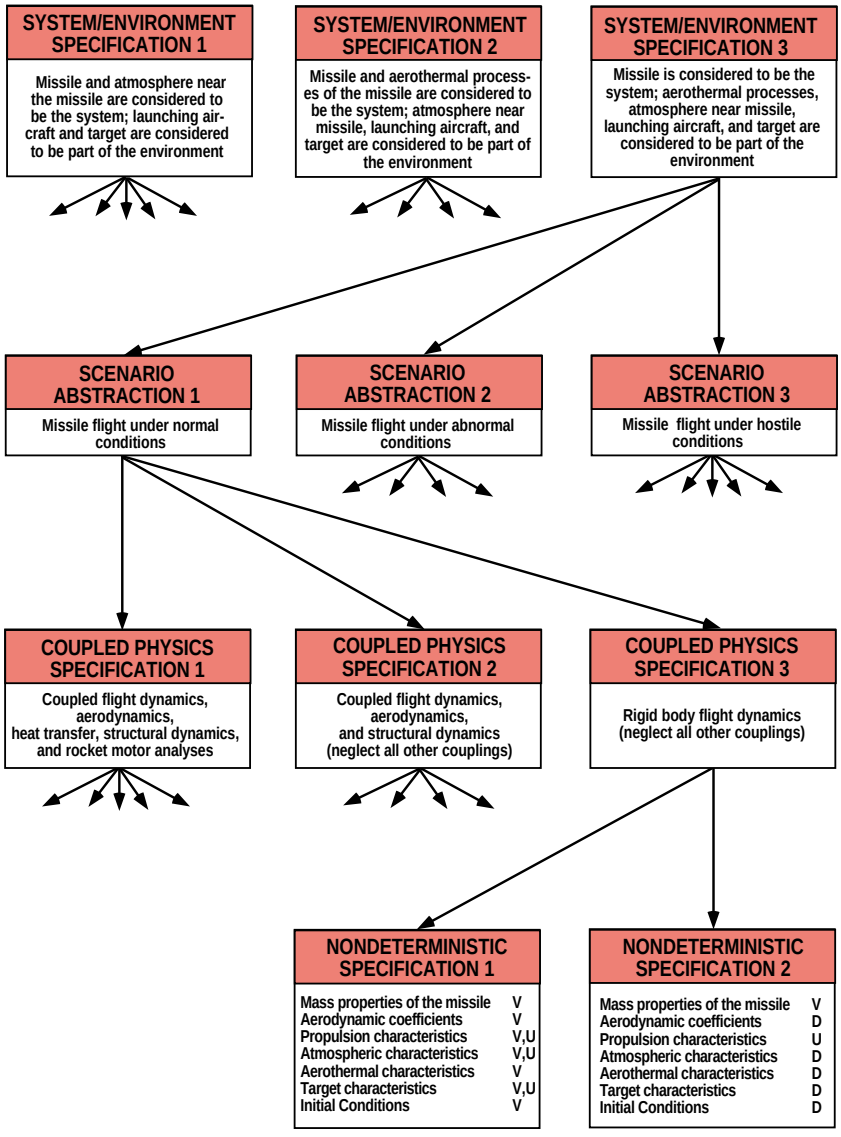

Figure 6
Conceptual Modeling Activities for the Missile Flight Example

Scenario abstraction attempts to identify all possible physical events, or sequences of events, that may affect the goals of the analysis. For relatively simple systems, isolated systems, or systems with very controlled environments or operational conditions, scenario abstraction can be straightforward. Complex engineered systems, however, can be exposed to a variety of natural and abnormal operating conditions, hostile environments, or a myriad of human-caused or accidentally caused failure modes. Scenario abstraction for these complex systems is a mammoth undertaking. The field of engineering that has achieved the highest development of scenario abstraction is probabilistic risk assessment (PRA) of nuclear power plants. PRA techniques construct a many-branched event tree for complex operating and failure scenarios. Even though the probability of occurrence of certain events may be extremely low, these events must be considered and analyzed for failure of nuclear power plants and other high consequence systems. The scenario abstraction considered here includes both event tree construction and decision tree construction, that is, decision tree construction does not necessarily depend on events, but it can identify possible results based on decisions or analyses that could be pursued.

As shown in Fig. 6, the missile flight example identifies three broad classes of scenarios; missile flight under normal, abnormal, and hostile conditions. Normal conditions are those that can be reasonably expected, such as typical launch conditions from aircraft that are expected to carry the missile, near nominal operation of the propulsion system, and reasonably expected weather conditions. Examples of flight under abnormal conditions would be improperly assembled missile components; explosive failure of the propulsion system; and flight through adverse weather conditions, such as snow or lightning. Examples of flight under hostile conditions would be detonation of near by explosive systems, damage to missile components resulting from small arms fire, and damage from laser or microwave defensive systems. The three scenario categories considered here have been commonly used for military systems, e.g., nuclear weapons. With the increasing concern of terrorist attacks on civilian systems, such as buildings, commercial aircraft, bridges, and dams, this categorization may, unfortunately, prove to be more broadly useful in the future.

Coupled physics specification, Fig. 5, identifies and carefully distinguishes the possible alternatives for physical and chemical processes in the system, and the coupling between them for the system/environment specification and scenario abstraction under consideration. A clear statement of the possible levels of physics coupling is required because of the wide variety of physics that may occur in a complex system. In the missile flight example, Fig. 6, we identify three levels of physics coupling, although more alternatives could be identified. Coupled Physics Specification 1 couples essentially all of the physics that could exist in the problem. For example, this specification could couple the structural deformation and dynamics with the aerodynamic loading and thermal loading due to atmospheric heating. It could also couple the deformation of the solid fuel rocket motor case due to combustion pressurization, the heat transfer from the motor case into the missile airframe structure, and the non-rigid body flight dynamics on the missile. Coupled Physics Specification 2 couples the missile flight dynamics, aerodynamics, and structural dynamics, neglecting all other couplings. This coupling permits the computation of the deformation of the missile structure due to inertial loading and aerodynamic loading and then the aerodynamic loading and aerodynamic damping due to the deformed structure. This would be a coupled, time dependent, fluids/structures interaction simulation. Coupled physics Specification 3 assumes a rigid missile body; not only is physics coupling disallowed, but the missile structure is assumed rigid. The missile only is allowed to respond to inputs or forcing functions from the environment.

Structural dynamics is removed from the analysis; i.e., only rigid body dynamics is considered.

Before we consider the last activity, nondeterministic specification, consider the possible sources of total uncertainty that could occur in the three activities discussed so far. System/environment specification and scenario abstraction activities introduce uncertainties into the modeling and simulation process. This occurs primarily because of what is *not* included, or scenarios that are *not* imagined. The wider the scope of the analysis or the more complex the system, the more possibilities there are for uncertainties due to lack of knowledge about aspects of the modeled system and environment. Indeed, an underlying weakness of modern technological analysis is events, effects, and possibilities not initially considered.[86] For example, automatic control systems designed to ensure safe operation of complex systems during normal operation or during maintenance of the system can fail, or the safety system can be overridden. During construction of coupled physics specifications, the primary source of total uncertainty introduced into the analysis is acknowledged error. A hierarchical ordering of levels of physical coupling in conceptual models can commonly be constructed. Based on experience with similar systems, previous analyses, risk of adverse safety, performance, economic consequences, and budget and schedule considerations, decisions are then made concerning which physics coupling is chosen. However, when physics couplings are neglected, an acknowledged error is introduced.

In the nondeterministic specification activity, Fig. 5, decisions are made concerning what aspects of the system and environment will be considered deterministic or nondeterministic. A deterministic system and environment exhibits one system response given a specification of all mathematical models and parameters of the system. Certain nonlinear systems, even when all models and parameters are specified, can yield multiple responses or even chaotic responses. We include these types of systems in the class of nondeterministic systems, although they are not addressed in the present work. The predominant cause of nondeterministic system response is inherent randomness in model parameters, initial conditions, or parameters specifying the environment. We have referred to these types of sources producing nondeterministic features as variabilities. Uncertainties in model parameters, initial conditions, or the environment also occur because of a lack of knowledge about these factors. The uncertainty can often be reduced by obtaining relevant data or by taking actions that limit the state of key elements of the system or the environment. Often such actions are policy or procedural decisions. Examples of these cases are: Was the system incorrectly manufactured or assembled? Has inadequate or improperly conducted system maintenance significantly altered certain system characteristics? Was the system damaged in the past, but the damage is unknown? In other situations it may not be possible to reduce the lack of knowledge, and reduce the uncertainty.

For the missile flight example we list only two alternative nondeterministic specifications shown in Fig. 6. Nondeterministic Specification 1 includes the following variabilities (indicated by a V in Fig. 6): mass properties of the missile, aerodynamic force and moment coefficients, aerothermal heating characteristics, and initial conditions at missile launch. These are considered variabilities because they are usually associated with random variation due to manufacturing processes or physically random processes. If a large number of missiles are manufactured, for example, sufficient inspection data would normally exist so that a representative probability distribution for each parameter could be constructed. Nondeterministic Specification 1 also includes the following as variabilities and/or uncertainties (indicated by a V,U in Fig. 6): propulsion characteristics, atmospheric characteristics, and target characteristics. These quantities

could be considered as variabilities, but their nondeterministic feature is usually dominated by lack of knowledge. For example, propulsion characteristics of solid rocket motors can vary substantially with age and temperature of the propellant. Suppose that statistical models which incorporate age and temperature of the propellant have been constructed. If the age of the propellant in a particular motor is not known or the temperature of the propellant is not known, a statistical model is of little value in estimating the variation in the performance of the motor. A similar argument can be made for estimating the uncertainty in atmospheric characteristics, for example, wind conditions. Without specifying additional knowledge, such as location on earth, month of the year, or even time of the day, statistical models are of limited value.

For nondeterministic Specification 2, Fig. 6, we chose one parameter of the mass properties of the missile as a variability, and one characteristic of the propulsion system as an uncertainty. We pursue Specification 2 in the example problem to distinguish the characteristics of each and to show how they might be represented differently in a computational simulation. All other parameters are considered deterministic (indicated by a D in Fig. 6).

4.3 Mathematical Modeling Activities

As shown in Fig. 5, we have identified four major activities in the mathematical modeling phase: formulation of the partial differential equations, choice of all the auxiliary equations that supplement the differential equations, formulation of all the initial and boundary conditions required to solve the PDEs, and selection of the mathematical representation of nondeterministic elements of the analysis. The PDEs commonly represent conservation equations for mass, momentum, and energy, but they can originate from any mathematical model of the system. The auxiliary equations are equations which are required to complete the PDEs. Examples of these would be turbulence modeling equations in fluid dynamics, equations of state in hydrodynamics, material constitutive equations in solid dynamics, and neutron cross-sections in neutron transport. The auxiliary equations can be of any type, e.g., algebraic equations, integral equations, or PDEs. The boundary and initial conditions provide the required closure equations needed for all PDEs.

Formulation of the nondeterministic representations is based on the needs of the analysis together with the quantity and quality of relevant information available. When the nondeterministic specification indicates that a range of values are of interest for a parameter or characteristic of the analysis, it may or may not include constraints on this range. Within these constraints an appropriate representation for the nondeterministic element will depend on available and/or obtainable relevant information. Probabilistic models (distributions or frequency functions) are appropriate only when enough information is available to determine relative likelihoods of different values over a range. In the absence of this information, one attempts to produce bounding values, or may hypothesize distributions for these elements and perform further analysis conditional on these bounding values or hypothesized distributions.

Variabilities commonly dominate the nondeterministic features of the auxiliary physical equations and boundary and initial condition activities. The most common variabilities are those due to inherent randomness of continuous parameters in these equations. Variabilities are nearly always represented by probability distributions. In some cases the form of these distributions are inferred from first principles of the processes involved in determining the parameter values. In

most cases the distributions are chosen based on convenience. Parameters associated with the probability distributions are then estimated when sufficient data are available or assigned values based on a subjective assessment when insufficient data are available.

Uncertainties can have a large impact on the nondeterministic formulation of the PDEs because the key issue can be limited, or inadequate, knowledge of the physical processes involved. Examples of uncertainties that occur in the PDEs are limited knowledge of the equations for turbulent reacting flow, conflicting models for crack propagation in materials, and competing models for elastic deformation of composite materials. For physical processes that are well understood, deficiencies in certain models should be considered as errors rather than uncertainties. This guideline is based on the argument that if significant knowledge of the process exists, a set of alternative models can be convincingly ordered in terms of increasing accuracy. In the modeling of fluid dynamic turbulence, the models can be generally ordered in terms of increasing accuracy as follows: algebraic models, two-equation models, Reynolds stress models, and large eddy simulation models. In general, this ordering is appropriate, but for individual flow fields there is no guarantee that any one model will be more accurate than any other because certain lower order models can be very accurate for specific cases.

Acknowledged errors in PDE models are those due to mathematically representing the physics in a more simplified or approximate form than the best available. It is invariably the case that for any mathematical model chosen to represent some physical process, one can identify higher fidelity models that are known to exist. In our definitions given in Section 2.2, this is precisely what is meant by acknowledged error. Higher fidelity models are usually not chosen because of the higher computational costs associated with their solution. The ratio of computational cost for a higher fidelity model to a lower fidelity model is commonly high, sometimes exceeding a factor of a 100. Analysts ordinarily choose a given level of model fidelity based on practical issues, such as computational resources available, options in computer codes they are familiar with, and schedule constraints, as well as technical issues. Some examples of acknowledged errors in mathematical modeling are the modeling of a process in two spatial dimensions when three spatial dimensions may be needed; the assumption of a steady state when unsteady effects may be important; and the assumption of homogenous material properties when mesoscale features play a substantial part. These examples of acknowledged errors are all characteristic of situations in which physical modeling approximations were made to simplify the mathematical model and the subsequent solution.

For the missile flight example, two mathematical models are chosen; a six-degree-of-freedom (6-DOF) model and a three-degree-of-freedom (3-DOF) model, Fig. 7. Both models are consistent with the conceptual model being analyzed: System/Environment Specification 3, Scenario Specification 1, Coupled Physics Specification 3, and Nondeterministic Specification 2 (Fig. 6). The translational equations of motion can be written as

$$m \frac{d\vec{V}}{dt} = \sum \vec{F}, \tag{1}$$

where m is the mass of the vehicle, $\vec{V}$ is the velocity, and $\sum \vec{F}$ is the sum of all forces acting on the vehicle. The rotational equations of motion can be written as

$$[\boldsymbol{I}]\frac{d\vec{\boldsymbol{\omega}}}{dt} = \sum \vec{\boldsymbol{M}} + \vec{\boldsymbol{\omega}} \times \left\{ [\boldsymbol{I}] \cdot \vec{\boldsymbol{\omega}} \right\}, \tag{2}$$

where $[\boldsymbol{I}]$ is the the inertia tensor of the vehicle, $\vec{\boldsymbol{\omega}}$ is the angular velocity, and $\sum \vec{\boldsymbol{M}}$ is the sum of all moments acting on the vehicle. Eq. (1) represents the 3-DOF equations of motion, and Eqs. (1) and (2) represent the 6-DOF equations of motion. A brief description of the derivation of the 3-DOF and 6-DOF equations of motion are given in Appendix A. Although these are ordinary differential equation models instead of the PDE models stressed in the present work, key aspects of the present framework can still be exercised.

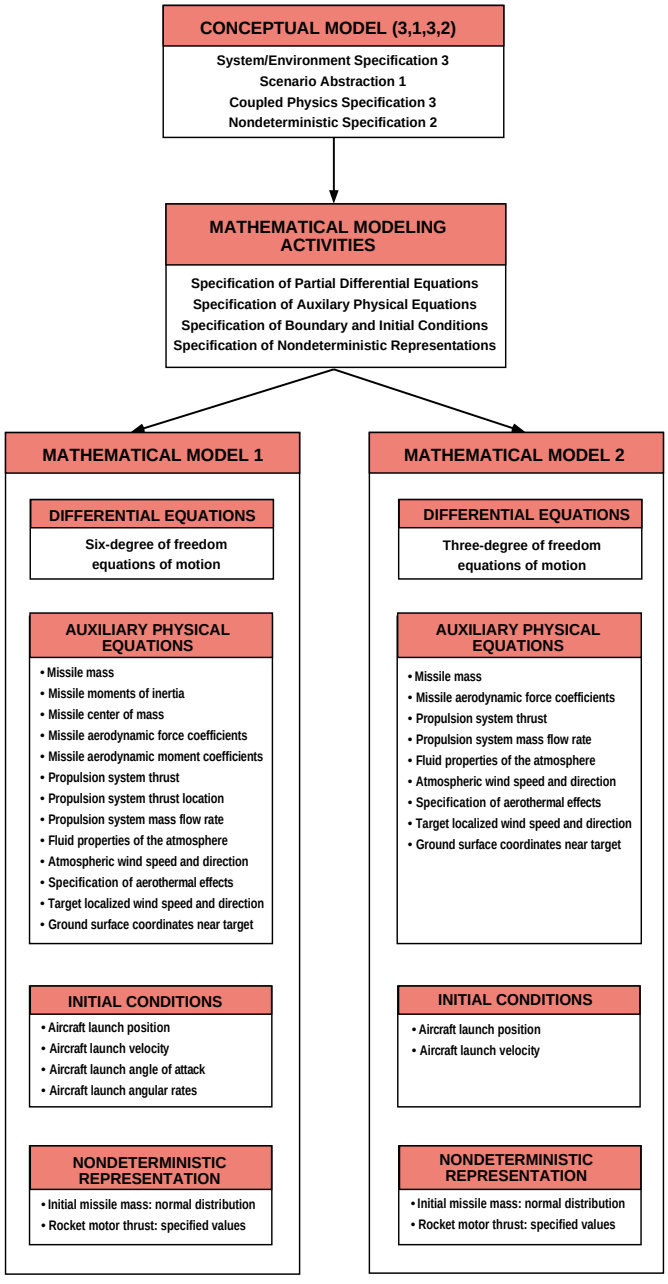

Figure 7
Mathematical Models for the Missile Flight Example

For these two mathematical models of flight dynamics, one can unequivocally order the models in terms of fidelity. Indeed, the physics and mathematics of the 6-DOF equations are so well understood that there is no need for experimental validation of these models. Their accuracy is only limited by the accuracy of the assumption of a rigid body, accuracy of the measured mass properties, and accuracy of the forces and moments acting on the vehicle. However, as mentioned above, for models of complex physical processes or systems this ordering is commonly not possible.

Figure 7 lists all of the auxiliary equations and initial conditions that are needed for each mathematical model. As is to be expected of higher fidelity models, the 6-DOF model requires physical information beyond that which is required by the 3-DOF model. This poses the question: When does the lack of information of the additional needed parameters in a higher fidelity model counteract its accuracy when compared to a lower fidelity model? Although this question is not addressed in the present work, it is an issue that must be weighed in many analyses. It is fallacious to claim that the higher the fidelity of the physics model, the better the results. Uncertainty of parameters and greater computer resources needed to solve higher fidelity models are critical factors in total uncertainty estimation. In addition, constraints on computer resources can obviate the usefulness of a higher quality model.

Two nondeterministic parameters will be considered in the missile flight example: the initial mass of the missile and the propulsion thrust characteristics. Both parameters appear in each of the mathematical models chosen so that direct comparisons of their effect on each model can be made. It is assumed that sufficient inspection data of manufactured missiles is available for the missile mass to justify a normal distribution with known mean and standard deviation. Thrust characteristics are considered to be an uncertainty that derives from nonspecificity, i.e., multiple situations are possible.[55, 63] We assume the nonspecificity is due to the temperature of the solid propellant. We choose a nominal value and two bounding values: normal operating temperature, highest allowed temperature within the manufacturer's specification, and the lowest allowed temperature. The high temperature condition causes the thrust to be higher and the burn time to be shorter, and the low temperature condition causes the thrust to be lower and the burn time to be longer. We assume that the thrust versus time profiles of the high and low temperature motors are accurately known from experimental data. It is clear that the uncertainty in propulsion thrust can be steadily reduced as information is added to the analysis. For example, if the temperature at launch could be specified within some experimental measurement uncertainty, then the propulsion uncertainty could be greatly reduced or eliminated. If age of the specific motor to be fired were known, then the uncertainty in performance due to age of the propellant could be eliminated. With this combined level of information, one could characterize the propulsion characteristics as a variability.

4.4 Discretization and Algorithm Selection Activities

The discretization and algorithm selection phase accomplishes two related tasks. First, it converts the continuum mathematics model, i.e., the differential equations, into a discrete mathematics problem suitable for numerical solution. Second, it provides the methodology that will be used to determine how a discrete set of computer solutions can be most appropriately used to accommodate the nondeterministic features of the analysis. The conversion from continuous to

discrete mathematics is fundamentally a mathematics-approximation topic; errors and not uncertainties are the dominant loss-of-confidence issue in this phase. (Note that for the remainder of the report when we refer to "errors," we will only be referring to *acknowledged* errors, unless otherwise stated.) Some may question why this conversion process should be separated from the solution process. We argue that this conversion process is the root cause of more difficulties in the numerical solution of nonlinear PDEs than is generally realized.[75, 87] When traditional nondeterministic methods are applied to systems described by differential equations, then one is dealing with stochastic differential equations. The discrete solution to these type equations, however, is much less developed than for deterministic differential equations.[88]

As shown in Fig. 5, we identify four activities in the discretization and algorithm selection phase: discretization of the PDEs, discretization of the boundary conditions (BCs) and initial conditions (ICs), selection of the propagation methods, and design of computer experiments. The types of errors that should be identified in the discretization of the PDEs, BCs, and ICs are those associated with possible inconsistencies between the discrete form of the equations in the limit and the continuum form of the equations. This normally is evaluated by analytically proving that the numerical algorithm approaches the continuum equations as the discretization size approaches zero. For simple differencing methods, this is straightforward. For complex differencing schemes, such as essentially non-oscillatory schemes, flux limiter schemes, and second-order, multidimensional upwind schemes, determining the consistency of the schemes can be difficult. For complex multiphysics in coupled PDEs, it is impossible to prove. Related issues are also treated in the discretization activities of differential equations, such as: Are the conservation laws satisfied for finite spatial grid sizes, or are mass, momentum, and energy only conserved in the limit? Does the numerical damping approach zero as the mesh size approaches zero? Note that discretization of PDEs are also involved in the conversion of Neumann and Robin's, i.e., derivative, boundary conditions into discrete equations. We have included the conversion of continuum initial conditions to discrete initial conditions not because there are derivatives involved, but because spatial singularities may be part of the initial conditions. An example is the time dependent decay of a vortex for which the initial condition is given as a singularity. Our point is also valid, indeed much more common, when singularities or discontinuities are specified as part of the boundary conditions.

The selection of propagation methods and design of computer experiments in Fig. 5 both address the conversion of the nondeterministic elements of the analysis into multiple runs, or solutions, of a deterministic computational simulation code. Selection of a propagation method involves the determination of an approach, or approaches, to propagating variabilities and uncertainties through the computational phases of the analysis. Examples of methods for propagating variabilities include: reliability methods;[41] sampling methods such as Monte Carlo or Latin Hypercube;[78, 79] or statistical design approaches.[80] Methods for the propagation of uncertainties defined using non-probabilistic representations, e.g., possibility theory and fuzzy sets, are a subject of current research.[64, 81-83] The design of computer experiments task performed as a part of this phase is driven to a large extent by the availability of resources and by the requirements of the analysis. Establishing an experimental design often involves more than just implementation of the propagation method specified above. The problems associated with large analyses can often be decomposed in a way that permits some variables and parameters to be investigated using only portions of the code or, perhaps, simpler models than are required for others. This decomposition of the problem and selection of appropriate models, together with the

formal determination of inputs for the computer runs, can have a major effect on the estimate of uncertainty introduced into the analysis in this phase. This activity is performed here because this detailed specification of inputs and models will impact programming requirements, as well as the running of the computer model in the numerical solution phase. These tasks may be performed differently for different mathematical models and may involve the specification of probabilities associated with different model choices, where available information warrants specification of probabilities.

For the missile flight example, the same discretization method was applied to both 6-DOF and the 3-DOF mathematical models. This resulted in two discretized models, but they only differ in the differential equations being solved. A Runge-Kutta-Fehlberg 4(5) method was chosen to solve each system of ODEs.[89] The RKF method is fifth order accurate at each time step, and the integrator coefficients of Ref. 90 were used. The method provides an estimate of the local truncation error, i.e., truncation error at each step, so that the estimated numerical solution error can be directly controlled by adjusting the step size as the solution progresses. The local truncation error is computed by comparing a fourth order accurate solution with the fifth order accurate solution. A more detailed description of the numerical integration procedure in given in Appendix B.

The method chosen for propagation of variability was the Latin Hypercube Sampling (LHS) method. LHS is a random sampling method for choosing discrete values from a probabilistically defined nondeterministic variable or parameter, and often provides an advantage in efficiency over strict Monte Carlo sampling. However, that advantage is degraded by the fact that direct estimates of sampling error cannot be computed without replicating the LHS runs. For propagation of the uncertainty, we simply chose three possible propulsion characteristics to bound the solution and provide a nominal result. The experimental design task for this example is simple because one of our objectives is to compare models of different fidelity. Hence, the experimental design calls for performing the same number of Latin Hypercube calculations for both the 3-DOF and 6-DOF models. In an actual analysis this phase would include selecting how to mix computer runs between the 3-DOF and 6-DOF models and determination of how results from both models might be combined to maximize the value of the computations. This maximization process is a research topic of major importance for complex systems.

4.5 Computer Programming Activities

Figure 5 identifies three activities in the computer programming phase: input preparation, module design and coding, and compilation and linkage. Input preparation refers to the analyst's conversion of the mathematical and discrete model elements into equivalent data elements usable by the application code. The second and third activities relate to the building of the application code itself. Here subroutine modules are designed and implemented through a high-level programming language. This high-level code is then compiled into object code and linked to the operating system and libraries of additional object code to produce the final executable code.

The correctness of the computer programming phase is most influenced by unacknowledged errors, i.e., mistakes. The potential for mistakes in all three of these activities is enormous. In addition to the most obvious programming bugs (which still occur frequently, despite being obvious), there is the more subtle problem of undefined code behavior. This occurs when a particular code syntax is undefined within the programming language, leading to executable code

whose behavior is compiler-dependent. Compilation and linkage introduce the potential for further errors unbeknownst to the developer. Primary among these are bugs and errors in the numerous libraries of object code linked to the application. These libraries can range from the ubiquitous, such as trigonometric functions, to matrix inversion and the solution of special classes of ODEs and PDEs. Such libraries allow the developer to reuse previously developed data handling and numerical analysis algorithms. Unfortunately, the developer also inherits the undiscovered or undocumented errors in these libraries. There is also the possibility that the developer misunderstands or makes an error in the values passed to the library routines.

The computer code that was used for the missile flight example was the TAOS code.[91] This is a general-purpose flight dynamics code that can be used for a wide variety of guidance, control, and optimization problems for flight vehicles. We used only the ballistic flight option to solve both the 6-DOF and 3-DOF equations of motion. Concerns with coding, compilation, and linkage on massively parallel computers were not a factor in this example problem because program execution was performed only on Unix workstations.

4.6 Numerical Solution Activities

As shown in Fig. 5, we have identified four activities occurring in the numerical solution phase: spatial and temporal convergence, iterative convergence, nondeterministic propagation convergence, and computer round-off accumulation. Spatial and temporal convergence addresses the accuracy of numerical solutions using finite spatial grids and finite time steps. These two can be grouped into the general category of truncation error due to the discrete solution of PDEs. By iterative convergence we mean the finite accuracy to which nonlinear algebraic, or transcendental, discrete equations are solved. Iterative convergence error normally occurs in two different procedures of the numerical solution: 1) during the iterative convergence which must be achieved within a time step and 2) during the global iterative convergence of an elliptic PDE, i.e., a boundary value problem. Examples of the iterative convergence which must be achieved during a time step are: intra-time step iteration to solve the unsteady heat conduction equation when the thermal conductivity depends on temperature, and the iterative solution of nonlinear constitutive equations. Iterative convergence error is different from error caused by finite precision arithmetic, i.e., round-off error.

Nondeterministic propagation convergence refers to activities related to adjustments in, or further specification of, inputs determining specifics of the multiple deterministic computer runs. Some methods for uncertainty propagation and experimental design rely on run-time results to help direct further computer experimentation. Reliability methods, for example, focus on finding a specific point (for functional expansion) that provides a “best approximation” to system performance. Convergence to this point is determined by the change in the movement of the approximation to this point from one computer run to the next. It is clear that the nondeterministic propagation convergence error, as well as those discussed in the previous paragraph, are all acknowledged errors.

For the flight dynamics example, the numerical solution method used a variable time step so that the local truncation error could be directly controlled at each step. The local truncation error is estimated at each step for each state variable for each system of differential equations. For the 6-DOF model there are 12 state variables, and for the 3-DOF model there are 6 state variables. Before

a new time step can be accepted in the numerical solution, a relative error criterion must be met for each state variable. If the largest local truncation error of all the state variables is less than 0.6 of the error criterion, then the step size is increased. Quantification of local solution error is important not only to measure its impact on an individual solution, but also to precisely determine its interaction with the variability and uncertainty in the problem. In the solution of PDEs for complex systems, general procedures for estimating solution error are very difficult to develop and compute. Global estimates of a posteriori solution error are commonly made with finite element methods, but local error estimates are not available. For finite difference and finite volume methods Richardson's method can be used to estimate local truncation error, but this becomes quite computationally expensive for complex problems.

4.7 Solution Representation Activities

In the solution representation phase shown in Fig. 5, we have identified five activities: input preparation, module design and coding, compilation and linkage, data representation, and data interpretation. The first three activities are very similar to those discussed in the computer programming phase. The data representation task includes two types of similar activities: first, the representation of individual solutions over the independent variables of the PDEs and, second, a summary representation that combines elements of the multiple individual deterministic computer runs. Representation of individual solutions refers to the construction of a continuum solution based on the numerical solution at discrete points in space and time. Data representation errors originate as a result of the inaccurate or inappropriate construction of continuous functions from the discrete solution of the PDEs in the post processor. Examples are oscillations of the continuous function between discrete solution points due to the use of a high-order polynomial function in the post processor and interpolation of the discrete solution between multiblock grids such that conservation of mass, momentum, and energy are not conserved. Note that we mean inaccurate construction with respect to the discrete solution, *not* with respect to the continuum PDEs. To clarify this point, consider the numerical solution of a shock wave passing through a fluid or a solid and the shock wave is physically modeled as a discontinuity in the continuum PDEs. If the discretization method approximates the discontinuity with a continuous function, e.g., a shock capturing method, then in the discrete representation the shock wave is no longer discontinuous. As a result, the construction error should be judged with respect to the continuous function approximation of the discrete solution; the discontinuity was lost in the discretization and it cannot be recovered here.

Representation of a nondeterministic simulation from the individual deterministic computer runs refers to the compilation of these multiple solutions into statistical or probabilistic measures that can be used to address the requirements of the analysis. This can include developing summary descriptions of the solution and discriminating which parts of the represented solutions will be reported through tables and figures. Errors can occur in the representation of a nondeterministic solution as a result of integrating the ensemble of individual solutions in a way which is inconsistent with the specified propagation method. Data representation errors are principally acknowledged errors in that a correct or consistent discrete-to-continuum mapping is known from the choice of discretization methods.

The data interpretation activity refers to the human perceptions or impressions that are formed based on the observation of the represented solutions. If the perceptions or impressions are correct,

then knowledge or understanding is generated. If they are incorrect, then an unacknowledged error has occurred. In other words, data interpretation errors occur when a user incorrectly interprets the numerical solutions. Examples of interpretation errors are: concluding that a computed solution is chaotic when it is not, or interpreting a computed flow as turbulent when it is only a spurious numerical solution. Importantly, our definition of data interpretation errors does not include inappropriate decisions made by the user based on the interpretation, such as incorrect design choices or inept policy decisions.

4.8 Summary Comments

Figure 8 illustrates the multiple models, numerical solutions, and solution representations that are addressed in the missile flight example. As shown in the figure, six conceptual models are identified, many more are implied, but only one is selected for further development and analysis. This single conceptual model spawns two alternative mathematical descriptions, both of which are carried through the remaining phases of the modeling and simulation process. For simplicity, Figure 8 then shows the further development of only one of these mathematical models, although it is understood that identical development of Mathematical Model 1 is taking place in parallel with Mathematical Model 2. The discretization and programming phases identify alternative model choices that are not considered further. Continuing into the numerical solution phase, nondeterministic effects that were identified in the conceptual model and further defined in the mathematical modeling phase are computed via multiple deterministic numerical solutions. How these solutions were computed was specified in the propagation method identified in the discretization and algorithm selection phase. Finally, in the solution representation phase, the multiple solutions are reintegrated to represent the nondeterministic solution.

It is clear from the missile flight example that the modeling and simulation process for complex systems involves the identification and use of multiple scenarios, analyses, and computations. At each phase of this process it is often possible to identify more than one viable choice of models or parameters that can be used to obtain a computational result. As these multiple model choices propagate through subsequent phases, a tree structure of potential computational results is developed.

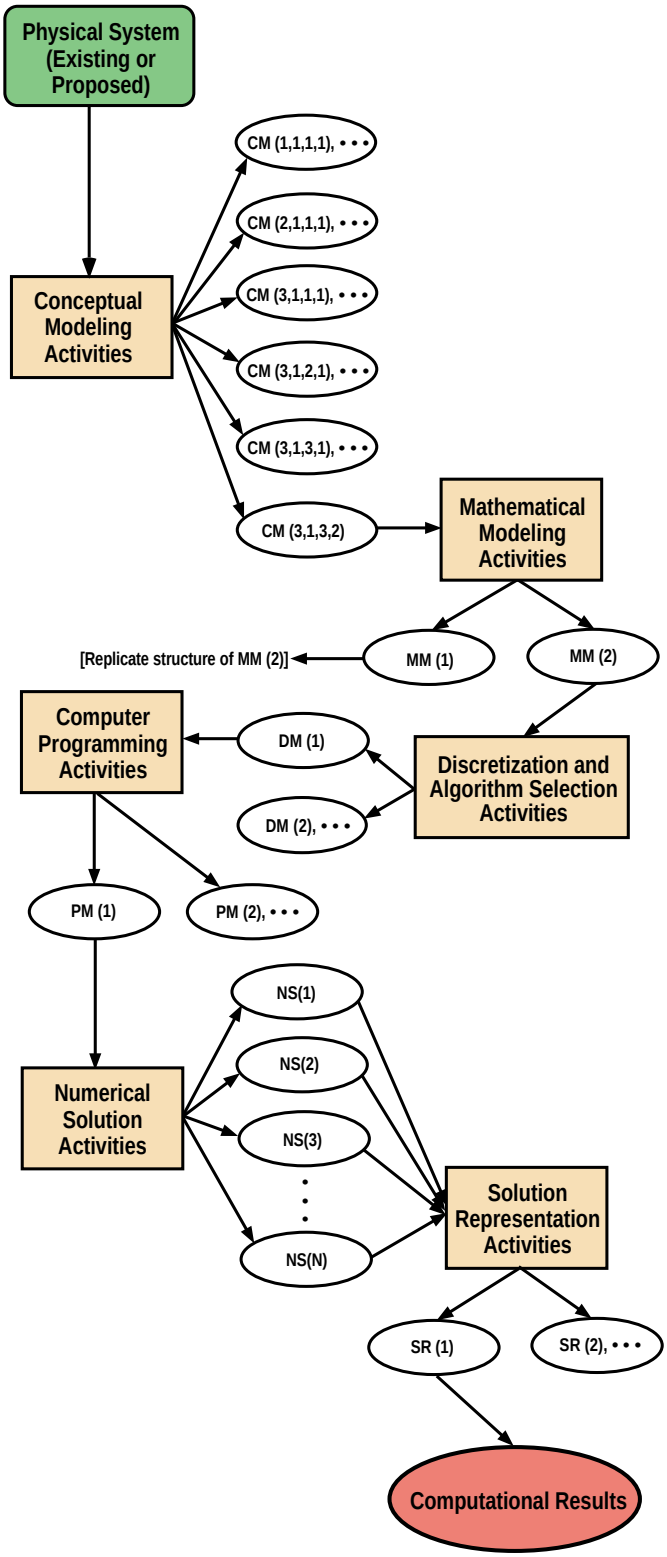

Figure 8
Tree-Structure for Models, Solutions, and Representations in the Missile Flight Example

5. Missile Flight Example Computational Results

Before the computational results from the missile flight example are presented, a few details must be given concerning the calculations. The missile is assumed to be launched from an aircraft flying straight and level at an altitude of 30 kft. above sea level and at a speed of 700 ft/sec. Assume a spherical, non-rotating earth. Define an earth fixed, three-dimensional, cartesian coordinate system, where x is vertical, z is in the direction of the aircraft flight, and y is normal to the xz plane (Fig. 9). Let the origin of the xyz coordinate system be at sea level, directly below the missile center-of-gravity at the initial condition. Assuming zero disturbance of the aircraft on the missile during launch and assuming uniform freestream flow approaching the missile, then the initial conditions for the 6-DOF equations of motion are

$x = 30,000$ ft., $y = z = 0.$
$V_x = V_y = 0.$, $V_z = 700$ ft/sec.
$\alpha = \beta = \phi = 0.$
$p = q = r = 0.$

α, β, and ϕ are the pitch, yaw, and roll angles of the missile, respectively. p, q, and r are the roll rate, pitch rate, and yaw rate of the missile, respectively. The initial conditions for the 3-DOF equations of motion are given by the x, y, z and V_x, V_y, and V_z conditions given above. Assume the fluid properties of the atmosphere are given by the 1976 U.S. Standard Atmosphere and that the winds are zero over the entire trajectory.[92] The trajectory calculation is terminated when $x = 0.$, i.e., at sea level.

For convenience, detailed missile characteristics were taken to be those of the Improved Hawk missile, since these were readily available.[93] Missile moments of inertia, center of mass, rocket motor thrust, and mass flow rate of the rocket motor are given in Appendix C. All of these parameters are functions of time during rocket motor operation but are constant after motor burnout. The rocket motor nominally operates for 24.5 sec., which is about half of the total flight time of the missile. The aerodynamic force and moment coefficient derivatives are assumed constant with pitch, yaw, and roll angle of the missile, i.e., linear aerodynamics is assumed. However, the aerodynamic force and moment coefficient derivatives are functions of Mach number. Detailed information on the aerodynamic force and moment coefficients is given in Appendix C. The system response measure discussed in the body of this report is the final range of the missile, since it captures most of the trajectory characteristics of interest. Detailed results of the flight dynamics are given in Appendix D and a sample input file for the TAOS code is given in Appendix E.

To illustrate the combined effects of variability, uncertainty, and error in the example, we select 500 values of initial mass through the Latin Hypercube Sample method. We then compute 500 values of range using combinations of the two mathematical models, the three thrust models, and five selected values of numerical integration solution error. Our purpose is to study and understand the effects and interactions of these sources of total uncertainty. We do not address in this report how all possible sources, many of which have be suggested earlier, could be represented and propagated in this flight dynamics example nor do we address the most appropriate way to summarize the effects of these total uncertainty sources for decision makers.

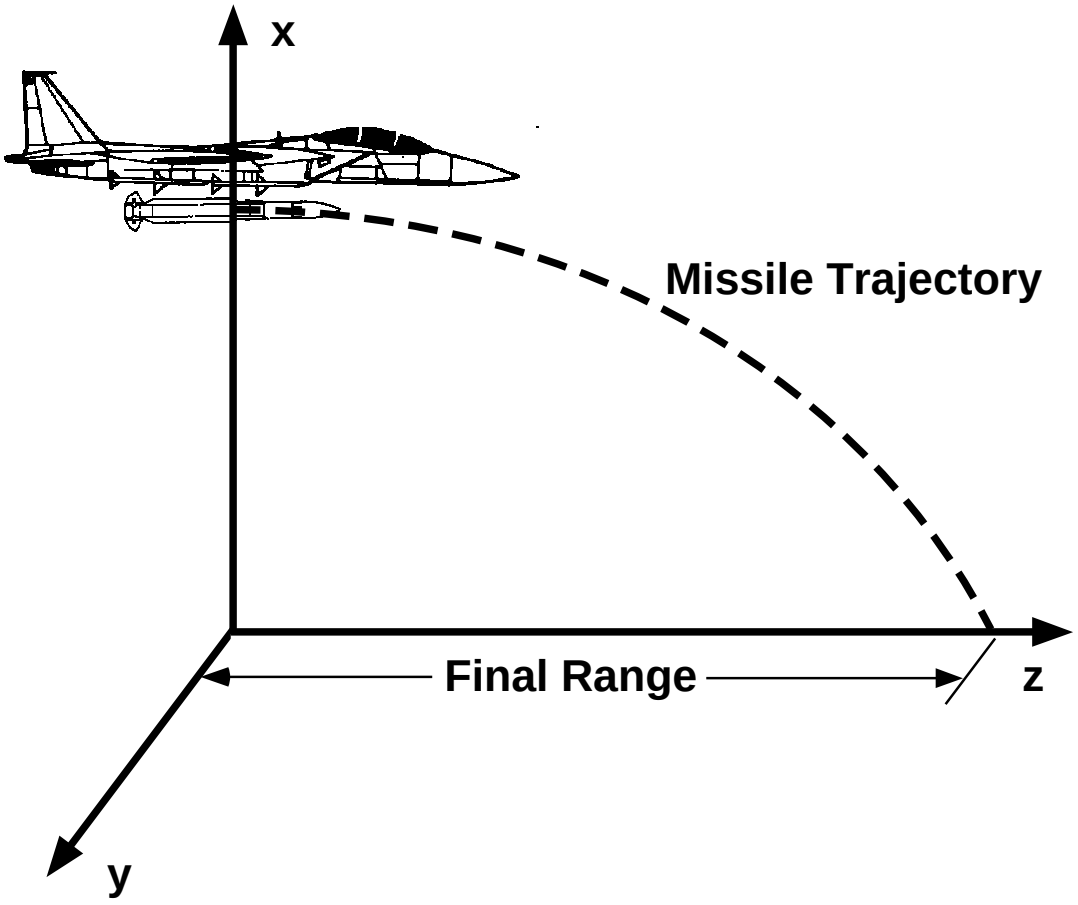

Figure 9
Cartesian Coordinate System for Missile Trajectory

5.1 Effects of Mass Variability

The first source of total modeling and simulation uncertainty examined was the variability of the initial mass of the missile. The mean initial mass was 1378.98 lb, of which 732 lb was inert mass and 647 lb was propellant. As mentioned in Section 3.3, a normal probability distribution for initial mass variability was assumed. The standard deviation, σ_W, was assumed to be 10 lb. Although it is not important for this example, σ_W = 10 lb is consistent with actual missile systems of this size.[94] We investigated the effects of numerical solution error for both the 6-DOF and 3-DOF models to be certain that this error was not entering into the mass variability results. We computed solutions with per step, relative, truncation error criteria of 10^{-12}, 10^{-9}, and 10^{-6}. Comparing these solutions at the end of the trajectory we found that error criteria of 10^{-12} and 10^{-9} produced the same values of the final range to seven significant digits. As a result, we used 10^{-9} for all remaining calculations when solution error was not of interest. Using this error criterion the computer run time on a SUN Sparc 20 workstation was 49 sec. and 1 sec., respectively, for one 6-DOF and one 3-DOF solution.

Since computer run time was not an issue, we computed 500 Latin Hypercube Sample (LHS) solutions for both the 6-DOF and 3-DOF models. Shown in Fig. 10 is the histogram from the LHS centering the weight at 1379 lb and using bins of width 5 lb. As can be seen with this number of samples, the histogram is a good approximation to the assumed normal distribution. Using LHS and 500 samples, the mean value was computed to be 1378.984 lb, and σ_W = 9.993 lb. The 500 samples is roughly a factor of 10 higher than is normally needed. We chose this large number to essentially eliminate any sampling error in the analysis. Since the same random number generator and the same seed were used on both the 6-DOF and 3-DOF models, each model computed trajectories using exactly the same missile weights. Indeed, for all results given in this report, exactly the same sampled initial missile weights were used.

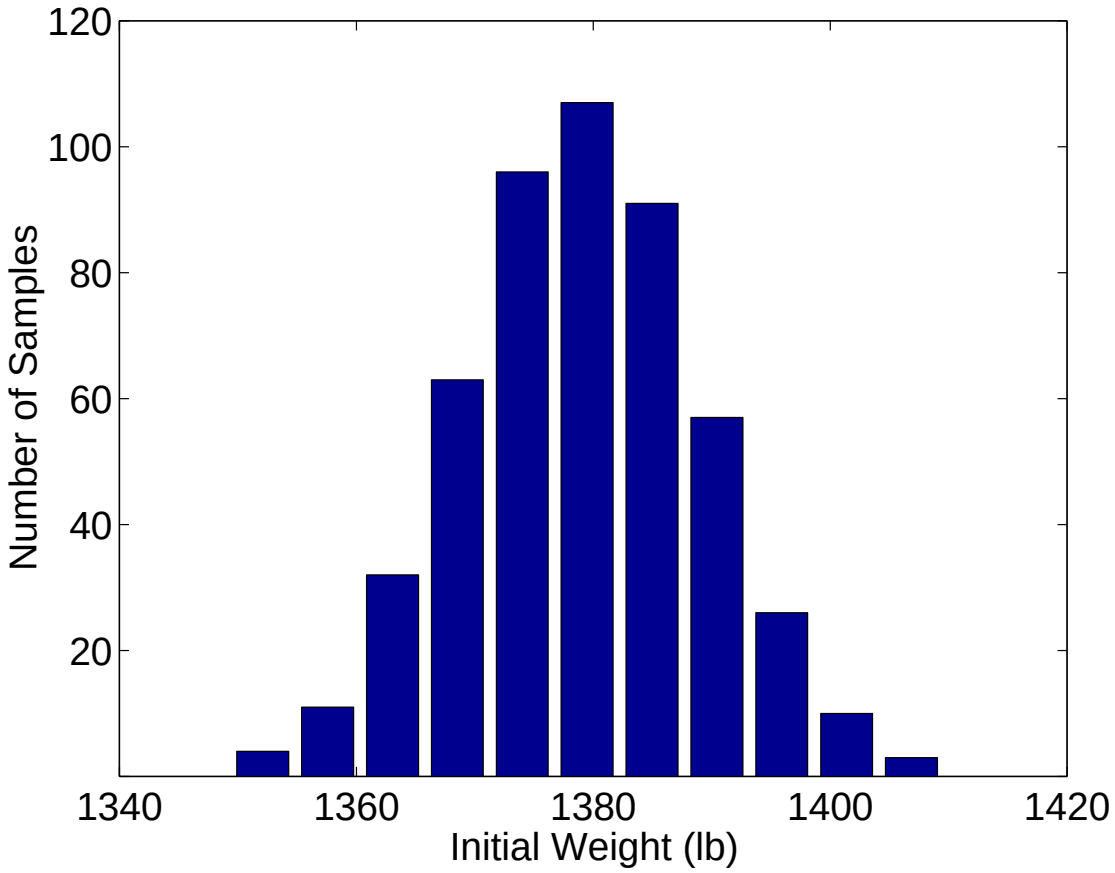

Figure 10
Histogram from LHS for Mass Variability

Figure 11 shows the computed range of the missile as a function of the initial mass for both the 6-DOF and 3-DOF models. The nominal thrust profile for the rocket motor was used. For both models, the missile range is linear for this small variation in initial weight. It is clear from this very well behaved system response measure that 5 to 10 LHS samples would have been typically sufficient to characterize this response measure. However, for our analysis we required sampling errors that were much less than typical analyses. It is also seen in Fig. 11 that the lower fidelity model (3-DOF) introduces a bias error of 0.040 nautical miles in range which is constant for all weights sampled. The generation of a bias error in the response of the system is disturbing because

it might go undetected if the higher fidelity model results or experimental measurements were not available. One does not, in general, expect this result. Lower fidelity models are used with the hope that the computational results will at least be distributed around the correct answer. For this relatively simple physics system one can easily see how this bias error in range occurs. The arching trajectory of the missile in a plane causes a small positive mean angle of attack during most of the trajectory. Computational results from the 6-DOF trajectory show this value to be about 0.01 to 0.02 deg. after the initial disturbance at launch decays (see Appendix D). This angle of attack causes a lift component on the missile, i.e., a small gliding effect, which results in a slightly longer trajectory. The lower fidelity model does not account for this physics, and as a result, the prediction of range is consistently shorter. From this understanding of the physics, one can then see that the magnitude of the bias will depend on a host of additional parameters that were not investigated, e.g., initial launch altitude, initial launch angle, and the aerodynamic lift coefficient.

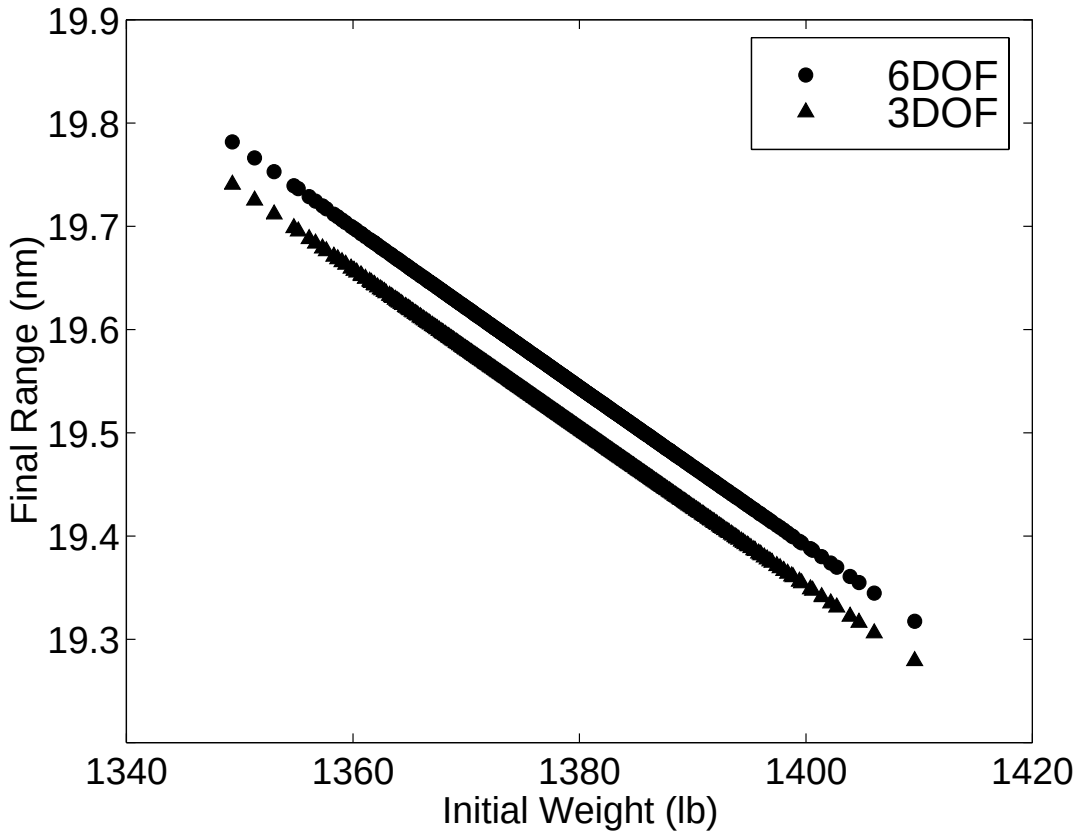

Figure 11
Variability in Range due to Variability in Initial Weight

Figure 12 shows frequency data of the LHS samples as a function of range off-set from the mean value range for the 6-DOF trajectory: 19.552 nm. That is, the range computed for the mean weight of 1378.98 lb for the 6-DOF trajectory is defined to have zero offset. In this figure the bias error in range of 0.040 nm of the 3-DOF model is seen as a shift of distribution to the left, i.e., shorter range. The frequency plot shows the distribution produced by each model is remarkably similar, as might be expected from the results of Fig. 11. For the 6-DOF model the standard deviation in range is computed to be 0.0770, whereas for the 3-DOF model, $\sigma_R = 0.0766$.

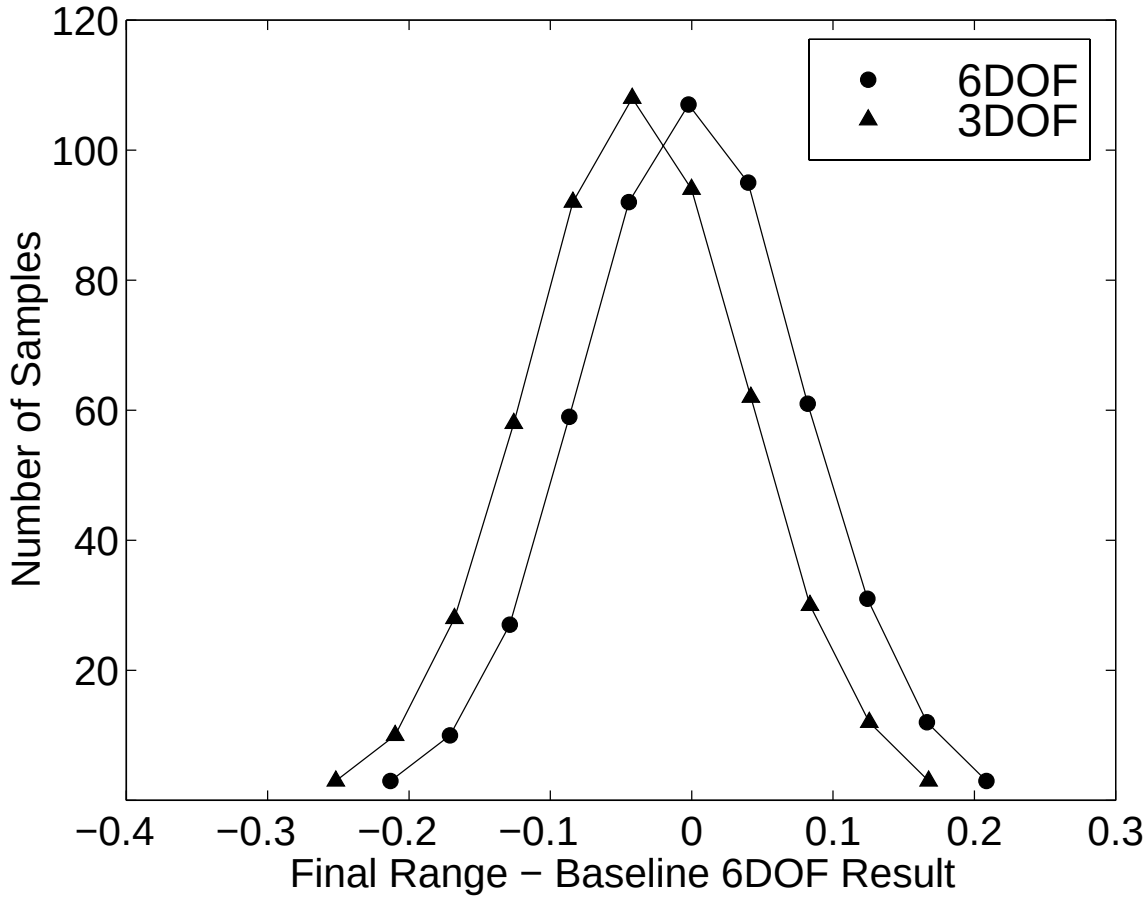

Figure 12
Frequency Data from LHS for Range Offset Due to Initial Weight

When nearly identical frequency data are computed for different fidelity models and computer resources restrict making all computations using the high fidelity model, the following design of computer experiment strategy is commonly used. Runs are initiated with the same random number seed with each model, and the distributions of the ranges of each model are plotted. These runs are compared to determine whether a bias shift in the mean of the distributions has occurred. If a bias does occur, then the lower fidelity model is "calibrated" to eliminate the bias and then used for the hundreds or thousands of runs typically needed to estimate total uncertainty. This same calibration strategy is used in computational simulations of complex processes when experimental measurements are used for the benchmark.

5.2 Effects of Thrust Uncertainty

As we discussed in Sections 4.3 and 4.4, our approach to determining the uncertainty in the trajectories due to uncertain temperature of the rocket motor is to compute bounding trajectories using three thrust profiles: a nominal profile, the highest profile resulting from the highest temperature allowed by the manufacturer, and the lowest profile resulting from the lowest temperature allowed by the manufacturer. To be representative of thrust uncertainty in actual motors, we chose the changes in performance that have been experimentally measured for the

Standard Hawk motor.[95] At the highest allowed temperature of 120°F, the total impulse of the motor is 2% above the nominal performance, but the burn time is decreased by 7%. At the lowest allowed temperature of -20°F, the total impulse of the motor is 2% below the nominal performance, and the burn time is increased by 7%. Stated qualitatively, the high temperature motor has a higher net performance over a shorter burn time, and the cold motor has a lower net performance over a longer burn time.

Figure 13 shows the 6-DOF computed range of the missile for each of the three temperature conditions of the motor as a function of initial weight variability. It can be seen from Fig. 13 that, as expected, the motor temperature uncertainty produces a shift in range: the high temperature motor flying 0.625 nm further than the nominal motor temperature, and the cold motor flying 0.616 nm shorter than the nominal motor. The linearity of the missile range as a function of weight continues to hold for both the high and low motor temperature cases. It is also seen that the uncertainty in range due to motor temperature uncertainty is significantly larger than that observed due to weight variability. The uncertainty in range due to uncertain rocket motor temperature is 1.24 nm. The uncertainty in range due to mass variability can be calculated as $4\sigma_R = 4 \times 0.077 = 0.308$ nm, which is only 25% of the uncertainty due to thrust.

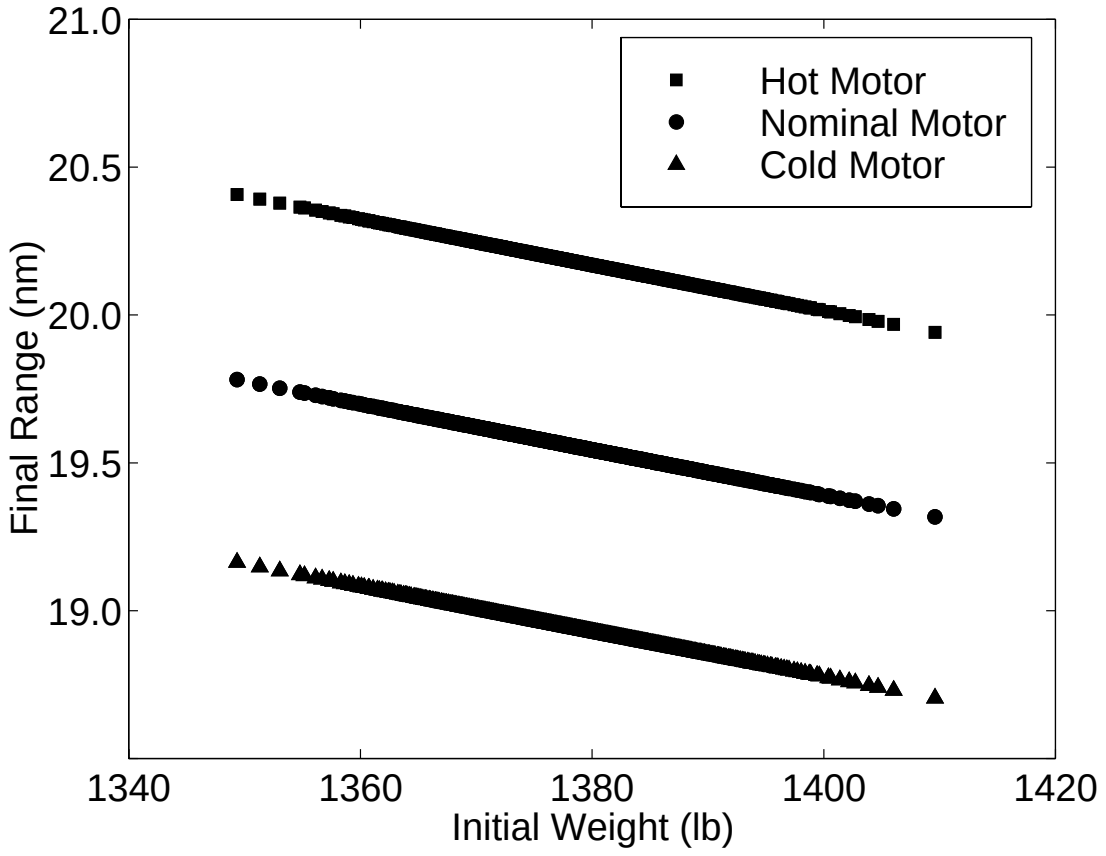

Figure 13
Uncertainty in Range due to Thrust Uncertainty and Mass Variability for 6-DOF Model

Figure 14 shows the frequency data from the LHS for the 6-DOF model as a function of missile range for each of the three motor temperatures. The mean range for the cold motor is shifted 0.62 nm toward shorter range, whereas the hot motor is shifted the same amount toward longer range. The standard deviation in range for the hot and cold motors are nearly identical: $(\sigma_R)_{hot} = 0.0773$ nm and $(\sigma_R)_{cold} = 0.0762$ nm. Recall these are essentially the same as the value of the nominal motor, $(\sigma_R)_{nom} = 0.0770$ nm. The results for the hot and cold motors using the 3-DOF model are very similar to the 6-DOF results presented in this section. The only difference is that the 3-DOF results show the 0.040 nm bias in range, as discussed in the previous section.

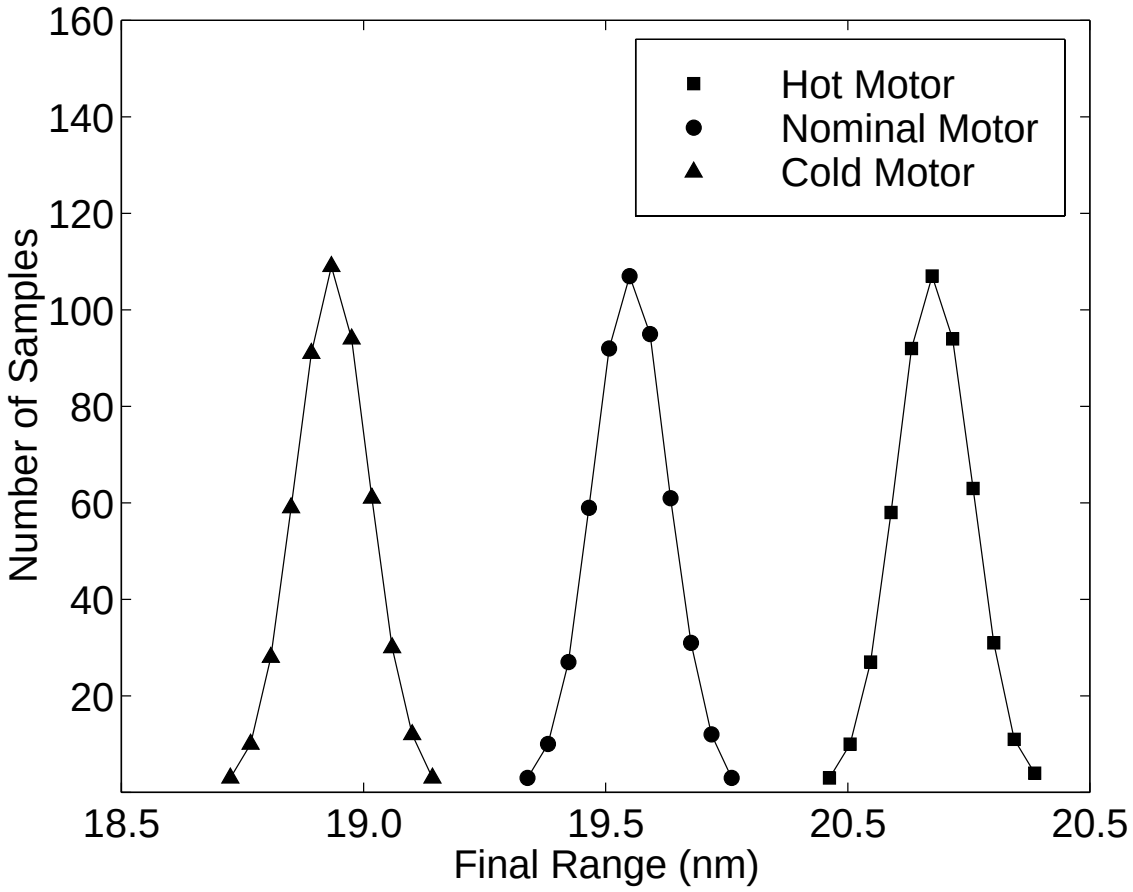

Figure 14
Frequency Data from LHS for Range Uncertainty due to Thrust Uncertainty for 6-DOF Model

We argue that the source of the potentially large uncertainty in missile performance due to motor temperature uncertainty should be characterized as lack of knowledge. Some would argue that the motor temperature uncertainty could be characterized as a variability instead of an uncertainty. The argument is that a probability distribution could be constructed based on experimentally measuring motor temperatures for a large number of actual missile deployments. The variability of motor temperature could then be represented by a probability distribution with some mean and standard deviation. Although this is a reasonable approach, we argue that the variability approach could lead to misleading estimates of system performance for certain

deployment situations. For example, if the deployment was in Alaska during the winter versus Saudi Arabia during the summer, the average range of the missile would be of little value. Additional knowledge of the type of deployment could change the representation. A deployment at a permanent installation with significant environmentally controlled space would be quite different than a makeshift battlefield deployment. As more and different kinds of knowledge are introduced into the analysis, representations other than probability distributions may be more appropriate, e.g., fuzzy sets, belief functions, and possibility theory. Guidance on developing these representations based on available information is not as well developed as probability theory.

5.3 Effects of Numerical Integration Error

As discussed in Sections 3.4 and 3.6, we are able to precisely control the numerical solution error at each step of the numerical integration of the ODEs. The per step, relative, truncation error is estimated using the Runge-Kutta-Fehlberg 4(5) method, and the time step is adjusted at each step so that the truncation error is less than the specified error criterion. Figure 15 shows the computed range of the missile for the 6-DOF model using the nominal thrust profile as a function of the mass variability for five different per step, relative error criteria. There is no effect on calculated range even though the error criterion is varied over eight orders of magnitude: up to 10% error per step. This result was not expected. Intuition leads us to believe that as the error criterion increased greatly, the accuracy of the solution would degrade. For certain state variables, like those that are periodic, the solution accuracy degrades only slightly. Most variables, including output variables that are derived from state variables, like range, do not degrade because the error criterion must be satisfied by all 12 state variables. The state variables that have the highest frequency are those that will restrict the growth of the time step and the resulting growth in solution error. The highest frequency state variables are the pitch rate, q, and the yaw rate, r. Both p and q have a frequency of 1 to 2 Hz, which limits the maximum time step to 0.1 to 0.2 sec. so that this element of physics can be adequately computed. All lower frequency state variables are computed much more accurately than required by the error criterion.

Figure 16 shows the 3-DOF computed range using the nominal thrust profile as a function of mass variability for five per step, relative, truncation error criteria. These five error criteria are the same as those used in the 6-DOF calculation illustrated in Fig. 15. The 3-DOF model has a completely different sensitivity to numerical solution error as compared to the high fidelity model. For a relative error of 10^{-4}, a slight roughness in the range as a function of weight can be seen. For a 10^{-3} error, the amplitude in roughness of range increases to 0.035 nm. This variation in amplitude can occur over a very small change in weight. For example, near the mean weight of 1379 lb, a jump of 0.035 nm can be seen over a change in weight of less than one-tenth of a pound. This type of predicted system response roughness due to solution error has been seen by many investigators, particularly those using first order response surface methods and those using optimization methods that rely on numerical differentiation of the system response.

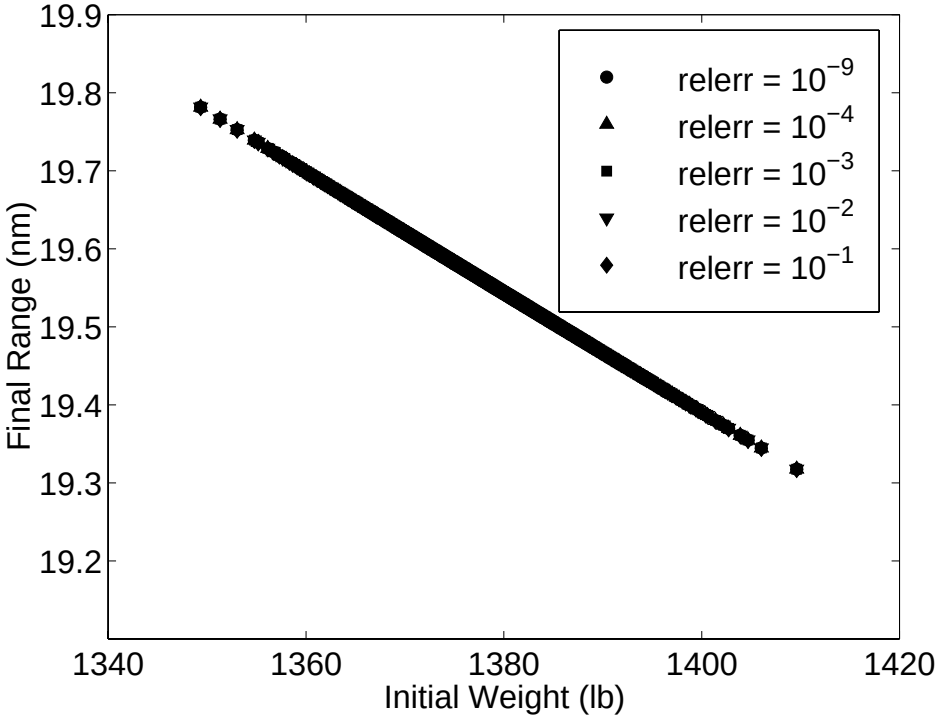

Figure 15: Uncertainty in Range due to Solution Error and Mass Variability for 6-DOF Model

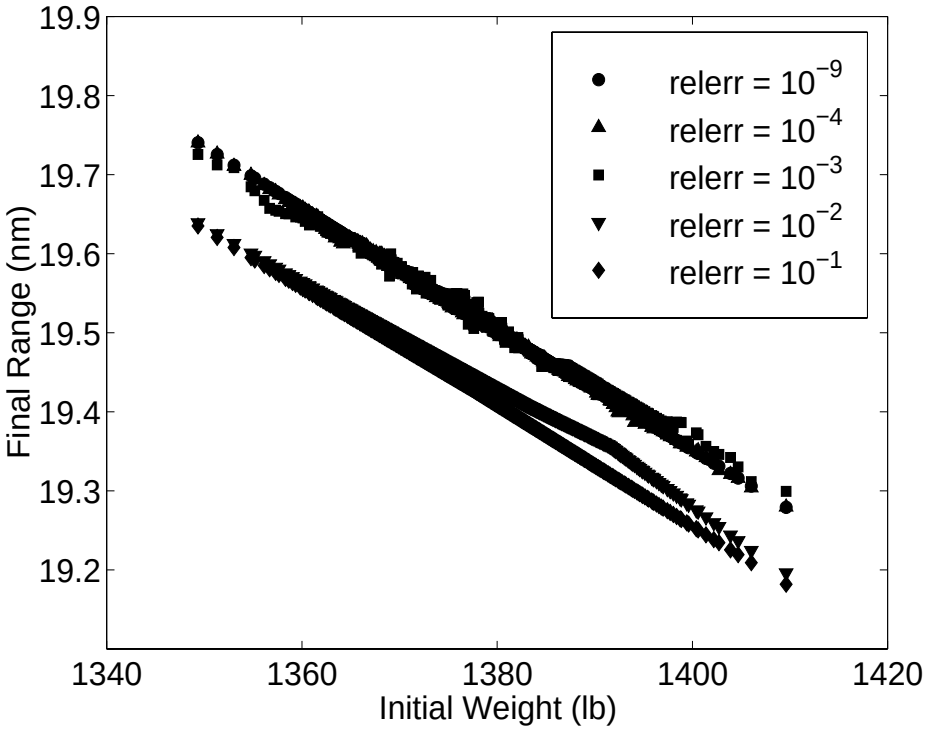

Figure 16: Uncertainty in Range due to Solution Error and Mass Variability for 3-DOF Model

As the numerical error is increased further, to 10^{-2} and 10^{-1}, Fig. 16 shows that a drop in the predicted range occurs. This introduces a bias error in range similar to that observed in the earlier comparison of the 3-DOF and 6-DOF models. The bias error varies slightly with weight for 10^{-2} error but becomes constant at a value of 0.10 nm for 10^{-1} error. In addition, the range becomes an extraordinarily smooth function of weight, with the same characteristic occurring at errors of 10^{-5} and smaller. To understand these unusual characteristics due to solution error, one must examine how the integration step size is changing to control the per step error in the state variables of the 3-DOF model. Contrary to the 6-DOF model, there are *no* periodic state variables in the 3-DOF system. As a result, the step size can increase rapidly from the fixed initial value of 0.1 sec., i.e., all solutions presented in this report attempt to use $\Delta t = 0.1$ sec. in stepping from $t = 0$. If the step size results in an estimated truncation error that satisfies the error criterion, then the step is taken. If the estimated error is 0.6 of the error criterion, then the time step is increased for the next step. If it does not meet the error criterion, then the time step is decreased until the error criterion is met. For the 3-DOF model, the time step increases rapidly because all of the state variables are extremely smooth as a function of time, relative to the 6-DOF model. When the error criterion is changed from 10^{-4} to 10^{-3}, Fig. 16, there is a rapid loss in accuracy of the major physical characteristic of the 3-DOF trajectory: the motor thrust profile. From the initial condition until 4.5 sec. the motor thrust is roughly 19,000 lb. Then it rapidly drops to a sustained thrust value of about 3,600 lb. for 20 sec., after which thrust terminates. For error criteria less than 10^{-5}, the numerical solution very accurately captures these two rapid drops in thrust. As the error criteria increases up to 10^{-3}, the numerical error becomes more erratic, depending on how the time steps fall with regard to the two rapid drops in thrust. For error criteria of 10^{-2} up to 10^{-1}, the error requirement becomes so loose that the time steps jump across the rapid drops in thrust with little notice.

5.4 Effects of Variability, Uncertainty, and Error

This section discusses the computational results for the combination of the mass variability, thrust uncertainty, and solution error for both the 6-DOF and 3-DOF models. Shown in Fig. 17 is the 6-DOF computed range as a function of mass variability, for all three thrust profiles, for the complete range of numerical solution error. As was seen in Figs. 13 and 15, the dominant characteristic is the very smooth variation in range as a function of initial weight for all three thrust profiles, regardless of the numerical solution error. The high temperature and low temperature motor cases are just as insensitive to solution error as the nominal motor temperature case shown in Fig. 15. The frequency plots, although not shown here, also show essentially no effect of solution error. For example, for the cold motor for a 10^{-9} and 10^{-1} relative error the mean range and standard deviation are, respectively: $R = 20.1771$ and $\sigma_R = 0.0773$ and $R = 20.1766$ and $\sigma_R = 0.0774$. As discussed earlier, the higher fidelity model is remarkably insensitive to solution error because of the temporal fine scale structure controlling the time step.

Figure 18 shows the 3-DOF computed range as a function of mass variability for a hot motor for the complete range of solution errors. As the solution error increases, the range calculation becomes even more erratic than that seen earlier for the 3-DOF model in Fig. 16. For an error criterion of 10^{-3} and for weight samples in the range of 1349 to 1355 lb, a nearly constant bias of 0.07 nm toward shorter range occurs. For weights higher than 1355 lb, the computed range wanders back to near the correct value. At a weight of 1399 lb, a discontinuous drop of 0.08 nm in range occurs. When the error criterion increases to 10^{-2}, the bias switches to longer ranges for

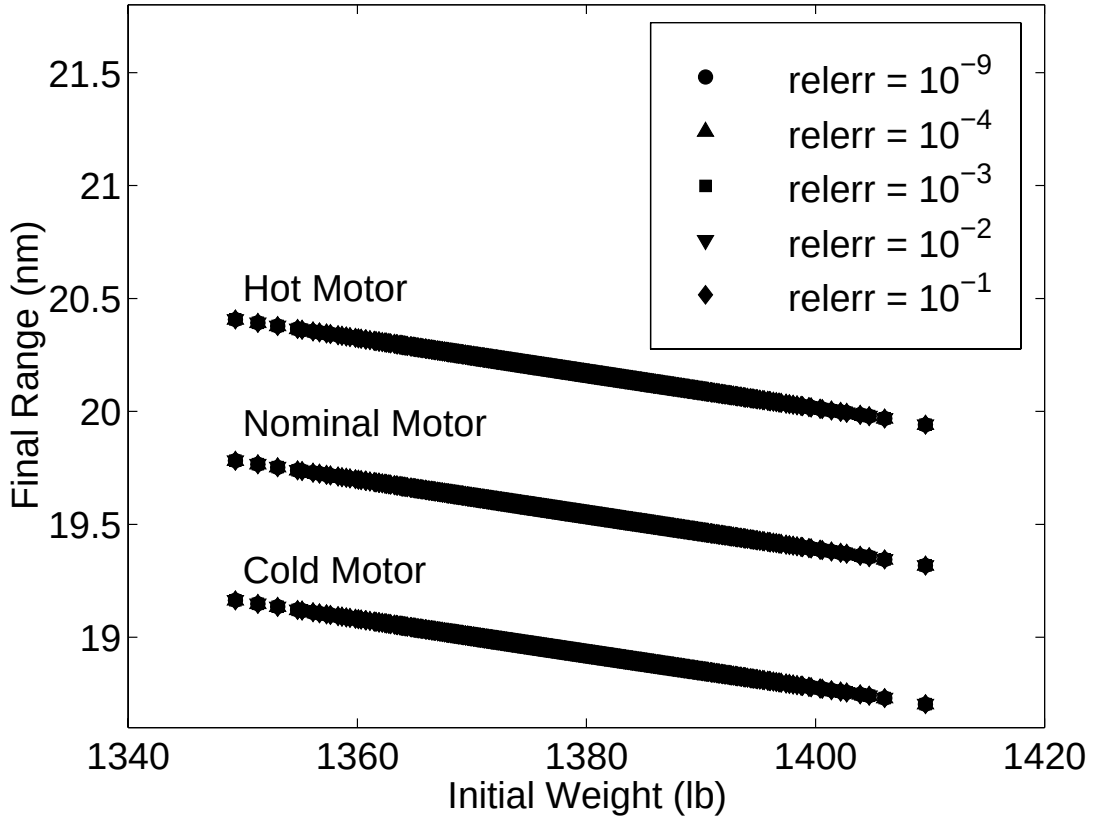

Figure 17
Uncertainty in Range due to Mass Variability, Thrust Uncertainty, and Solution Error for 6-DOF Model

all weights. When the error criterion increases to 10^{-1}, the bias error switches back to shorter ranges for all weights. The frequency plots for this hot motor case, although not shown here, also show erratic behavior for error criteria greater than 10^{-5}. That is, the frequency plots for range show even more sensitivity to solution error than the plot of range as a function of weight. The computed characteristics of the cold motor case are similar to those for the hot motor case.

5.5 Summary Comments

Probably the most surprising computational results obtained in this example problem are those related to the aggregation and interaction of numerical solution error with variability and uncertainty. The counter intuitive result that the higher fidelity model is much less sensitive to solution error than the lower fidelity model needs further comment. The discussion given earlier for the controlling factor in solution error for each model provides an explanation to why this surprising result occurs. These results have implications for the effect of numerical solution error on uncertainty analyses when the mathematical model equations are given by PDEs. The per-step numerical solution error in the present work was precisely controlled by the adaptive step-size control of the ODE integrator. This level of solution error control and robustness does not presently exist in the numerical solution of PDEs. Even if one only considers elliptic boundary value problems, robust adaptive grid generation for the control of local spatial discretization error

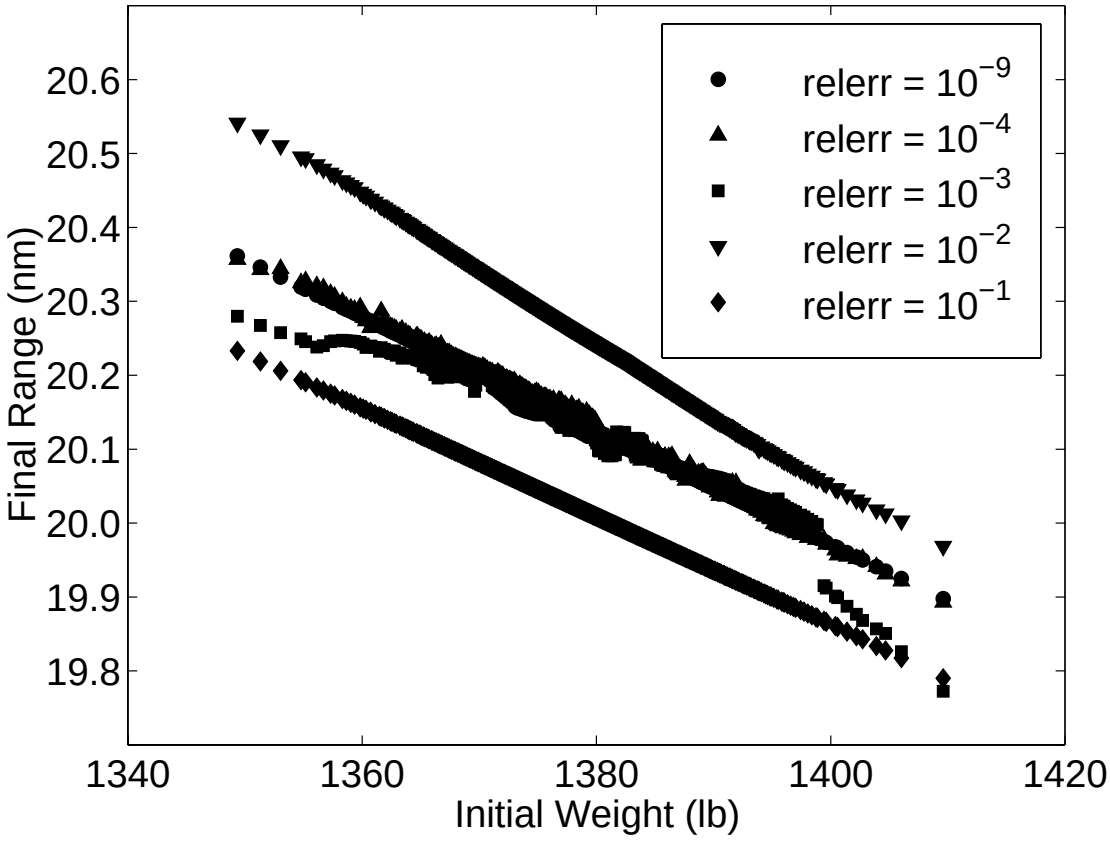

Figure 18
Uncertainty in Range due to Mass Variability and Solution Error
for the High Temperature Motor for 3-DOF Model

does not presently exist. For certain special cases, such as linear boundary value problems or problems with no large gradients, reliable methods for adaptive grid control do exist.

It is our view that the present results for the widely different sensitivity of each mathematical model to solution error would only apply to the numerical solution of PDEs with robust, adaptive, grid generation methods. If one were to use non-adaptive grid generation methods for the solution of the PDEs, very different sensitivities would occur than those observed here. Non-adaptive grid methods would be analogous to a constant time step method in the solution of ODEs. For the present example problem we computed numerical solutions using a constant time step over the length of the trajectory for the 6-DOF and 3-DOF models. Table 1 shows the numerical error in range for various constant time steps for both models using the nominal weight and nominal thrust. As the time step increases, the numerical error for both models increases, but the 6-DOF model error increases more rapidly. When the time step becomes roughly half of the period of the finest scale structure in the 6-DOF model, the error increases exponentially. For a time step of 0.1, the error in the 6-DOF solution has become so large that the trajectory is no longer computable. For the 3-DOF model, the same time steps cause a gradual increase in the solution error. This table shows the opposite sensitivity to numerical error as compared to the adaptive time step method.

Time Step	0.001	0.01	0.07	0.09	0.1
6-DOF	0.0000	0.004	0.038	0.397	∞
3-DOF	0.0000	0.004	0.040	0.052	0.058

Table 1
Error in Range for 6-DOF and 3-DOF for Constant Time Steps

6. Summary and Conclusions

We have presented a comprehensive, new framework for modeling and simulation that blends the perspective of three technical communities; the systems view from the operations research community, propagation of uncertainty from the risk assessment community, and the numerical solution of partial differential equations from the computational physics community. The activities that are conducted in each of the six phases of modeling and simulation are discussed in detail. Consistent with recent work in the risk assessment community, we carefully define and distinguish between variability and uncertainty. In addition, we delineate and discuss acknowledged and unacknowledged errors. In each of the activities in each phase of modeling and simulation we discuss which type of source (variability, uncertainty, or error) typically dominates the activity. Particular emphasis is given to distinguishing the continuous and discrete mathematical modeling activities, and to the nondeterministic features of the analysis. The emphasis includes discussions of the representations of variability and uncertainty, and their propagation methods. Our framework applies regardless of whether the discretization procedure for solving the partial differential equations is based on finite elements, finite volumes, or finite differences.

The formal distinction between variability and uncertainty in this framework drives one toward different mathematical representations for variability and uncertainty; probabilistic representations for variability and various other modern information theories for representation of uncertainty. One approach that has been used for uncertainty is Bayesian probability. This approach takes a subjective view of probability as a measure of degree of belief in a hypothesis. Although we believe this is a step in the right direction to represent uncertainty, we do not believe it is satisfactory. We recommend research into modern theories of uncertainty-based information; such as evidence (Dempster/Shafer) theory, possibility theory, fuzzy set theory, and imprecise probability theory. These theories, however, are not well developed when compared to probabilistic inference. In addition, none of these theories, except fuzzy set theory, have been applied to engineering analysis problems. If one were to take the step and represent variability probabilistically and uncertainty with a non-probabilistic theory, then one must face the question of propagating these concurrently in the modeling and simulation process; they are not combinable. Propagation methods of this type are even more of a research topic.

Two example analyses were provided to demonstrate the flexibility and wide range of applicability of this framework and to formally detail some of the technical aspects of the approach. The weapon in a fire example provided a listing of possible sources of variability, uncertainty and error that might contribute to total uncertainty in the first three modeling phases of the analysis. The second example, involving missile flight dynamics, focused on a more limited set of sources, but

provided specific details of how these sources might be accommodated using this framework. System responses were calculated to illustrate how these components of total modeling uncertainty could be estimated and propagated.

We believe the usefulness of the present framework results from two aspects. First, it formalizes and merges a broad range of activities conducted in complex system modeling and modern computational simulation. It collects into one picture all of the activities so that each one can be clearly distinguished, relationships can be unambiguously depicted, and assumptions can be formalized. The framework can be viewed as a many-branched event and decision tree and, as such, the connection and propagation of scenarios, nondeterminism, and modeling decisions and assumptions are unequivocal. Second, it identifies how and where all sources of variability, uncertainty, and error contribute to the total uncertainty in the modeling and simulation process. For the analysis of complex systems, this formal recognition of sources of nondeterminism and error shows the compounding effect and rapid growth of each source through the modeling and simulation process. Some have referred to this growth of total uncertainty through the modeling and simulation process as “overwhelming.” However, it is an issue that must be faced by analysts, and decision makers who use the results of modeling and simulation.

References

1. Hora, S. C., and Iman, R. L., "Expert Opinion in Risk Analysis: The NUREG-1150 Methodology," *Nuclear Science and Engineering*, Vol. 102, 1989, pp. 323-331.
2. Morgan, M. G., and Henrion, M., *Uncertainty: A guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*, 1st ed., Cambridge University Press, New York, 1990.
3. Boyack, B. E., Catton, I., Duffey, R. B., Griffith, P., Katsma, K. R., Lellouche, G. S., Levy, S., Rohatgi, U. S., Wilson, G. E., Wulff, W., and Zuber, N., "Quantifying Reactor Safety Margins Part 1: An Overview of the Code Scaling, Applicability, and Uncertainty Evaluation Methodology," *Nuclear Engineering and Design*, Vol. 119, 1990, pp. 1-15.
4. Breeding, R. J., Helton, J. C., Gorham, E. D., and Harper, F. T., "Summary Description of the Methods used in the Probabilistic Risk Assessments for NUREG-1150," *Nuclear Engineering and Design*, Vol. 135, 1992, pp. 1-27.
5. Hauptmanns, U., and Werner, W., *Engineering Risks Evaluation and Valuation*, First Edition ed., Springer-Verlag, 1991.
6. Bergeron, K. D., Siezak, S. E., and Leach, C. E., "Proposed Deterministic Severe Accident Criteria for the Heavy Water Reactor-New Production Reactor Containment," *Nuclear Safety*, Vol. 34, No. 1, 1993, pp. 20-32.
7. Modarres, M., *What Every Engineer should know about Reliability and Risk Analysis*, Marcel Dekker, Inc., 1993.
8. Helton, J. C., "Treatment of Uncertainty in Performance Assessments for Complex Systems," *Risk Analysis*, Vol. 14, No. 4, 1994, pp. 483-511.
9. Kumamoto, H., and Henley, E. J., *Probabilistic Risk Assessment and Management for Engineers and Scientists*, 2nd ed., IEEE Press, New York, NY, 1996.
10. Ang, A. H. S., and Tang, W. H., *Probability Concepts in Engineering Planning and Design: Vol. I Basic Principles*, 1st ed., John Wiley & Sons, New York, NY, 1975.
11. Ditlevsen, O., *Uncertainty Modeling with Applications to Multidimensional Civil Engineering Systems*, 1st ed., McGraw-Hill, Inc., New York, 1981.
12. Ayyub, B. M., Gupta, M. M., and Kanal, L. N., eds. *Analysis and Management of Uncertainty: Theory and Applications*, 1st ed., North-Holland, New York, 1992.
13. Ayyub, B. M., and Gupta, M. M., eds. *Uncertainty Modelling and Analysis: Theory and Applications*, 1st ed., North-Holland, New York, 1994.
14. Downing, D. J., Gardner, R. H., and Hoffman, F. O., "An Examination of Response-Surface Methodologies for Uncertainty Analysis in Assessment Models," *Technometrics*, Vol. 27, No. 2, 1985, pp. 151-163.
15. Beck, M. B., "Water Quality Modeling: A Review of the Analysis of Uncertainty," *Water Resources Research*, Vol. 23, No. 8, 1987, pp. 1393-1442.
16. Bogen, K. T., and Spear, R. C., "Integrating Uncertainty and Interindividual Variability in Environmental Risk Assessment," *Risk Analysis*, Vol. 7, No. 4, 1987, pp. 427-436.
17. Helton, J. C., Garner, J. W., and M. G. Marietta, e. a., "Uncertainty and Sensitivity Analysis Results Obtained in a Preliminary Performance Assessment for the Waste Isolation Pilot Plant," *Nuclear Science and Engineering*, Vol. 114, 1993, pp. 286-331.
18. Ballin, P. R., Aziz, K., Journel, A. G., and Zuccolo, L., "Quantifying the Impact of Geological Uncertainty on Reservoir Performing Forecasts," Society of Petroleum Engineers, *12th Symposium on Reservoir Simulation*, New Orleans, LA, 1993.

19. Frank, M. V., "Treatment of Uncertainties in Space Nuclear Risk Assessment with Examples from Cassini Mission Applications," *Reliability Engineering and System Safety*, Vol. 66, 1999, pp. 203-221.
20. Helton, J. C., Anderson, D. R., Jow, H.-N., Marietta, M. G., and Basabilvazo, G., "Performance Assessment in Support of the 1996 Compliance Certification Application for the Waste Isolation Pilot Plant," *Risk Analysis*, Vol. 19, No. 5, 1999, pp. 959-986.
21. Vamos, T., "Epistemic Background Problems of Uncertainty," IEEE Computer Society Press, *First International Symposium on Uncertainty Modeling and Analysis*, College Park, MD, 1990.
22. Hoffman, F. O., and Hammonds, J. S., "Propagation of Uncertainty in Risk Assessments: The Need to Distinguish Between Uncertainty Due to Lack of Knowledge and Uncertainty Due to Variability," *Risk Analysis*, Vol. 14, No. 5, 1994, pp. 707-712.
23. Rowe, W. D., "Understanding Uncertainty," *Risk Analysis*, Vol. 14, No. 5, 1994, pp. 743-750.
24. Hora, S. C., "Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management," *Reliability Engineering and System Safety*, Vol. 54, 1996, pp. 217-223.
25. Frey, H. C., and Rhodes, D. S., "Characterizing, Simulating, and Analyzing Variability and Uncertainty: An Illustration of Methods Using an Air Toxics Emissions Example," *Human and Ecological Risk Assessment*, Vol. 2, No. 4, 1996, pp. 762-797.
26. Ferson, S., and Ginzburg, L. R., "Different methods are needed to propagate ignorance and variability," *Reliability Engineering and System Safety*, Vol. 54, 1996, pp. 133-144.
27. NCRP, "A Guide for Uncertainty Analysis in Dose and Risk Assessments Related to Environmental Contamination," National Council on Radiation Protection and Measurements, NCRP Commentary No. 14, Bethesda, MD, 1996.
28. Johnson, C., *Numerical Solution of Partial Differential Equations by the Finite Element Method*, Cambridge University Press, New York, 1987.
29. Ames, W. F., *Numerical Methods for Partial Differential Equations*, Academic Press, Boston, MA, 1992.
30. Morton, K. W., and Mayers, D. F., *Numerical Solution of Partial Differential Equations*, Cambridge University Press, New York, 1994.
31. Zeigler, B. P., *Multifaceted Modelling and Discrete Event Simulation*, 1st ed., Academic Press, Orlando, 1984.
32. Neelamkavil, F., *Computer Simulation and Modelling*, 1st ed., John Wiley & Sons, New York, 1987.
33. Law, A. M., and Kelton, W. D., *Simulation Modeling and Analysis*, 2nd ed., McGraw-Hill, New York, 1991.
34. Bossel, H., *Modeling and Simulation*, 1st ed., A. K. Peters, Ltd., Wellesley, MA, 1994.
35. Schlesinger, S., "Terminology for Model Credibility," *Simulation*, Vol. 32, No. 3, 1979, pp. 103-104.
36. Jacoby, S. L. S., and Kowalik, J. S., *Mathematical Modeling with Computers*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1980.
37. Sargent, R. G., "Simulation Model Validation," in *Simulation and Model-Based Methodologies: An Integrative View*, T.I. Oren, B.P. Zeigler, and M.S. Elzas Eds. , Springer-Verlag Berlin Heidelberg, Syracuse, 1984, pp. 537-555.
38. Sargent, R. G., "An Expository on Verification and Validation of Simulation Models," *1985 Winter Simulation Conference*, Sacramento, CA, 1985.

39. Nance, R. E., "Model Representation in Discrete Event Simulation: The Conical Methodology," Virginia Polytechnic Inst. and State University, Dept. of Computer Science, Technical Rept. CS81003-R, 1981.
40. Balci, O., "Guidelines for Successful Simulation Studies," *Proceedings of the 1990 Winter Simulation Conf.*, New Orleans, LA, 1990.
41. Ang, A. H. S., and Tang, W. H., *Probability Concepts in Engineering Planning and Design: Vol. II Decision, Risk, and Reliability*, John Wiley & Sons, New York, NY, 1984.
42. Davis, P. A., Price, L. L., Wahi, K. K., Goodrich, M. T., Gallegos, D. P., Bonano, E. J., and Guzowski, R. V., "Components of an Overall Performance Assessment Methodology," Sandia National Laboratories, NUREG/CR-5256; SAND88-3020, Albuquerque, NM, 1990.
43. Davis, P. A., Bonano, E. J., Wahi, K. K., and Price, L. L., "Uncertainties Associated with Performance Assessment of High-Level Radioactive Waste Repositories," Sandia National Laboratories, NUREG/CR-5211; SAND88-2703, Albuquerque, NM, 1990.
44. Giere, R. N., "Knowledge, Values, and Technological Decisions: A Decision Theoretic Approach," in *Acceptable Evidence: Science and Values in Risk Management*, D.G. Mayo and R.D. Hollander Eds. , Oxford University Press, New York, 1991, pp. 183-203.
45. Apostolakis, G., "A Commentary on Model Uncertainty," U.S. Nuclear Regulatory Commission, *Proceedings of Workshop I in Advanced Topics in Risk and Reliability Analysis —- Model Uncertainty: Its Characterization and Quantification*, Annapolis, MD, 1994.
46. Haimes, Y. Y., Barry, T., and Lambert, J. H., "When and How Can You Specify a Probability Distribution When You Don't Know Much?," *Risk Analysis*, Vol. 14, No. 5, 1994, pp. 661-706.
47. Ayyub, B. M., "The Nature of Uncertainty in Structural Engineering," in *Uncertainty Modelling and Analysis: Theory and Applications*, B.M. Ayyub and M.M. Gupta Eds. , Elsevier, New York, 1994, pp. 195-210.
48. Ferson, S., "What Monte Carlo Methods Cannot Do," *Human and Ecological Risk Assessment*, Vol. 2, No. 4, 1996, pp. 990-1007.
49. Helton, J. C., and Burmaster, D. E., "Guest editorial: treatment of aleatory and epistemic uncertainty in performance assessments for complex systems," *Reliability Engineering and System Safety*, Vol. 54, 1996, pp. 91-94.
50. Rai, S. N., Krewski, D., and Bartlett, S., "A General Framework for the Analysis of Uncertainty and Variability in Risk Assessment," *Human and Ecological Risk Assessment*, Vol. 2, No. 4, 1996, pp. 972-989.
51. Parry, G. W., "The characterization of uncertainty in Probabilistic Risk Assessments of complex systems," *Reliability Engineering and System Safety*, Vol. 54, 1996, pp. 119-126.
52. Paté-Cornell, M. E., "Uncertainties in risk analysis: Six levels of treatment," *Reliability Engineering and System Safety*, Vol. 54, 1996, pp. 95-111.
53. Ayyub, B. M., ed. *Uncertainty Modeling and Analysis in Civil Engineering*, CRC Press, Boca Raton, FL, 1998.
54. Cullen, A. C., and Frey, H. C., *Probabilistic Techniques in Exposure Assessment: A Handbook for Dealing with Variability and Uncertainty in Models and Inputs*, Plenum Press, New York, 1999.
55. Klir, G. J., and Folger, T. A., *Fuzzy Sets, Uncertainty, and Information*, 1st ed., Prentice Hall, Englewood Cliffs, NJ, 1988.
56. Smithson, M., *Ignorance and Uncertainty: Emerging Paradigms*, Springer-Verlag, New York, 1989.
57. Walley, P., *Statistical Reasoning with Imprecise Probabilities*, Chapman and Hall, London,

1991.
58. Guan, J., and Bell, D. A., *Evidence Theory and Its Applications*, Vol. I, Elsevier Science Publishers B.V., Amsterdam, The Netherlands, 1991.
59. Krause, P., and Clark, D., *Representing Uncertain Knowledge: An Artificial Intelligence Approach*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1993.
60. Ross, T. J., *Fuzzy Logic with Engineering Applications*, McGraw-Hill, Inc., New York, NY, 1995.
61. Klir, G. J., and Wierman, M. J., *Uncertainty-Based Information: Elements of Generalized Information Theory*, Vol. 15, Physica-Verlag, Heidelberg, 1998.
62. Cox, E., *The Fuzzy Systems Handbook: A Practitioner's Guide to Building, Using, and Maintaining Fuzzy Systems*, 2nd ed., AP Professional, Division of Academic Press, San Diego, CA, 1999.
63. Cox, E., *The Fuzzy Systems Handbook*, AP Professional, New York, NY, 1998.
64. Almond, R. G., *Graphical Belief Modeling*, 1st ed., Chapman & Hall, London, U. K., 1995.
65. Dubois, D., and Prade, H., *Possibility Theory: An Approach to Computerized Processing of Uncertainty*, Plenum Press, New York, 1986.
66. de Cooman, G., Ruan, D., and Kerre, E. E., eds. *Foundations and Applications of Possibility Theory*, World Scientific Publishing Co., Singapore, 1995.
67. Kozine, I., "Imprecise Probabilities Relating to Prior Reliability Assessments," *1st International Symposium on Imprecise Probabilities and Their Applications*, Ghent, Belgium, 1999.
68. Coleman, H. W., and Steele, W. G., Jr., *Experimentation and Uncertainty Analysis for Engineers*, 1st ed., John Wiley & Sons, New York, 1989.
69. Oberkampf, W. L., Diegert, K. V., Alvin, K. F., and Rutherford, B. M., "Variability, Uncertainty, and Error in Computational Simulations," ASME, ASME-HTD-Vol. 357-2, *AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, Albuquerque, NM, 1998.
70. Oberkampf, W. L., DeLand, S. M., Rutherford, B. M., Diegert, K. V., and Alvin, K. F., "A New Methodology for the Estimation of Total Uncertainty in Computational Simulation," American Institute of Aeronautics and Astronautics, AIAA Paper No. 99-1612, *AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, St. Louis, MO, 1999.
71. Box, G. E. P., "Sampling and Bayes' Inference in Scientific Modeling and Robustness," *Journal Statist. Soc. A*, Vol. 143, No. A, 1980, pp. 383-430.
72. Oberkampf, W. L., and Blottner, F. G., "Issues in Computational Fluid Dynamics Code Verification and Validation," *AIAA Journal*, Vol. 36, No. 5, 1998, pp. 687-695.
73. Lewis, R. O., *Independent Verification and Validation*, 1st ed., John Wiley & Sons, Inc., New York, 1992.
74. Knepell, P. L., and Arangno, D. C., *Simulation Validation: A Confidence Assessment Methodology*, 1st ed., IEEE Computer Society Press, Washington, 1993.
75. Yee, H. C., and Sweby, P. K., "Global Asymptotic Behavior of Iterative Implicit Schemes," *International Journal of Bifurcation and Chaos*, Vol. 4, No. 6, 1994, pp. 1579-1611.
76. Yee, H. C., and Sweby, P. K., "Dynamical Approach Study of Spurious Steady-State Numerical Solutions of Nonlinear Differential Equations II. Global Asymptotic Behavior of Time Discretizations," *Computational Fluid Dynamics*, Vol. 4, 1995, pp. 219-283.
77. Yee, H. C., Torczynski, J. R., Morton, S. A., Visbal, M. R., and Sweby, P. K., "On Spurious Behavior of CFD Simulations," American Institute of Aeronautics and Astronautics, AIAA Paper No. 97-1869, *13th AIAA Computational Fluid Dynamics Conference*,

Snowmass, CO, 1997.
78. McKay, M. D., Beckman, R. J., and Conover, W. J., "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, Vol. 21, No. 2, 1979, pp. 239-245.
79. Iman, R. L., and Conover, W. J., "A Distribution-Free Approach to Introducing Rank Correlation Among Input Variables," *Communications in Statistics, Simulation and Computation*, Vol. 11, No. 3, 1982, pp. 311-334.
80. Box, E. P., and Draper, N. R., *Emperical Model-Building and Response Surfaces*, Wiley and Sons, New York, NY, 1987.
81. Klir, G. J., "Probabilistic versus possibilistic conceptualization of uncertainty," in *Analysis and Management of Uncertainty: Theory and Applications*, B.M. Ayyub, M.M. Gupta, and L.N. Kanal Eds. , North-Holland, New York, 1992, pp. 13-26.
82. Bier, V. M., "Fuzzy set theory, probability theory, and truth functionality," in *Analysis and Management of Uncertainty: Theory and Applications*, B.M. Ayyub, M.M. Gupta, and L.N. Kanal Eds. , North-Holland, New York, 1992, pp. 65-78.
83. Muhanna, R. L., and Mullen, R. L., "Development of Interval Based Methods for Fuzziness in Continuum Mechanics," IEEE Computer Society, *The Third International Symposium on Uncertainty Modeling and Analysis and Annual Conference of the North American Fuzzy Information Processing Society*, College Park, MD, 1995.
84. Hatton, L., "The T Experiments: Errors in Scientific Software," *IEEE Computational Science & Engineering*, Vol. April-June, 1997, pp. 27-38.
85. Richardson, L. F., "The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam," *Transaction of the Royal Society of London*, Vol. Series A, Vol. 210, 1910, pp. 307-357.
86. Tenner, E., *Why Things Bite Back*, Alfred A. Knopf, New York, NY, 1996.
87. Yee, H. C., and Sweby, P. K., "Aspects of Numerical Uncertainties in Time Marching to Steady-State Numerical Solutions," *AIAA Journal*, Vol. 36, No. 5, 1998, pp. 712-724.
88. Kloeden, P. E., and Platen, E., *Numerical Solution of Stochastic Differential Equations*, Springer-Verlag, New York, 1992.
89. Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical Recipes in FORTRAN*, Cambdrige University Press, New York, NY, 1992.
90. Cash, J. R., and Karp, A. H., "A Variable Order Runge-Kutta Method for Initial-Value Problems with Rapidly Varying Right-Hand Sides," *ACM Transactions on Mathematical Software*, Vol. 16, No. 3, 1990, pp. 201-222.
91. Salguero, D. E., "Trajectory Analysis and Optimization Software (TAOS)," Sandia National Labs., SAND99-0811, Albuquerque, NM, 1999.
92. NOAA, NASA, and USAF, "U. S. Standard Atmosphere, 1976," U. S. Government Printing Office, Washington, DC, 1976.
93. Rollstin, L. R., *personal communication*, 1998.
94. Rollstin, L. R., and Fellerhoff, R. D., "Aeroballistic and Mechanical Design and Development of the Talos-Terrier-Recurit (Tater) Rocket System with Flight Test Results," Sandia National Laboratories, SAND74-0440, Albuquerque, NM, 1976.
95. CPIA, "Rocket Motor Manual," Chemical Propulsion Information Agency, CPIA/M1, Laurel, MD, 1982.
96. DOD, "Department of Defense World Geodetic System 1984," Defense Mapping Agency, WGS 84 Development Committee, DMA TR 8350.2, Washington, DC, 1987.

Appendix A: Flight Dynamics Equations of Motion

A.1 Introduction

The equations of motion for both the 3DOF model and the 6DOF model are derived in this Appendix. Both sets of equations are derived for fixed-mass vehicles. They can be applied to variable mass vehicles (e.g., a rocket-powered missile) by accounting properly for the forces and moments due to the propulsion system and by integrating the mass flow equation along with the vehicle equations of motion in the integration procedure. The derivation of the equations of motion given in this Appendix follows closely the derivation given in the TAOS reference manual.[91]

In order to discuss the equations of motion, we will need to define three coordinate systems: an Earth-centered, Earth-fixed, Cartesian (ECFC) coordinate system, an Earth-centered, inertial, Cartesian (ECIC) coordinate system, and a body-fixed Cartesian coordinate system.

A.2 Coordinate Systems

A.2.1 Earth-Centered, Earth-Fixed, Cartesian (ECFC) Coordinate System

One coordinate system for describing the translational motion of a vehicle over the surface of the Earth is an Earth-centered, Earth-fixed, Cartesian (ECFC) coordinate system. As shown in Figure A-1, the origin for this coordinate system is located at the center of the Earth, with the x- and y- axes in the Equatorial plane and the z-axis pointing through the North Pole. The x-axis is always aligned with the Greenwich meridian ($longitude = 0°$) and the y-axis is always aligned with the $90°$ meridian. Note that the ECFC coordinate system rotates with the Earth about the z-axis.

A.2.2 Earth-Centered, Inertial, Cartesian (ECIC) Coordinate System

Newton's Second Law, which is needed for deriving the translational equations of motion, applies to an inertial reference frame. The ECFC coordinate system defined above rotates with time and so is not an inertial reference frame. Since the trajectories we are concerned with are of relatively short duration, we can neglect the effect of the Earth's rotation about the Sun and define an Earth-centered inertial coordinate system. As shown in Figure A-2, the origin of the ECIC coordinate system is the center of the Earth with the x- and y-axes in the Equatorial plane and the z-axis pointing through the North Pole. The ECIC coordinate system is fixed in space. Thus the ECFC coordinate system defined previously rotates with respect to the ECIC coordinate system at the Earth's rotation rate, ω_E.

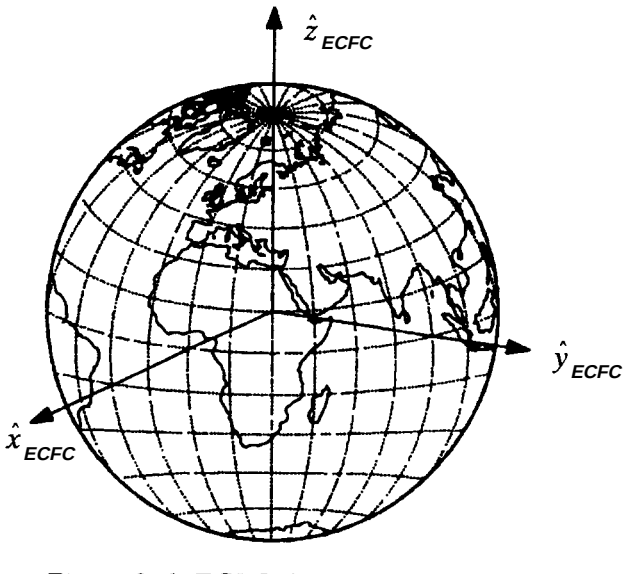

Figure A-1
The Earth-Centered, Earth-Fixed Cartesian (ECFC) Coordinate System[91]

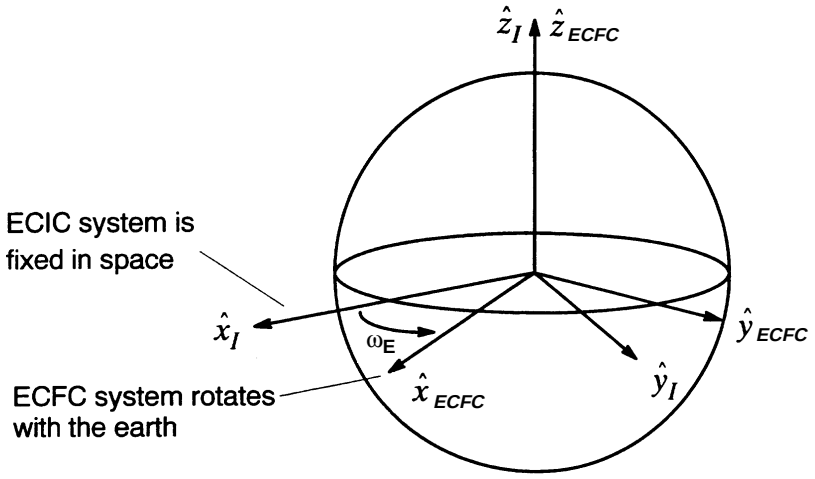

Figure A-2
The Earth-Centered, Inertial Cartesian (ECIC) Coordinate System[91]

A.2.3 Body-Fixed Cartesian Coordinate System

The forces and moments acting on the body and its mass properties are most naturally described in a coordinate system based on the body of the vehicle. The body-fixed, Cartesian coordinate system is shown in Figure A-3a. Note that this coordinate system has its origin at the center of mass with the x-axis oriented toward the front of the vehicle. In a winged or finned vehicle, the y-axis points toward the right wing or right fin (looking in the positive x-axis direction) and the z-axis is oriented downward to form a right-handed coordinate system.

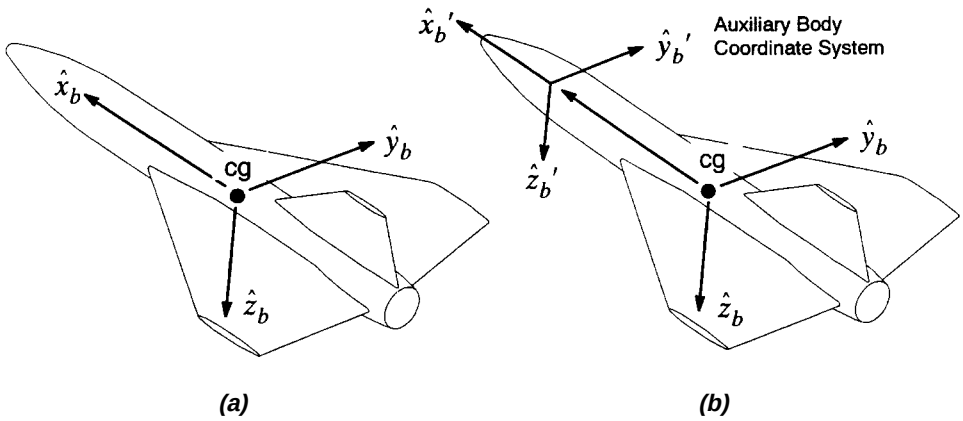

Figure A-3
Body-FixedCoordinates[91]

Since the 3DOF model represents the vehicle as a point which is, by definition, the center of mass, this is the only body coordinate system used in the 3DOF model. The 6DOF model requires information about the location and magnitude of all forces acting on the body as well as the magnitude of all moments on the body. It is sometimes convenient to define an auxiliary body coordinate system to simplify the input files. The auxiliary body coordinate system has the same orientation as the primary body coordinate system, but is offset to some other convenient location. An auxiliary body coordinate system is also shown in Figure A-3b.

A.3 Translational Equations of Motion

The translational equations of motion are derived from Newton's second law applied to a point mass:

$$\sum \vec{F} = m\vec{a}_I \qquad \text{(EQ A-1)}$$

where the $\sum \vec{F}$ represents all the forces acting on the vehicle, m is the mass of the vehicle, and $\vec{a}_I$ is the acceleration in an inertial reference frame. In the 3DOF model, (EQ A-1) describes the motion of the vehicle while in the 6DOF model, it describes the translational motion of the vehicle's center-of-mass.

While Newton's second law applies to inertial coordinate systems, it is convenient for trajectory analysis problems to express the acceleration in a non-inertial earth-fixed coordinate system. Therefore, the inertial acceleration in (EQ A-1) must be transformed to the earth-fixed coordinate system. As noted above, the ECFC coordinate system rotates with respect to the inertial coordinate system (ECIC) at the Earth's rotation rate, $\vec{\omega}_E$. The general relationship for transforming a vector quantity, $\frac{d}{dt}\vec{Q}$, between a fixed reference frame and one rotating at an angular rate, $\vec{\omega}$, with respect to the fixed frame is given by:

$$\left(\frac{d}{dt}\vec{Q}\right)_{fixed} = \left(\frac{d}{dt}\vec{Q}\right)_{rotating} + \vec{\omega} \times \vec{Q}\,. \qquad \text{(EQ A-2)}$$

(EQ A-1) can be rewritten using (EQ A-2) as:

$$\vec{a} = \sum \frac{\vec{F}}{m} - 2(\vec{\omega}_E \times \vec{V}) - \vec{\omega}_E \times (\vec{\omega}_E \times \vec{r}) \qquad \text{(EQ A-3)}$$

where $\vec{r}$ and $\vec{V}$ are the position and velocity of the vehicle, respectively, and all of the vectors are in the ECFC coordinate system.

(EQ A-3) is a second-order differential equation which can be written as a set of coupled first order equations:

$$\dot{\vec{r}} = \vec{V} \qquad \text{(EQ A-4)}$$

$$\dot{\vec{V}} = \sum \frac{\vec{F}}{m} - 2(\vec{\omega}_E \times \vec{V}) - \vec{\omega}_E \times (\vec{\omega}_E \times \vec{r}) \qquad \text{(EQ A-5)}$$

If an initial position, $\vec{r}_o$, and velocity, $\vec{V}_o$, with respect to the Earth are known at $t=t_0$, and if the vehicle's mass and the forces acting on it are known, these equations can be integrated to determine the trajectory of the vehicle. In the case of the unguided Hawk missile studied in this flight dynamics example, the forces are the gravitational force, propulsion, and aerodynamic forces. The assumptions and parameters for these forces during the flight of the missile are given in Appendix C.

A.4 Rotational Equations of Motion

The rotational equations of motion for rigid body (6DOF) dynamics are derived by noting that the rate of change of the angular momentum of the body is equal to the sum of the moments acting on it:

$$\frac{d}{dt}(\vec{L}) = \sum \vec{M} \qquad \text{(EQ A-6)}$$

The angular momentum is given by

$$\vec{L} = [I] \cdot {}_I\vec{\omega}_b \qquad \text{(EQ A-7)}$$

where $[I]$ is the inertia tensor and ${}_I\vec{\omega}_b$ is the angular velocity of the vehicle in the inertial reference frame.

Transforming the derivative in (EQ A-6) to the body-fixed coordinate system using (EQ A-2) and assuming a constant mass, one can obtain:

$$\begin{aligned}\sum \vec{M} = {} & \left[I_{xx}\dot{\omega}_x + I_{xy}\dot{\omega}_y + I_{xz}\dot{\omega}_z + I_{yz}\left(\omega_y^2 - \omega_z^2\right) + (I_{xz}\omega_y - I_{xy}\omega_z)\omega_x + (I_{zz} - I_{yy})\omega_y\omega_z\right]\hat{x}_b + \\ & \left[I_{xy}\dot{\omega}_x + I_{yy}\dot{\omega}_y + I_{yz}\dot{\omega}_z + I_{xz}\left(\omega_z^2 - \omega_x^2\right) + (I_{xy}\omega_z - I_{yz}\omega_x)\omega_y + (I_{xx} - I_{zz})\omega_x\omega_z\right]\hat{y}_b + \\ & \left[I_{xz}\dot{\omega}_x + I_{yz}\dot{\omega}_y + I_{zz}\dot{\omega}_z + I_{xy}\left(\omega_x^2 - \omega_y^2\right) + (I_{yz}\omega_x - I_{xz}\omega_y)\omega_z + (I_{yy} - I_{xx})\omega_x\omega_y\right]\hat{z}_b\end{aligned} \qquad \text{(EQ A-8)}$$

This equation relates the total moment acting on the vehicle to the angular acceleration of the vehicle in the body-fixed coordinate system. The total moment acting on the vehicle is composed of the aerodynamic moment and the propulsive moment. Details of the assumptions and parameters for computing these moments as a function of flight conditions are given in Appendix C.

(EQ A-8) is a set of coupled, first-order differential equations which are linear in $\dot{\omega}_x$, $\dot{\omega}_y$, and $\dot{\omega}_z$. Therefore this equation can be rewritten in the standard form $[A]X = B$ as follows:

$$\begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \sum \vec{M} \cdot \hat{x}_b + \omega_x(I_{xy}\omega_z - I_{xz}\omega_y) + I_{yz}\left(\omega_z^2 - \omega_y^2\right) + \omega_y\omega_z(I_{yy} - I_{zz}) \\ \sum \vec{M} \cdot \hat{y}_b + \omega_y(I_{yz}\omega_x - I_{xy}\omega_z) + I_{xz}\left(\omega_x^2 - \omega_z^2\right) + \omega_x\omega_z(I_{zz} - I_{xx}) \\ \sum \vec{M} \cdot \hat{z}_b + \omega_z(I_{xz}\omega_y - I_{yz}\omega_x) + I_{xy}\left(\omega_y^2 - \omega_x^2\right) + \omega_x\omega_y(I_{xx} - I_{yy}) \end{bmatrix} \qquad \text{(EQ A-9)}$$

The angular acceleration obtained from solving (EQ A-9) can be integrated to give the angular velocity in the inertial (ECIC) reference frame.

In order to fully define the state of the vehicle in the 6DOF model, we also need equations describing its angular orientation. The vehicle's orientation with respect to the inertial frame is described by the coordinate transformation matrix, $\left[{}_I T_b\right]$, between the ECIC reference frame and the body-fixed reference frame. Let the elements of this matrix be denoted as C_{ij}:

$$\left[{}_I T_b\right] \equiv \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \qquad \text{(EQ A-10)}$$

The elements of the transformation matrix can be thought of as unit vectors defining the body coordinate system in terms of the inertial ECIC frame. Therefore,

$$\begin{aligned} \hat{x}_B &= C_{11}\hat{x}_I + C_{21}\hat{y}_I + C_{31}\hat{z}_I \\ \hat{y}_B &= C_{12}\hat{x}_I + C_{22}\hat{y}_I + C_{32}\hat{z}_I \\ \hat{z}_B &= C_{13}\hat{x}_I + C_{23}\hat{y}_I + C_{33}\hat{z}_I \end{aligned} \qquad \text{(EQ A-11)}$$

The tranformation is orthogonal and so we can express the inertial unit vectors in terms of the body unit vectors as:

$$\begin{aligned} \hat{x}_I &= C_{11}\hat{x}_B + C_{12}\hat{y}_B + C_{13}\hat{z}_B \\ \hat{y}_I &= C_{21}\hat{x}_B + C_{22}\hat{y}_B + C_{23}\hat{z}_B \\ \hat{z}_I &= C_{31}\hat{x}_B + C_{32}\hat{y}_B + C_{33}\hat{z}_B \end{aligned} \qquad \text{(EQ A-12)}$$

The inertial unit vectors do not change with time so differentiating them with respect to time and setting the result equal to zero yields equations giving the rate of change of the body unit vectors in terms of the transformation components and the angular velocity. (EQ A-2) has again been applied to carry out the derivative in the (rotating) body -fixed coordinate system. The resulting equations, called Poisson's kinematical equations, are:

$$\begin{aligned} \dot{C}_{11} &= C_{12}\omega_z - C_{13}\omega_y \\ \dot{C}_{12} &= C_{13}\omega_x - C_{11}\omega_z \\ \dot{C}_{13} &= C_{11}\omega_y - C_{12}\omega_x \end{aligned} \qquad \text{(EQ A-13)}$$

$$\begin{aligned} \dot{C}_{21} &= C_{22}\omega_z - C_{23}\omega_y \\ \dot{C}_{22} &= C_{23}\omega_x - C_{21}\omega_z \\ \dot{C}_{23} &= C_{21}\omega_y - C_{22}\omega_x \end{aligned} \qquad \text{(EQ A-14)}$$

$$\begin{aligned} \dot{C}_{31} &= C_{32}\omega_z - C_{33}\omega_y \\ \dot{C}_{32} &= C_{33}\omega_x - C_{31}\omega_z \\ \dot{C}_{33} &= C_{31}\omega_y - C_{32}\omega_x \end{aligned} \qquad \text{(EQ A-15)}$$

Given the initial orientation of the vehicle and the angular velocity as a function of time, these first-order differential equations can be integrated to give the orientation of the vehicle as a function of time.

A.5 The State Vector

The trajectory of the vehicle is defined by its state vector, which is a function of time. For a point mass trajectory, the state vector is composed of the instantaneous position and velocity of the vehicle:

$$\vec{S}_{3DOF}(t) = \begin{bmatrix} \vec{r} \\ \vec{V} \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \qquad \text{(EQ A-16)}$$

For rigid body trajectories, the state vector is the position and velocity of the center-of-mass of the vehicle plus the angular orientation and angular velocity. It is given by:

$$\vec{S}_{6DOF}(t) = \begin{bmatrix} \vec{r} \\ \vec{V} \\ \vec{\omega} \\ \left[{}_I T_b\right] \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \\ C_{11} \\ C_{12} \\ C_{13} \\ C_{21} \\ C_{22} \\ C_{23} \\ C_{31} \\ C_{32} \\ C_{33} \end{bmatrix} \qquad \text{(EQ A-17)}$$

The equations of motion derived in the previous sections can be expressed as a set of coupled first-order differential equations. For the point mass trajectories, the equations are:

$$\frac{d}{dt}\vec{S}(t) = \dot{\vec{S}}(t)$$

$$\dot{\vec{S}}(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \qquad \text{(EQ A-18)}$$

where $\ddot{x}$, $\ddot{y}$, and $\ddot{z}$ are the *x*-, *y*-, and *z*- components of the acceleration, respectively. Expressions for $\ddot{x}$, $\ddot{y}$, and $\ddot{z}$ can be obtained from the *x*-, *y*-, and *z*- components of (EQ A-3) as follows:

$$\begin{aligned} \ddot{x} &= \frac{\sum \vec{F} \cdot \hat{x}}{m} + \omega_E(2\dot{y} + \omega_E x) \\ \ddot{y} &= \frac{\sum \vec{F} \cdot \hat{y}}{m} + \omega_E(-2\dot{x} + \omega_E y) \\ \ddot{z} &= \frac{\sum \vec{F} \cdot \hat{z}}{m} \end{aligned} \qquad \text{(EQ A-19)}$$

In our calculations, we assume a non-rotating Earth: $\omega_E = 0$. Under this assumption, (EQ A-19) simplifies to

$$\ddot{x} = \frac{\sum \vec{F} \cdot \hat{x}}{m} \qquad \ddot{y} = \frac{\sum \vec{F} \cdot \hat{y}}{m} \qquad \ddot{z} = \frac{\sum \vec{F} \cdot \hat{z}}{m} \qquad \text{(EQ A-20)}$$

For rigid body trajectories, the coupled equations are:

$$\frac{d}{dt}\vec{S}(t) = \dot{\vec{S}}(t)$$

$$\dot{\vec{S}}(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \\ \dot{C}_{11} \\ \dot{C}_{12} \\ \dot{C}_{13} \\ \dot{C}_{21} \\ \dot{C}_{22} \\ \dot{C}_{23} \\ \dot{C}_{31} \\ \dot{C}_{32} \\ \dot{C}_{33} \end{bmatrix} \qquad \text{(EQ A-21)}$$

The first six elements of the derivative vector are the same as those for the point mass trajectory. The derivatives $\dot{\omega}_x$, $\dot{\omega}_y$, and $\dot{\omega}_z$, are obtained by solving (EQ A-9). The transformation matrix derivatives $\dot{C}_{ij}$ are obtained from (EQ A-13) - (EQ A-15).

The trajectory of the missile for the 3DOF model or the 6DOF model is obtained by integrating the appropriate set of coupled equations to obtain the state vector as a function of time. Details of the numerical integration procedure used for the flight dynamics example are given in Appendix B.

Appendix B: Numerical Integration Procedure

In Appendix A, we derived the equations of motion for both a 3DOF model and a 6DOF model of vehicle flight dynamics. These equations of motion were solved for the air-to-ground missile flight dynamics example using the Trajectory Analysis and Optimization Software (TAOS).[91] In this Appendix, we discuss the numerical procedures used to solve the equations of motion and review how various numerical effects were quantified in the final results.

B.1 The Augmented State Vector

In Appendix A, the equations of motion were written in terms of the state vector of the missile. For a point mass trajectory, the state vector is composed of the instantaneous position and velocity of the missile. For a rigid body trajectory, the state vector is composed of the position and velocity of the center-of-mass of the vehicle plus its angular orientation and angular velocity. TAOS augments these nominal state vectors with three additional variables: the mass of the vehicle, the flight path length, and the ground range. The mass is added to the state vectors so that variable mass vehicles can be modeled. The path length and ground range are key outputs associated with the trajectory that require integration to compute and, therefore, are included as part of the state vector.

Thus, the augmented state vector for a point mass trajectory is

$$\hat{S}_{3DOF,a} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ m \\ r \\ r_S \end{bmatrix} \qquad \text{(EQ B-1)}$$

where x, y, and z are the components of the position vector; $\dot{x}$, $\dot{y}$, and $\dot{z}$ are the components of the velocity vector; m is the mass; r is the path length; and r_S is the ground range.

The augmented state vector for a rigid body trajectory is given by:

$$\vec{S}_{6DOF,a} = \begin{bmatrix} \vec{r} \\ \vec{V} \\ \vec{\omega} \\ [{}_I T_b] \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \\ C_{11} \\ C_{12} \\ C_{13} \\ C_{21} \\ C_{22} \\ C_{23} \\ C_{31} \\ C_{32} \\ C_{33} \\ m \\ r \\ r_S \end{bmatrix} \quad \text{(EQ B-2)}$$

where x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$, m, r, and r_S were previously defined; ω_x, ω_y, and ω_z are the components of the angular velocity vector, and the C_{ij} define the orientation of the vehicle with respect to the ECIC (Earth-Centered, Inertial Coordinate System) reference frame.

The state vector is computed numerically by integrating with respect to time a set of first-order, coupled differential equations for the state vector. For the point mass trajectories, the equations are:

$$\frac{d}{dt}\vec{S}_{3DOF,a}(t) = \dot{\vec{S}}_{3DOF,a}(t)$$

$$\dot{\vec{S}}_{3DOF,a}(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{m} \\ \left\|\vec{V}\right\| \\ \left\|\vec{V}_S\right\| \end{bmatrix} \qquad \text{(EQ B-3)}$$

where

$$\ddot{x} = \frac{\sum \vec{F} \cdot \hat{x}}{m}$$

$$\ddot{y} = \frac{\sum \vec{F} \cdot \hat{y}}{m} \qquad \text{(EQ B-4)}$$

$$\ddot{z} = \frac{\sum \vec{F} \cdot \hat{z}}{m}$$

The mass flow rate, $\dot{m}$, is an input to the problem; $\left\|\vec{V}\right\|$ is the magnitude of the velocity; and $\left\|\vec{V}_S\right\|$ is the ground speed. Recall from Appendix A that the simplified equations in (EQ B-4) result from the assumption of a non-rotating Earth. (EQ B-3) is a system of nine, first-order, nonlinear differential equations.

For rigid body trajectories, the coupled equations are:

$$\frac{d}{dt}\vec{S}_{6DOF,a}(t) = \dot{\vec{S}}_{6DOF,a}(t)$$

$$\dot{\vec{S}}_{6DOF,a}(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \\ \dot{C}_{11} \\ \dot{C}_{12} \\ \dot{C}_{13} \\ \dot{C}_{21} \\ \dot{C}_{22} \\ \dot{C}_{23} \\ \dot{C}_{31} \\ \dot{C}_{32} \\ \dot{C}_{33} \\ \dot{m} \\ \left\| \dot{\vec{V}} \right\| \\ \left\| \dot{\vec{V}}_S \right\| \end{bmatrix} \qquad \text{(EQ B-5)}$$

The first six and last three elements of the 6DOF derivative vector are the same as those for the point mass trajectory. The derivatives $\dot{\omega}_x$, $\dot{\omega}_y$, and $\dot{\omega}_z$ are obtained from solving (EQ A-9). The transformation matrix derivatives C_{ij} are obtained from (EQ A-13)-(EQ A-15). (EQ B-5) is a system of twenty-one, first order nonlinear differential equations.

As we saw in Appendix A, (EQ A-9) is a system of linear equations which can be written in the standard matrix notation. The equations are generally well-behaved and are solved using Gaussian elimination at each derivative evaluation.

B.2 Runge-Kutta Integration

Given an initial state vector $\vec{S}(t_0)$, numerical integration techniques will compute a new state vector at time $t + \Delta t$. TAOS uses an embedded Runge-Kutta-Fehlberg technique to do the integration.[89] A Runge-Kutta method estimates the function to be integrated at specified points forward in time and combines these function evaluations to match the lower-order terms in a Taylor series expansion for the solution. It is possible to find function evaluation points which can be combined with different coefficients to yield different order of accuracy estimates for the solution. These combinations of evaluation points and coefficients make up the class known as "embedded" Runge-Kutta integrators. The advantage to this approach is that the difference in estimates for the solution given by the different order methods can be used to estimate the magnitude of the integration error. This estimate can then be used to ensure that each time step taken meets a specified per-step truncation error criterion. The per-step integration error estimate can also be used to estimate an optimum value for the next time step.

The state vector is integrated using a fifth-order accurate formula with coefficients found by Cash and Karp:[90]

$$\vec{S}(t + \Delta t) = \vec{S}(t) + \Delta t\left(c_1\dot{\vec{S}}_1 + c_3\dot{\vec{S}}_3 + c_4\dot{\vec{S}}_4 + c_6\dot{\vec{S}}_6 \right) + O(\Delta t^6) \qquad \text{(EQ B-6)}$$

where

$$\begin{aligned}
\dot{\vec{S}}_1 &= \dot{\vec{S}}(t, \vec{S}) \\
\dot{\vec{S}}_2 &= \dot{\vec{S}}\left(t + a_2\Delta t, \vec{S} + \Delta t\left(b_{21}\dot{\vec{S}}_1 \right) \right) \\
\dot{\vec{S}}_3 &= \dot{\vec{S}}\left(t + a_3\Delta t, \vec{S} + \Delta t\left(b_{31}\dot{\vec{S}}_1 + b_{32}\dot{\vec{S}}_2 \right) \right) \\
\dot{\vec{S}}_4 &= \dot{\vec{S}}\left(t + a_4\Delta t, \vec{S} + \Delta t\left(b_{41}\dot{\vec{S}}_1 + b_{42}\dot{\vec{S}}_2 + b_{43}\dot{\vec{S}}_3 \right) \right) \\
\dot{\vec{S}}_5 &= \dot{\vec{S}}\left(t + a_5\Delta t, \vec{S} + \Delta t\left(b_{51}\dot{\vec{S}}_1 + b_{52}\dot{\vec{S}}_2 + b_{53}\dot{\vec{S}}_3 + b_{54}\dot{\vec{S}}_4 \right) \right) \\
\dot{\vec{S}}_6 &= \dot{\vec{S}}\left(t + a_6\Delta t, \vec{S} + \Delta t\left(b_{61}\dot{\vec{S}}_1 + b_{62}\dot{\vec{S}}_2 + b_{63}\dot{\vec{S}}_3 + b_{64}\dot{\vec{S}}_4 + b_{65}\dot{\vec{S}}_5 \right) \right)
\end{aligned} \qquad \text{(EQ B-7)}$$

The term $O(\Delta t^6)$ shows that the numerical integration error is of the order Δt^6. The coefficients are:

$$a_2 = \frac{1}{5} \qquad a_3 = \frac{3}{10} \qquad a_4 = \frac{3}{5} \qquad a_5 = 1 \qquad a_6 = \frac{7}{8}$$

$$b_2 = \frac{1}{5}$$

$$b_{31} = \frac{3}{40} \qquad b_{32} = \frac{9}{40}$$

$$b_{41} = \frac{3}{10} \qquad b_{42} = \frac{-9}{10} \qquad b_{43} = \frac{6}{5}$$

$$b_{51} = \frac{-11}{54} \qquad b_{52} = \frac{5}{2} \qquad b_{53} = \frac{-70}{27} \qquad b_{54} = \frac{35}{27}$$

$$b_{61} = \frac{1631}{55296} \qquad b_{62} = \frac{175}{512} \qquad b_{63} = \frac{575}{13824} \qquad b_{64} = \frac{44275}{110592} \qquad b_{65} = \frac{253}{4096}$$

$$c_1 = \frac{37}{378} \qquad c_3 = \frac{250}{621} \qquad c_4 = \frac{125}{594} \qquad c_6 = \frac{512}{1771}$$

The “embedded” fourth-order formula is:

$$\vec{\dot{S}}'(t + \Delta t) = \vec{\dot{S}}(t) + \Delta t\left(c_1'\vec{\dot{S}}_1 + c_3'\vec{\dot{S}}_3 + c_4'\vec{\dot{S}}_4 + c_6'\vec{\dot{S}}_6\right) + O(\Delta t^5) \qquad \text{(EQ B-8)}$$

where

$$c_1' = \frac{2825}{27648} \qquad c_3' = \frac{18575}{48384} \qquad c_4' = \frac{13525}{55296} \qquad c_5' = \frac{277}{14336} \qquad c_6' = \frac{1}{4}$$

The fourth-order and fifth-order accurate solutions can be used to estimate a relative, per-step, truncation error, $\Delta\vec{\dot{S}}$. The elements of $\Delta\vec{\dot{S}}$ are the relative, per-step truncation error of each dependent variable in the system of equations to be integrated:

$$\Delta S_i = \left(\frac{\left|S_i(t + \Delta t) - S'_i(t + \Delta t)\right|}{\left|S_i(t + \Delta t)\right|}\right) \qquad \text{(EQ B-9)}$$

where $i = 1, 2 \ldots 9$ for the 3DOF model and $i = 1, 2 \ldots 21$ for the 6DOF model.

The next three sections discuss how this estimate of integration error at each step is used to control the integration error and estimate the next time step.

B.3 Requirement to Satisfy Relative Error Criterion for all Variables

In order to accept a given time step, Δt, the integrator requires that each element of the error vector defined in (EQ B-9) be less than a user-defined relative, per-step truncation error, E_t. That is, for a step to be accepted, the following equations must be satisfied for the 3DOF and 6DOF equations of motion:

$$\begin{aligned} \Delta S_i < E_t \qquad & i = 1, 2, \ldots 9 \qquad (3DOF) \\ \Delta S_i < E_t \qquad & i = 1, 2, \ldots 21 \qquad (6DOF) \end{aligned} \tag{EQ B-10}$$

If one or more elements of the error vector do not meet the error criterion, the time step is rejected and a new time step is estimated as described in Section B.5 below.

One implication of applying the relative error criterion to each component of the error vector is that variables that have rapidly changing derivatives (oscillatory functions) or no derivative at all (step functions) tend to dominate the determination of the step size.

B.4 Switching from Relative Error to Absolute Error

(EQ B-9) will not be usable when a component of the state vector is very near the floating-point zero of the computer. Therefore, one must switch from the relative error criterion to an absolute error criterion for this case. In TAOS, the per-step truncation error estimate is modified from that in (EQ B-9) as follows:

$$\Delta S_i = \frac{\left| S_i(t + \Delta t) - S'_i(t + \Delta t) \right|}{f^{\,i}_{scale}}. \tag{EQ B-11}$$

$f^{\,i}_{scale}$ are non-zero scale factors for each component of the state vector. For a given state variable, S_i, the scale factor is given by:

$$f^{\,i}_{scale}(S_i) = max(\left|S_i\right|, limit S_i) \tag{EQ B-12}$$

where *limit* S_i is the limiting value for each component in the state vector. The limiting value is the point at which the error criterion switches from a relative criterion to an absolute criterion. The limiting values for each state variable are given in Table .

Note that the limiting values given in Table differ from those in Ref. 91. The values given above are correct for the version of TAOS used to perform the calculations described in this report. The limiting values for the elements of $[C]$ were later changed to 1×10^{-6} as described in Reference 91.

B.5 Estimating the next Δt

If the per-step truncation error using (EQ B-9) or (EQ B-11) exceeds the specified error tolerance, E_t, a smaller time step is attempted. The time step estimation algorithm is somewhat heuristic in nature and described elsewhere;[89] therefore we will simply state the method used without further justification.

Let $\Delta\underset{\sim}{\dot{S}}$ be the estimated error in the state vector for a given time step, Δt_{old}. Let Δ_{max} be the largest element of ΔS, i.e., the maximum relative error. The new attempted time step is computed using:

$$\Delta t_{new} = 0.90 \Delta t_{old} \left| \frac{E_t}{\Delta_{max}} \right|^{0.25} \qquad \text{(EQ B-13)}$$

Table B-1
Limiting Values of State Variables

State Variable	Limiting Value	Comment
x	1.0 ft.	3DOF and 6DOF
y	1.0 ft.	3DOF and 6DOF
z	1.0 ft.	3DOF and 6DOF
$\dot{x}$	0.01 ft./s	3DOF and 6DOF
$\dot{y}$	0.01 ft/s	3DOF and 6DOF
$\dot{z}$	0.01 ft/s	3DOF and 6DOF
ω_x	1×10^{-6} radians/s	6DOF only
ω_y	1×10^{-6} radians/s	6DOF only
ω_z	1×10^{-6} radians/s	6DOF only
C_{11}	1×10^{-5}	6DOF only
C_{12}	1×10^{-5}	6DOF only
C_{13}	1×10^{-5}	6DOF only
C_{21}	1×10^{-5}	6DOF only
C_{22}	1×10^{-5}	6DOF only
C_{23}	1×10^{-5}	6DOF only
C_{31}	1×10^{-5}	6DOF only
C_{32}	1×10^{-5}	6DOF only
C_{33}	1×10^{-5}	6DOF only
m	1×10^{-6} slugs	3DOF and 6DOF
r	1×10^{-6} ft.	3DOF and 6DOF
r_S	1×10^{-6} ft.	3DOF and 6DOF

This new attempted time step is a conservative estimate of a time step that should yield a per step truncation error that will satisfy the error criterion. Since the integration method is fifth-order, small changes in time step can lead to very large changes in integration error. To provide a bound on the maximum reduction in the time step for a given attempted step, the new time step is constrained to satisfy:

$$\Delta t_{new} \geq 0.1 \Delta t_{old}. \qquad \text{(EQ B-14)}$$

Similarly, after either the relative error or absolute error criterion is satisfied for each element of the state vector, the size of the next time step is also estimated. The time step is based on how well the current time step has satisfied the required error criterion. The goal of the time step adjustment is to choose the next time step so that it is nearly as large as possible while still satisfying the error criterion. If the error criterion is easily satisfied, a larger time step will be attempted; if the error criterion is barely satisfied, the time step will change little.

The estimate of the new time step is given by:

$$\Delta t_{new} = 0.90 \Delta t_{old} \left| \frac{E_t}{\Delta_{max}} \right|^{0.20} \qquad \text{(EQ B-15)}$$

Again, small changes in time step can lead to very large changes in integration error. To provide a bound on the maximum increase in the step size, it is constrained using:

$$\Delta t_{new} \leq 5 \Delta t_{old}. \qquad \text{(EQ B-16)}$$

Generally, numerical integration methods also use a minimum time step to prevent the integrator from using exceedingly small time steps. This minimum time step value is an input parameter to the code. The invocation of the minimum time step implies that the per step truncation error exceeds the requested error criterion in at least one state variable. Since we wished to have tight control over the numerical solution error in the computations for the flight dynamics problem, we were careful to select a minimum time step smaller than that ever used by the integrator. Our value was 1×10^{-8} s.

TAOS is programmed such that the initial time step attempted (i.e., at $t = 0$) is 0.1 s. This arbitrary initial value for the time step has no effect on the solution accuracy because the solution accuracy is controlled by the per step truncation error criterion.

B.6 Print Interval Effects

It is generally desirable to have trajectory information at uniformly spaced intervals; the use of a variable step-size integrator therefore necessitates providing some method for estimating the state vector at uniform intervals based on the non-uniform steps actually taken by the integrator. A common approach is to interpolate using the values of the state vector obtained from the integrator. This would cause a data representation error, in the terminology discussed in Sections 3 and 4. TAOS takes an alternate approach that eliminates the interpolation error, but this

approach can affect the numerical integration process itself. Let us define the print interval as the (uniform) time interval for which the user wants state vector information. The essence of the approach used in TAOS is that the time step used by the integrator is adjusted so that the integrator computes the value of the state vector at all of the print intervals. We describe the details of this approach and potential effects on the calculations below.

Suppose the integrator is at time t_1 and the current time step, Δt, has satisfied the per-step truncation error criterion. Suppose further that taking this time step would advance the integrator past the next print interval. TAOS will reduce the time step, Δt, to a new value $\Delta t'$ such that the integrator will advance just to the print interval. The new reduced time step $\Delta t'$ will then be checked to ensure that it satisfies the per-step truncation error criterion. If the time step $\Delta t'$ satisfies the error criterion, the state vector will be computed and saved as an acceptable numerical integration step. Now the question is how to proceed with the integration from the print interval. Normally, the process of ensuring that the error criterion is satisfied results in a new recommended time step according to the methods of Section B.5. Three options for proceeding are available:

1. Back up to t_1 and proceed with the recommended time step Δt_{new}, associated with the acceptable time step, Δt, that would have stepped over the print interval. While this method eliminates any effects of the print interval on the numerical integration, it complicates the implementation of the integrator. As a result, this option was not implemented.
2. Proceed from the print interval with the recommended time step, $\Delta t'_{new}$, associated with the reduced time step $\Delta t'$. However, $\Delta t'_{new}$ is likely to be artificially small as a result of the time step reduction required to hit the print interval. This approach thus maximizes the effect of the print interval on the numerical integration and was also deemed unsatisfactory.
3. Proceed from the print interval with the recommended time step, Δt_{new}, associated with the acceptable time step, Δt, that would have stepped over the print interval. This approach keeps the implementation of the integrator relatively straightforward while minimizing the impact of the print interval on the numerical integration. This method is used in TAOS.

Occasionally we observed that the (reduced) time step, $\Delta t'$, required to land on the print interval failed to satisfy the per-step truncation error criterion. When this happened, the associated recommended new time step, referred to as $\Delta t'_{new}$ above, was used to re-initiate the process of finding an acceptable time step from time t_1.

In order to monitor the behavior of the integrator, TAOS provides the option to output the largest and smallest step size taken in each print interval. If these are equal to the print interval itself, the print interval is dominating the time step selection process and each time step has less error (by an indeterminate amount) than the error criterion suggests. While this is not a problem for most analyses, it was for most of the calculations in this report. As discussed in the body of this report, we wished to investigate the effect of numerical error and, therefore, we wanted to directly control the error. For the detailed trajectories described in Appendix D, we used a print interval of 0.005 s. in order to capture all of the details of the trajectories. This print interval dominated the time step selection process for the 3DOF model most of the time, but did not dominate

the time step selection process in the 6DOF model. We were able to tolerate this for the detailed trajectories since the purpose of these runs was to illustrate the flight dynamics of the missile.

For all of the runs discussed in the body of the report, we were only concerned with the final range. Therefore, we used a print interval larger than the length of the trajectory in order to be certain that the print interval was not affecting the results.

Intentionally blank

Appendix C: Detailed Problem Description

C.1 Introduction

The initial conditions, environment specification, and vehicle properties for the flight dynamics example are given in this Appendix. The mass properties and aerodynamic force coefficients are given in either the body coordinate system or auxiliary body coordinate system as described in Appendix A. We provide a brief review of these two coordinate systems below.

The primary body coordinate system is shown in Figure C-1a. Note that this coordinate system has its origin at the center of mass with the x-axis oriented toward the front of the vehicle. Since the 3DOF model represents the vehicle as a point which is, by definition, the center of mass, this is the only body coordinate system used in the 3DOF model. The 6DOF model requires input about the location of the aerodynamic forces as well as their magnitude. It is convenient to define an auxiliary body coordinate system to simplify the input files. The auxiliary body coordinate system has the same orientation as the primary body coordinate system, but is offset to some other convenient location. An auxiliary body coordinate system is shown in Figure C-1b. For our calculations, the origin of the auxiliary body coordinate system was located at the nose of the missile.

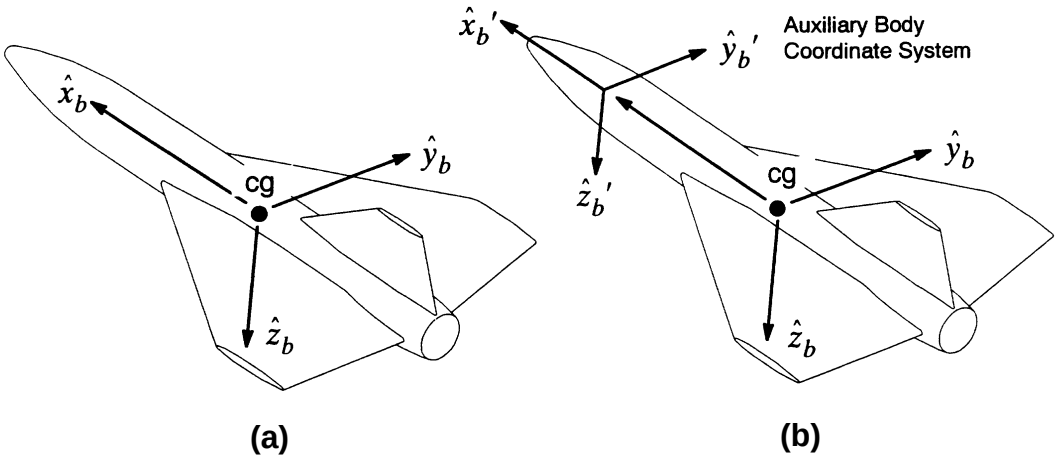

Figure C-1
Body Coordinate Systems[91]

C.2 Initial Conditions

The flight dynamics example studied in this report is the flight of an unguided, rocket-boosted, air-launched missile. For convenience, detailed missile characteristics were taken to be those of the Improved Hawk missile, since these were readily available.[93]

Calculations were done using both a 6DOF model and a 3DOF model with TAOS (version 98.7), the Trajectory Analysis and Optimization Software.[91] This Appendix describes the input parameters required in TAOS.

We assume the launching aircraft is flying straight and level at an altitude of 30,000 ft. above mean sea level at the time of launch. We assume that the missile at launch is undisturbed due to the aricraft flow field or release from the aircraft. Therefore, the missile is launched horizontally, at zero angle of attack and zero sideslip angle, and with zero angular rates. The missile is launched due north along the Greenwich meridian from the Equator. The initial conditions for the missile launch are summarized in Table C-.

Table C-1
Initial Conditions

Parameter	Symbol	Initial Value
altitude	h	30,000 ft.
longitude	δ	0.0^o
latitude	λ	0.0^o
velocity	v	700 ft./s
flight path angle	γ	0.0^o
heading angle	ψ	0.0^o
yaw	Ψ	0.0^o
pitch	Θ	0.0^o
roll	Φ	0.0^o
rate about body x-axis	p	0.0 deg/s
rate about body y-axis	q	0.0 deg/s
rate about body z-axis	r	0.0 deg/s
mass	m	42.86 slugs

C.3 Environment Specification

We assume a spherical, non-rotating Earth with the Earth's radius, gravitational constant and acceleration of gravity based on the *U.S. Department of Defense World Geodetic System 1984* values.[96] The values are summarized in Table C-2.

The atmospheric properties are given by the 1976 U.S. Standard Atmosphere.[92] It is assumed that there is no wind over the entire path of the trajectory.

Table C-2
Earth Model Properties

Description	Symbol	Value
Earth radius	r_{eqtr}	$2.09256463255 \times 10^{7}$ ft
gravitational constant	G	$1.40764438125 \times 10^{16}$ ft^3/s^2
acceleration of gravity	g	32.1740485 ft/s^2

C.4 Propulsion

The trajectory is split into two phases -- a boost phase and a coast phase. For the Improved Hawk rocket motor at nominal operating temperature, the boost phase lasts for 24.9 seconds. The propulsion properties used to characterize this phase are a static thrust profile and a mass flow rate. The actual thrust experienced in flight is calculated from the static thrust under vacuum conditions, t_{vac}, minus the atmospheric pressure, p, acting on the exhaust nozzle of the motor. This gives the equation for the net thrust:

$$thrust = t_{vac} - 0.5248p \quad \text{(EQ C-1)}$$

As noted in the body of the report, we considered the effects of uncertainty in the initial temperature of the motor. In addition to a nominal motor temperature, we considered flight with a hot motor and a cold motor, corresponding to the highest operational temperature allowed by the manufacturer and the lowest operational temperature allowed by the manufacturer, respectively. To be representative of thrust uncertainty in actual motors, we used changes in performance that have been experimentally measured for the Standard Hawk motor.[95] At the highest allowed temperature of 120^o F, the total impulse of the motor is 2% above nominal performance while the burn time is 7% shorter. At the lowest allowed temperature of -20^o F, the total impulse is 2% below nominal and the burn time is 7% longer. To obtain input tables for the hot and cold motors, we adjusted the tabular values for the nominal motor as follows:

$$t_{hot} = 0.93 t_{nom} \quad \text{(EQ C-2)}$$

$$t_{vac,\,hot} = 1.02 t_{vac,\,nom} / 0.93 \quad \text{(EQ C-3)}$$

$$\dot{m}_{hot} = \dot{m}_{nom} / 0.93 \quad \text{(EQ C-4)}$$

$$t_{cold} = 1.07 t_{nom} \quad \text{(EQ C-5)}$$

$$t_{vac,\,cold} = 0.98 t_{vac,\,nom} / 1.07 \quad \text{(EQ C-6)}$$

$$\dot{m}_{cold} = \dot{m}_{nom} / 1.07 \quad \text{(EQ C-7)}$$

The static thrust and mass flow rate as a function of time for each of the three motor temperatures are given in Table C-3.

The thrust data are plotted in Figure C-2. During the first five seconds, the thrust of the nominal motor varies in an oscillatory fashion near a nominal value of 19,000 lbs. The nominal thrust then drops rapidly to a value near 3,000 lbs. and remains near this lower level until finally tapering off between 22 and 24.9 seconds. Note that the thrust of the hot motor is higher and of shorter duration than that of the nominal motor while the thrust of the cold motor is lower and of longer duration.

The mass flow rate data are plotted in Figure C-3. The mass flow rate for the nominal temperature motor oscillates near -2.2 slugs/s during the high thrust portion. It falls to a lower rate of -0.5 to -0.75 slugs/s at approximately 5 seconds and remains at this lower rate until finally tapering off to zero between 21 and 24.9 seconds. The mass flow rate as a functon of time is, as would be expected, very similar to the thrust versus time. However, they are not simply different by a constant factor. One of the reasons for this is the efficiency of the nozzle is not constant during motor burn.

In the 6DOF model, the point at which the thrust acts must also be specified in order to allow proper calculation of moments. For our problem, the location was specified as (-14.46 ft, 0.0 ft, 0.0 ft) in the auxiliary body coordinate system defined above (i.e., with the origin located at the nose of the missile). The point at which the thrust acts was assumed constant with time.

Table C-3
Thrust Data for Nominal,[93] Cold, and Hot Motors

Nominal Motor			Cold Motor			Hot Motor		
Time (s)	t_{vac} (lb)	$\dot{m}$ (slugs/s)	Time (s)	t_{vac} (lb)	$\dot{m}$ (slugs/s)	Time (s)	t_{vac} (lb)	$\dot{m}$ (slugs/s)
0.000	1110.6	0.000	0.000	1017.19	0.000	0.000	1218.08	0.000
0.001	20525.9	-2.306	0.00107	18799.4	-2.15514	0.00093	22512.3	-2.47957
0.200	19696.5	-2.208	0.214	18039.8	-2.06355	0.186	21602.6	-2.37419
0.400	18867.2	-2.109	0.428	17280.2	-1.97103	0.372	20693.1	-2.26774
0.600	18379.6	-2.030	0.642	16833.7	-1.8972	0.558	20158.3	-2.1828
0.800	18331.3	-2.045	0.856	16789.4	-1.91121	0.744	20105.3	-2.19892
1.000	18574.6	-2.074	1.07	17012.3	-1.93832	0.93	20372.1	-2.23011
1.200	18932.2	-2.117	1.284	17339.8	-1.9785	1.116	20764.3	-2.27634
1.400	19290.7	-2.159	1.498	17668.1	-2.01776	1.302	21157.5	-2.32151
1.600	19648.3	-2.202	1.712	17995.6	-2.05794	1.488	21549.7	-2.36774
1.800	19867.9	-2.228	1.926	18196.8	-2.08224	1.674	21790.6	-2.3957
2.000	19671.9	-2.205	2.14	18017.3	-2.06075	1.86	21575.6	-2.37097
2.200	19258.2	-2.156	2.354	17638.4	-2.01495	2.046	21121.9	-2.31828
2.400	18843.5	-2.106	2.568	17258.5	-1.96822	2.232	20667.1	-2.26452
2.600	18599.3	-2.077	2.782	17034.9	-1.94112	2.418	20399.2	-2.23333
2.800	18526.4	-2.069	2.996	16968.1	-1.93364	2.604	20319.3	-2.22473
3.000	18452.5	-2.060	3.21	16900.4	-1.92523	2.79	20238.2	-2.21505
3.200	18644.6	-2.083	3.424	17076.4	-1.94673	2.976	20448.9	- 2.23978
3.400	18836.6	-2.105	3.638	17252.2	-1.96729	3.162	20659.5	-2.26344
3.600	19027.7	-2.128	3.852	17427.2	-1.98879	3.348	20869.1	- 2.28817
3.800	19225.7	-2.152	4.066	17608.6	-2.01121	3.534	21086.3	- 2.31398
4.000	19428.6	-2.176	4.28	17794.4	-2.03364	3.72	21308.8	-2.33978
4.200	19306.5	-2.161	4.494	17682.6	-2.01963	3.906	21174.9	- 2.32366
4.400	15515.2	-1.711	4.708	14210.2	1.59907	4.092	17016.7	-1.83978
4.600	11724.0	-1.261	4.922	10737.9	-1.1785	4.278	12858.6	-1.35591
4.800	8697.1	-0.901	5.136	7965.57	-0.842056	4.464	9538.75	-0.968817
5.000	7965.2	-0.814	5.35	7295.23	-0.760748	4.65	8736.03	-0.875269

Table C-3
Thrust Data for Nominal,[93] Cold, and Hot Motors

Nominal Motor			Cold Motor			Hot Motor		
Time (s)	t_{vac} (lb)	$\dot{m}$ (slugs/s)	Time (s)	t_{vac} (lb)	$\dot{m}$ (slugs/s)	Time (s)	t_{vac} (lb)	$\dot{m}$ (slugs/s)
5.200	7233.3	-0.727	5.564	6624.89	-0.679439	4.836	7933.3	-0.78172
5.400	6596.0	-0.652	5.778	6041.2	-0.609346	5.022	7234.32	-0.701075
5.600	5957.8	-0.576	5.992	5456.68	-0.538318	5.208	6534.36	- 0.619355
5.800	5320.5	-0.500	6.206	4872.98	-0.46729	5.394	5835.39	-0.537634
5.850	5160.9	-0.481	6.2595	4726.81	-0.449533	5.4405	5660.34	-0.51720
6.000	4480.3	-0.520	6.42	4103.45	-0.485981	5.58	4913.88	0.55914
6.200	3766.1	-0.567	6.634	3449.33	-0.529907	5.766	4130.56	-0.609677
6.400	3533.7	-0.517	6.848	3236.47	-0.483178	5.952	3875.67	-0.555914
6.600	3304.2	-0.468	7.062	3026.28	-0.437383	6.138	3623.96	-0.503226
6.800	3180.1	-0.442	7.276	2912.61	-0.413084	6.324	3487.85	-0.475269
7.000	3159.4	-0.437	7.49	2893.66	-0.408411	6.51	3465.15	- 0.469892
7.200	3138.7	-0.433	7.704	2874.7	-0.404673	6.696	3442.45	-0.465591
7.400	3159.4	-0.437	7.918	2893.66	-0.408411	6.882	3465.15	-0.469892
8.000	3221.4	-0.451	8.56	2950.44	-0.421495	7.44	3533.15	-0.484946
10.000	3402.7	-0.489	10.7	3116.49	-0.457009	9.3	3731.99	-0.525806
12.000	3581.9	-0.528	12.84	3280.62	-0.493458	11.16	3928.54	-0.567742
14.000	3763.2	-0.566	14.98	3446.67	-0.528972	13.02	4127.38	-0.608602
16.000	3946.4	-0.606	17.12	3614.46	-0.566355	14.88	4328.31	-0.651613
18.000	4131.6	-0.645	19.26	3784.08	-0.602804	16.74	4531.43	-0.693548
20.000	4347.3	-0.691	21.4	3981.64	-0.645794	18.6	4768.01	-0.743011
20.200	4369.9	-0.696	21.614	4002.34	-0.650467	18.786	4792.79	-0.748387
21.000	4150.3	-0.649	22.47	3801.21	-0.606542	19.53	4551.94	-0.697849
22.000	3369.2	-0.482	23.54	3085.81	-0.450467	20.46	3695.25	-0.51828
23.000	1020.0	-0.214	24.61	934.206	-0.2	21.39	1118.71	-0.230108
24.000	280.2	-0.056	25.68	256.632	-0.0523364	22.32	307.316	-0.0602151
24.900	17.2	0.000	26.643	15.7533	0.000	23.157	18.8645	0.000
99.000	17.2	0.000	105.93	15.7533	0.000	92.07	18.8645	0.000

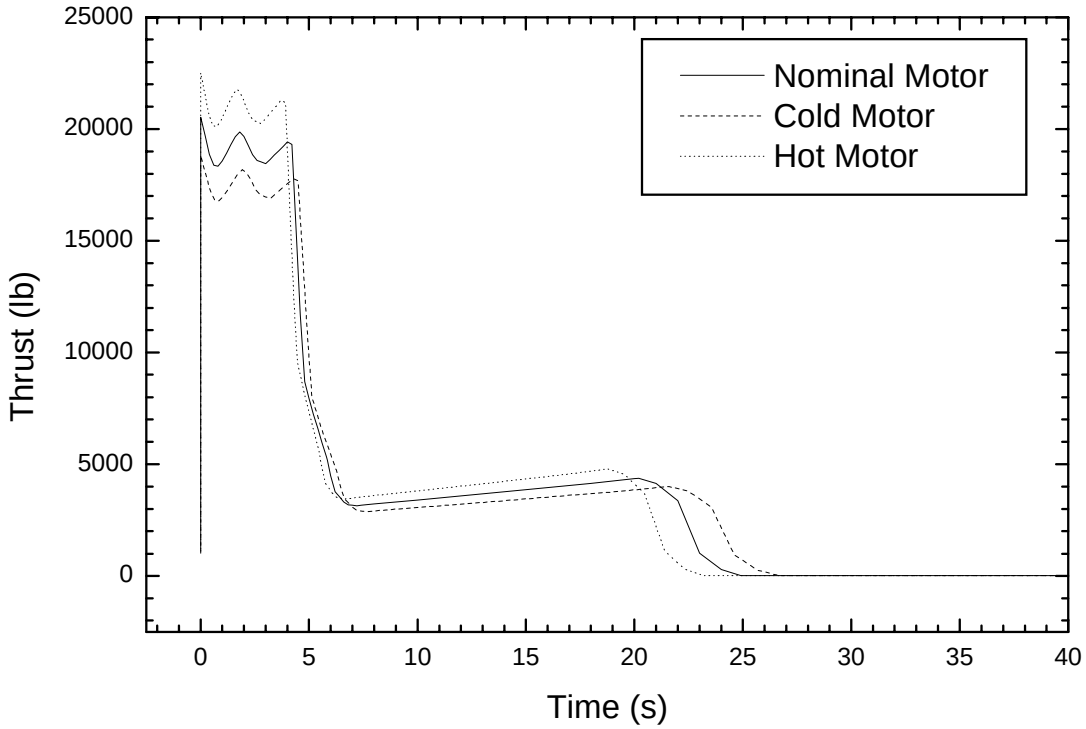

Figure C-2
Static Thrust in Vacuum

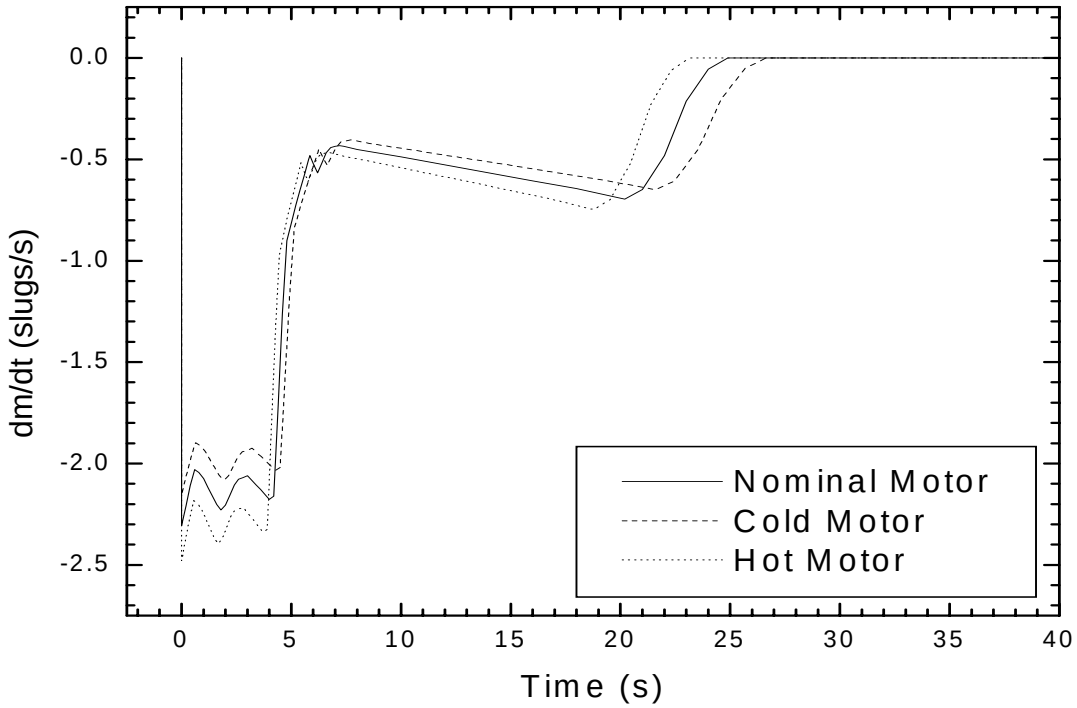

Figure C-3
Mass Flow Rate

C.5 Mass Properties

The change in the center-of-mass and the moments of inertia as fuel is consumed are modeled as piecewise linear functions that are a function of the mass of the missile. The values used are given in Table C-4. Note that the initial mass of the missile is 42.86 slugs and as time increases, mass decreases. Also note that the location of the center of gravity, X_{cg}, is given in the auxiliary body coordinate system. The moments of inertia are taken about the center of mass. The products of inertia, also taken about the center of mass, are assumed to be zero: $I_{xy} = 0$, $I_{yz} = 0$, and $I_{xz} = 0$.

Since no information was available concerning how the mass properties would vary with the temperature of the motor, we used the nominal values given in Table C-4 for all trajectories.

Table C-4
Mass Properties of the Missile[93]

Mass (slugs)	X_{cg} (ft)	I_{xx} (slug-ft^2)	I_{yy} (slug-ft^2)	I_{zz} (slug-ft^2)
0.00	-9.021	7.097	884.3	884.3
22.75	-9.021	7.097	884.3	884.3
23.25	-9.051	7.267	884.9	884.9
32.36	-9.552	9.537	914.5	914.5
42.86	-9.847	10.667	934.3	934.3
99.99	-9.847	10.667	934.3	934.3

C.6 Aerodynamic Forces and Moments

Aerodynamic forces and moments are forces and moments which act on a vehicle due to its motion through the atmosphere. They can be described in a coordinate system based on the body of the vehicle.

C.6.1 Aerodynamic Forces

The aerodynamic forces are modeled with a nondimensional force coefficient vector as:

$$\vec{F}_{aero} = \vec{C}_F q S_{ref} \quad \text{(EQ C-8)}$$

where $\vec{C}_f$ is the aerodynamic force coefficient vector, q is the dynamic pressure, and S_{ref} is a reference area. The dynamic pressure, q, is the increase in pressure on a vehicle due to its motion through the atmosphere and is given by

$$q = \frac{1}{2}\rho V_w^{\,2} \quad \text{(EQ C-9)}$$

where ρ is the density of the atmosphere and V_w is the magnitude of the wind-corrected velocity or airspeed. In our case, V_w is equal to $|\vec{V}|$ because there is no wind.

For small angles of attack and sideslip, the aerodynamic force can be approximated as linear with respect to angle of attack and sideslip. The functional form for the linearized aerodynamic force coefficients in the body coordinate system is given by:

$$C_X = C_{X_0} + C_{X_\alpha}|\alpha| + C_{X_\beta}|\beta| \quad \text{(EQ C-10)}$$

$$C_Y = C_{Y_0} + C_{Y_\beta}\beta + \frac{C_{Y_{\dot{\beta}}}\dot{\beta} l_{ref_y}}{2V_w} + \frac{C_{Y_p} p l_{ref_y}}{2V_w} + \frac{C_{Y_r} r l_{ref_y}}{2V_w} \quad \text{(EQ C-11)}$$

$$C_Z = C_{Z_0} + C_{Z_\alpha}\alpha + \frac{C_{Z_{\dot{\alpha}}}\dot{\alpha} l_{ref_x}}{2V_w} + \frac{C_{Z_p} p l_{ref_y}}{2V_w} + \frac{C_{Z_q} q l_{ref_x}}{2V_w} \quad \text{(EQ C-12)}$$

where α and β are the angle of attack and sideslip angle, respectively; p, q, and r are the components of the angular velocity vector about the body x, y, and z axes, respectively; l_{ref_x} and l_{ref_y} are reference lengths; and V_w is the magnitude of the wind-corrected velocity.

In this linearized aerodynamics approach, the aerodynamic force coefficients are assumed to be a function of Mach number only. However, the aerodynamic forces depend not only on the aerodynamic force coefficients, but also on the dynamic pressure, α, β, p, q, and r. The physical description of each of the aerodynamic force coefficients is:

C_{X_0}	Aerodynamic force coefficient along the x-axis for zero angle of attack and zero side slip.
C_{X_α}	Rate of change of the aerodynamic force coefficient along the x-axis with respect to the angle of attack.
C_{X_β}	Rate of change of the aerodynamic force coefficient along the x-axis with respect to the side slip angle.
C_{Y_0}	Aerodynamic force coefficient along the y-axis for zero side slip angle, zero rate of change in the side slip angle, and zero angular rates about the x and z axes.
C_{Y_β}	Rate of change of the aerodynamic force coefficient along the y-axis with respect to the angle of side slip.
$C_{Y_{\dot{\beta}}}$	Rate of change of the aerodynamic force coefficient along the y-axis with respect to the rate of change of angle of side slip.
C_{Y_p}	Rate of change of the aerodynamic force coefficient along the y-axis with respect to the angular rate about the x-axis.
C_{Y_r}	Rate of change of the aerodynamic force coefficient along the y-axis with respect to the angular rate about the z-axis.
C_{Z_0}	Aerodynamic force coefficient along the z-axis for zero angle of attack and zero side slip.
C_{Z_α}	Aerodynamic force coefficient along the z-axis for zero angle of attack, zero rate of change in the angle of attack, and zero angular rates about the x and y axes.
$C_{Z_{\dot{\alpha}}}$	Rate of change of the aerodynamic force coefficient along the z-axis with respect to the rate of change of the angle of attack.

C_{Z_p} Rate of change of the aerodynamic force coefficient along the z-axis with respect to the angular rate about the x-axis.

C_{Z_q} Rate of change of the aerodynamic force coefficient along the z-axis with respect to the angular rate about the y-axis.

In our analysis we used the aerodynamic force coefficients for the Improved Hawk missile given in Table C-5. Note that for the assumed symmetric missile, $C_{Z_\alpha} = C_{Y_\beta}$ and $C_{Z_q} = C_{Y_r}$.

Table C-5
Aerodynamic Force Coefficients and Derivatives[93]

Mach Number	C_X (thrust on)	C_X (thrust off)	C_{Y_β} (1/deg)	C_{Y_r} (1/deg)	C_{Z_α} (1/deg)	C_{Z_q} (1/deg)
0.00	-0.346	-0.483	-0.411	255.4	-0.411	255.4
0.40	-0.346	-0.483	-0.411	255.4	-0.411	255.4
0.70	-0.335	-0.470	-0.436	271.1	-0.436	271.1
0.90	-0.357	-0.489	-0.456	283.1	-0.456	283.1
0.95	-0.389	-0.544	-0.464	301.6	-0.464	301.6
1.00	-0.415	-0.598	-0.482	313.4	-0.482	313.4
1.05	-0.438	-0.648	-0.475	309.1	-0.475	309.1
1.10	-0.448	-0.651	-0.480	312.4	-0.480	312.4
1.20	-0.447	-0.640	-0.492	319.7	-0.492	319.7
1.56	-0.386	-0.561	-0.422	295.5	-0.422	295.5
1.96	-0.361	-0.511	-0.352	246.6	-0.352	246.6
2.44	-0.344	-0.504	-0.311	217.3	-0.311	217.3
5.00	-0.218	-0.274	-0.178	125.2	-0.178	125.2

All other force coefficients, i.e., C_{X_α}, C_{X_β}, C_{Y_0}, $C_{Y_{\dot\beta}}$, C_{Y_p}, C_{Z_0}, $C_{Z_{\dot\alpha}}$, and C_{Z_p}, were assumed to be zero.

Graphs of the aerodynamic force coefficients given in Table C-5 are shown in Figure C-4 through Figure C-6. Because of the symmetry between the C_Y coefficients and the C_Z coefficients, only graphs of the C_Y coefficients are provided.

As shown in Figure C-4, the axial force coefficient is smaller in magnitude when the motor is on. The exhaust plume reduces the base drag of the missile and results in a lower axial force.

The reference area and lengths used for the Improved Hawk missile were as follows:

$$S_{ref} = 1.0690 ft^2 \quad \text{(EQ C-13)}$$

$$l_{ref_x} = l_{ref_y} = 1.16667 ft \quad \text{(EQ C-14)}$$

We have set the longitudinal and transverse reference lengths equal to the diameter of the missile. It may seem unusal to use the diameter for the reference length in the longitudinal direction rather than the length of the vehicle. The approach we used (EQ C-14) allows the static margin of the vehicle to be expressed directly in diameters of the vehicle (or *calibers* in some usage); this is a common approach for long vehicles.

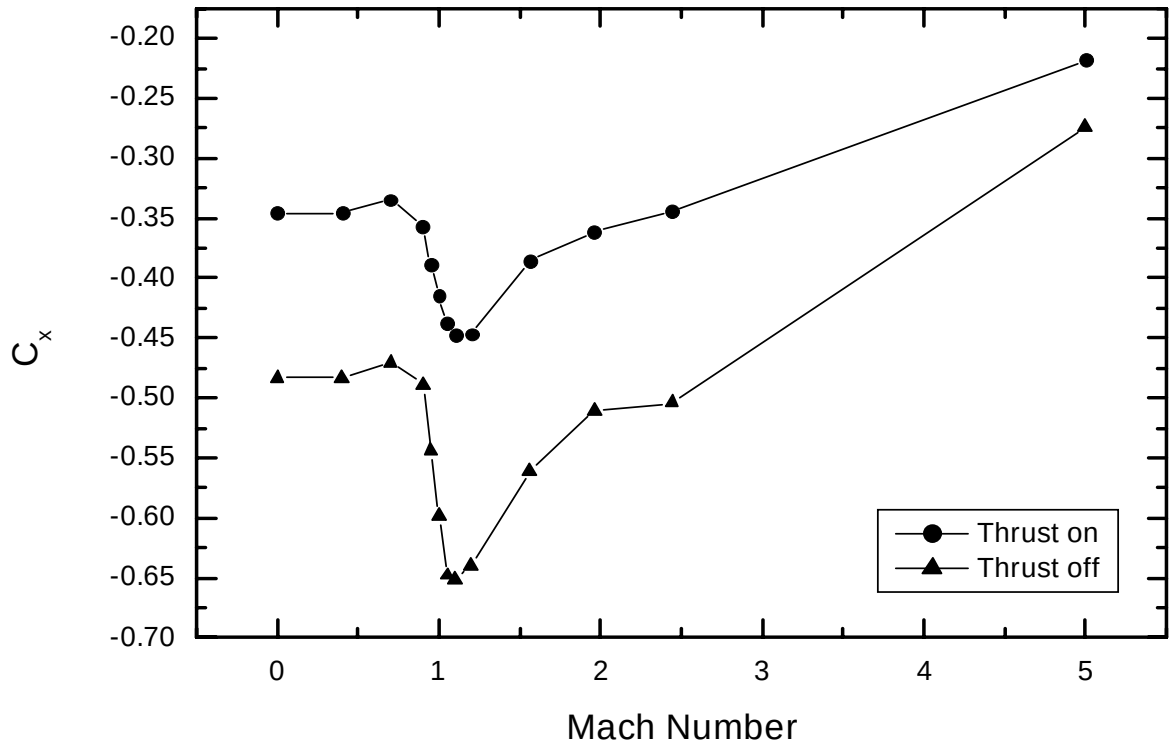

Figure C-4
Axial Force Coefficient

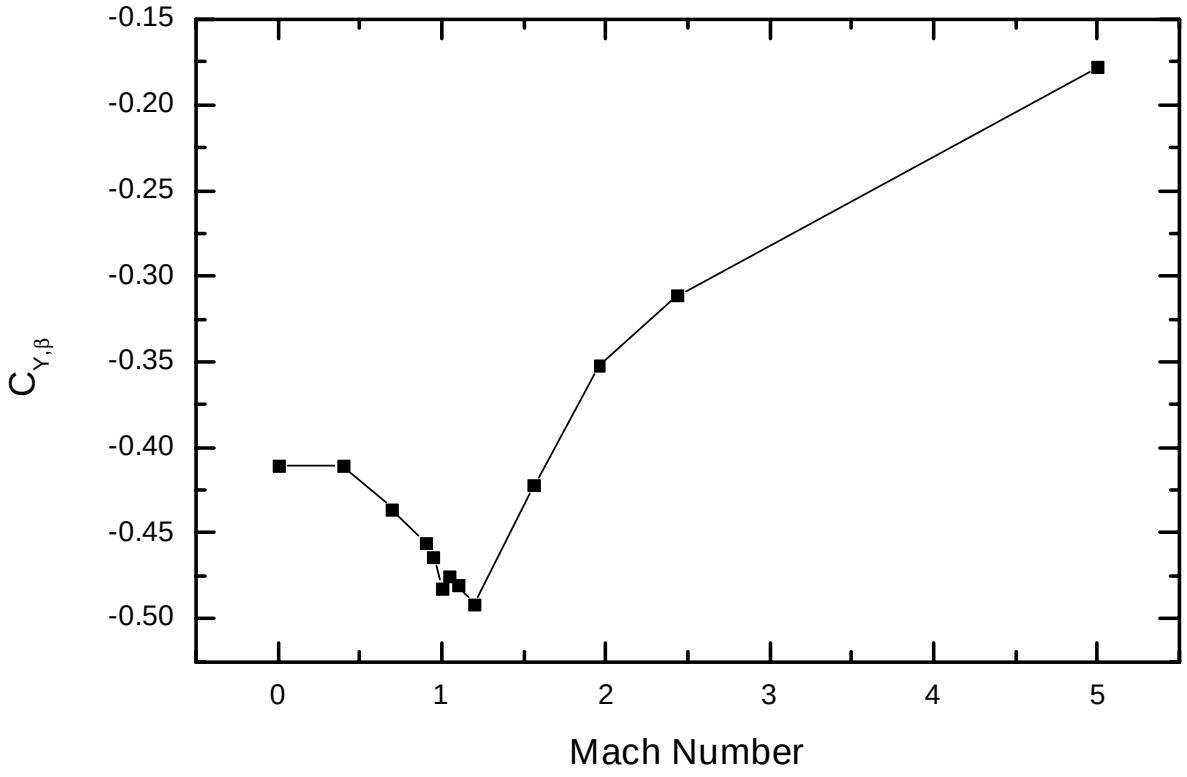

Figure C-5
C_{Y_β} as a Function of Mach Number.

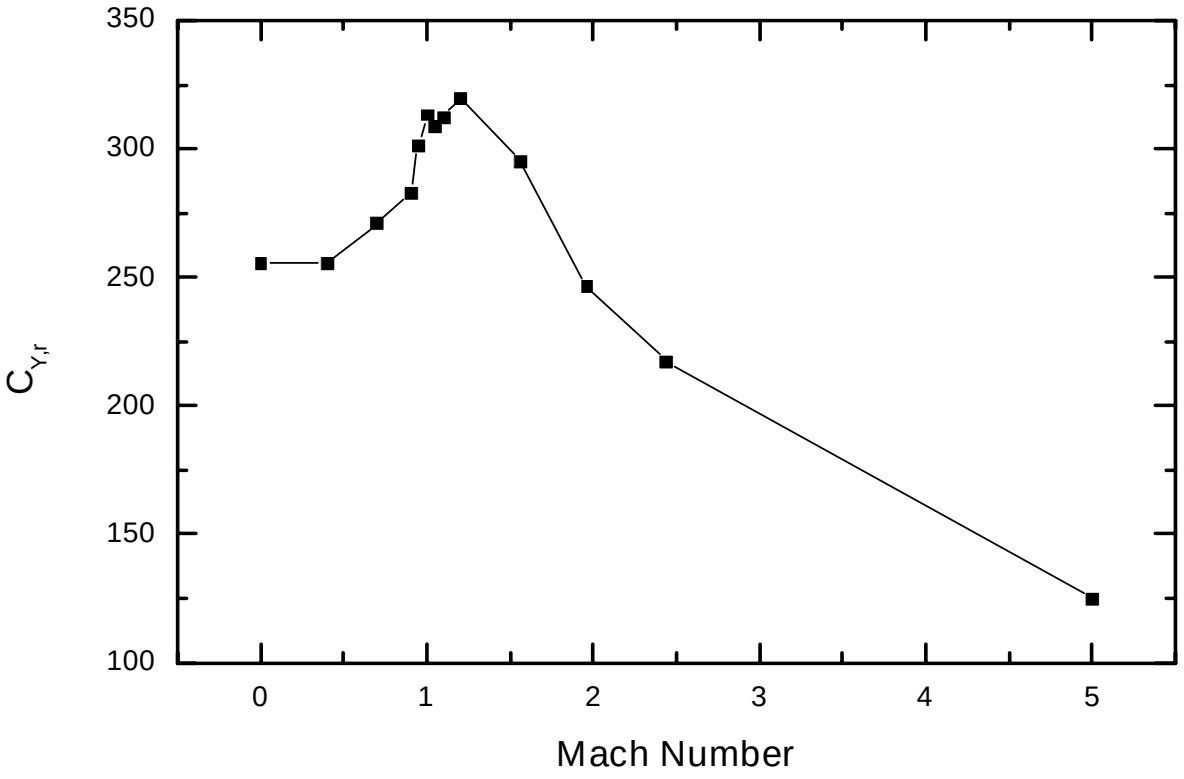

Figure C-6
C_{Y_r} as a Function of Mach Number.

C.6.2 Aerodynamic Moments

As is typical of fin-stabilized missiles, the example missile is statically stable. That is, when the missile is perturbed in angle of attack or angle of sideslip, then the angle of sideslip and angle of attack tend to return to zero.

For the 3DOF model, the aerodynamic moments are not defined. However, for the 6DOF rigid body trajectory, the aerodynamic forces act through the center of pressure which is normally different than the center of mass. The location of the aerodynamic force must then be provided in order to compute the aerodynamic moment. Since, in general, the center of pressure moves during flight, the location of the center of pressure must either be given as a function of flight conditions or an alternative approach using a fixed reference point and an aerodynamic moment vector can be used. We selected the latter approach for the calculations in this report.

The aerodynamic moment vector is given by:

$$\vec{M}_{aero} = \vec{C}_M q S_{ref} l_{ref_{x,y}} \quad \text{(EQ C-15)}$$

where $\vec{C}_M$ is the aerodynamic moment coefficient vector; $l_{ref_{x,y}}$ is the appropriate reference length for the longitudinal and horizontal axes; and q and S_{ref} were defined earlier. As noted in the last section, we have chosen to set $l_{ref_x} = l_{ref_y} = 1.16667 ft$.

In TAOS, the total aerodynamic moment is computed in the body coordinate system and is given by:

$$M_x = C_l q S_{ref} l_{ref_y} + C_Z q S_{ref} (y_{ref} - y_{cg}) - C_Y q S_{ref} (z_{ref} - z_{cg}) \quad \text{(EQ C-16)}$$

$$M_y = C_m q S_{ref} l_{ref_x} + C_X q S_{ref} (z_{ref} - z_{cg}) - C_Z q S_{ref} (x_{ref} - x_{cg}) \quad \text{(EQ C-17)}$$

$$M_z = C_n q S_{ref} l_{ref_y} + C_Y q S_{ref} (x_{ref} - x_{cg}) - C_X q S_{ref} (y_{ref} - z_{cg}) \quad \text{(EQ C-18)}$$

where C_l, C_m, and C_n are the aerodynamic coefficients about the x, y, and z body axes, respectively; q, S_{ref}, l_{ref_x}, l_{ref_y}, C_X, C_Y, and C_Z were previously defined; (x_{ref}, y_{ref}, z_{ref}) is the aerodynamic reference point, and (x_{cg}, y_{cg}, z_{cg}) is the center of mass. The aerodynamic reference point used for this missile was located at the nose of the missile.

The aerodynamic moment coefficients can be linearized just as the aerodynamic force coefficients were to yield:

$$C_l = C_{l_0} + C_{l_\delta}\delta + C_{l_\alpha}\alpha + C_{l_\beta}\beta + \frac{C_{l_p} p l_{ref_y}}{2V_w} + \frac{C_{l_r} r l_{ref_y}}{2V_w} \quad \text{(EQ C-19)}$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + \frac{C_{m_{\dot{\alpha}}} \dot{\alpha} l_{ref_x}}{2V_w} + \frac{C_{m_p} p l_{ref_y}}{2V_w} + \frac{C_{m_q} q l_{ref_x}}{2V_w} \quad \text{(EQ C-20)}$$

$$C_n = C_{n_0} + C_{n_\beta}\beta + \frac{C_{n_{\dot{\beta}}} \dot{\beta} l_{ref_x}}{2V_w} + \frac{C_{n_p} p l_{ref_y}}{2V_w} + \frac{C_{n_r} r l_{ref_x}}{2V_w} \quad \text{(EQ C-21)}$$

where δ is the fin-cant angle, and α, β, p, q, r, l_{ref_x}, l_{ref_y}, and V_w are as defined earlier. In a fin-stabilized missile, the fins are angled in order to induce a roll about the x-axis of the missile. The second term in (EQ C-19) is the roll moment coefficient due to the fins.

The physical description of each of the aerodynamic moment coefficients is:

C_{l_0}	Roll moment coefficient for zero angle of attack, zero side slip, zero fin-cant angle, and zero angular rates about the x and z axes.
C_{l_δ}	Rate of change of the roll moment coefficient with respect to the cant of the fins.
C_{l_α}	Rate of change of the roll moment coefficient with respect to the angle of attack.
C_{l_β}	Rate of change of the roll moment coefficient with respect to the side slip angle.
C_{l_p}	Rate of change of the roll moment coefficient with respect to the angular rate about the x-axis.
C_{l_r}	Rate of change of the roll moment coefficient with respect to the angular rate about the z-axis.
C_{m_0}	Pitching moment coefficient for zero angle of attack, zero rate of change of angular attack, and zero angular rates about the x and y axes.
C_{m_α}	Rate of change of the pitching moment coefficient with respect to the angle of attack.
$C_{m_{\dot{\alpha}}}$	Rate of change of the pitching moment coefficient with respect to the rate of change in the angle of attack
C_{m_p}	Rate of change of the pitching moment coefficient with respect to the angular rate about the x-axis
C_{m_q}	Rate of change of the pitching moment coefficient with respect to the angular rate about the y-axis
C_{n_0}	Yawing moment coefficient for zero side slip angle, zero rate of side slip angle, and zero angular rates about the x and z axes.
C_{n_β}	Rate of change of the yawing moment coefficient with respect to the side slip angle.
$C_{n_{\dot{\beta}}}$	Rate of change of the yawing moment coefficient with respect to the rate of change in the side slip angle
C_{n_p}	Rate of change of the yawing moment coefficient with respect to the angular rate about the x-axis
C_{n_r}	Rate of change of the yawing moment coefficient with respect to the angular rate about the z-axis

In practice, it is difficult to separate $C_{m_{\dot{\alpha}}}$ from C_{m_q} in measurements of the aerodynamic moment coefficients. As a result, it is common practice to model the effects from these two terms

using one term or the other, but not both. This effectively combines the two terms into a single term that depends on either $\dot{\alpha}$ or q, since $\dot{\alpha}$ and q are similar, but not the same. In our analysis, we selected C_{m_q} and set the $C_{m_{\dot{\alpha}}}$ term to zero. A similar difficulty arises in trying to separate $C_{n_{\dot{\beta}}}$ from C_{n_r}. In our analysis we used C_{n_r} and set the $C_{n_{\dot{\beta}}}$ term to zero.

The aerodynamic moment coefficients for the Improved Hawk missile used in our analysis are given in Table C-6 and Table C-7. Note that for the assumed symmetric missile, $C_{m_\alpha} = -C_{n_\beta}$ and $C_{m_q} = C_{n_r}$.

Table C-6
Linearized Aerodynamic Moment Coefficient Derivatives for C_l[93]

Mach Number	C_{l_δ} (1/deg)	C_{l_p} (1/deg)	Mach Number	C_{l_δ} (1/deg)	C_{l_p} (1/deg)
0.00	0.3244	-45.69	1.80	0.4705	-66.32
0.50	0.3244	-45.69	1.90	0.4705	-66.69
0.75	0.3369	-47.53	2.00	0.4670	-66.32
0.95	0.3493	-49.00	2.25	0.4563	-63.74
1.00	0.3529	-49.74	2.50	0.4367	-60.79
1.05	0.3547	-50.11	2.75	0.4135	-57.85
1.10	0.3529	-49.74	3.00	0.3903	-54.90
1.15	0.3386	-49.37	3.25	0.3672	-52.32
1.20	0.2709	-44.21	3.50	0.3493	-49.37
1.25	0.2852	-38.69	3.75	0.3351	-47.16
1.30	0.3351	-44.21	4.00	0.3208	-44.95
1.40	0.3957	-56.00	4.50	0.2923	-41.27
1.50	0.4367	-61.90	5.00	0.2691	-37.95
1.60	0.4563	-64.48	5.50	0.2495	-36.48
1.70	0.4670	-65.95			

Table C-7
Linearized Aerodynamic Moment Coefficients for C_m and C_n[93]

Mach Number	C_{m_α} (1/deg)	C_{m_q} (1/deg)	C_{n_β} (1/deg)	C_{n_r} (1/deg)
0.00	-4.953	-1385.3	4.953	-1385.3
0.40	-4.953	-1385.3	4.953	-1385.3
0.70	-5.281	-1470.2	5.281	-1470.2
0.90	-5.542	-1535.6	5.542	-1535.6
0.95	-5.657	-1711.6	5.657	-1711.6
1.00	-5.882	-1778.8	5.882	-1778.8
1.05	-5.847	-1754.4	5.847	-1754.4
1.10	-5.928	-1772.9	5.928	-1772.9
1.20	-5.680	-1814.2	5.680	-1814.2
1.56	-4.817	-1804.3	4.817	-1804.3
1.96	-3.967	-1505.3	3.967	-1505.3
2.44	-3.432	-1326.8	3.432	-1326.8
5.00	-1.889	-767.6	1.889	-767.6

The fins are canted at a 0.25 degree angle so C_{l_δ} is multiplied by 0.25 before use as shown in (EQ C-20). The fin cant causes roll moment which results in a roll rate directly related to the speed of the missile.

All other moment coefficients, i.e., C_{l_α}, C_{l_β}, C_{l_r}, C_{m_0}, C_{m_p}, $C_{m_{\dot\alpha}}$, C_{n_0}, $C_{n_{\dot\beta}}$, and C_{n_p}, were assumed to be zero. Recall from the earlier discussion that the effects of $C_{m_{\dot\alpha}}$ are included in C_{m_q} while the effects of $C_{n_{\dot\beta}}$ are included in C_{n_r}.

Graphs of the aerodynamic moment coefficients given in Table C-6 and Table C-7 are shown in Figure C-7 through Figure C-10. Because of the symmetry between the C_m coefficients and the C_n coefficients, only graphs of the C_m coefficients are provided.

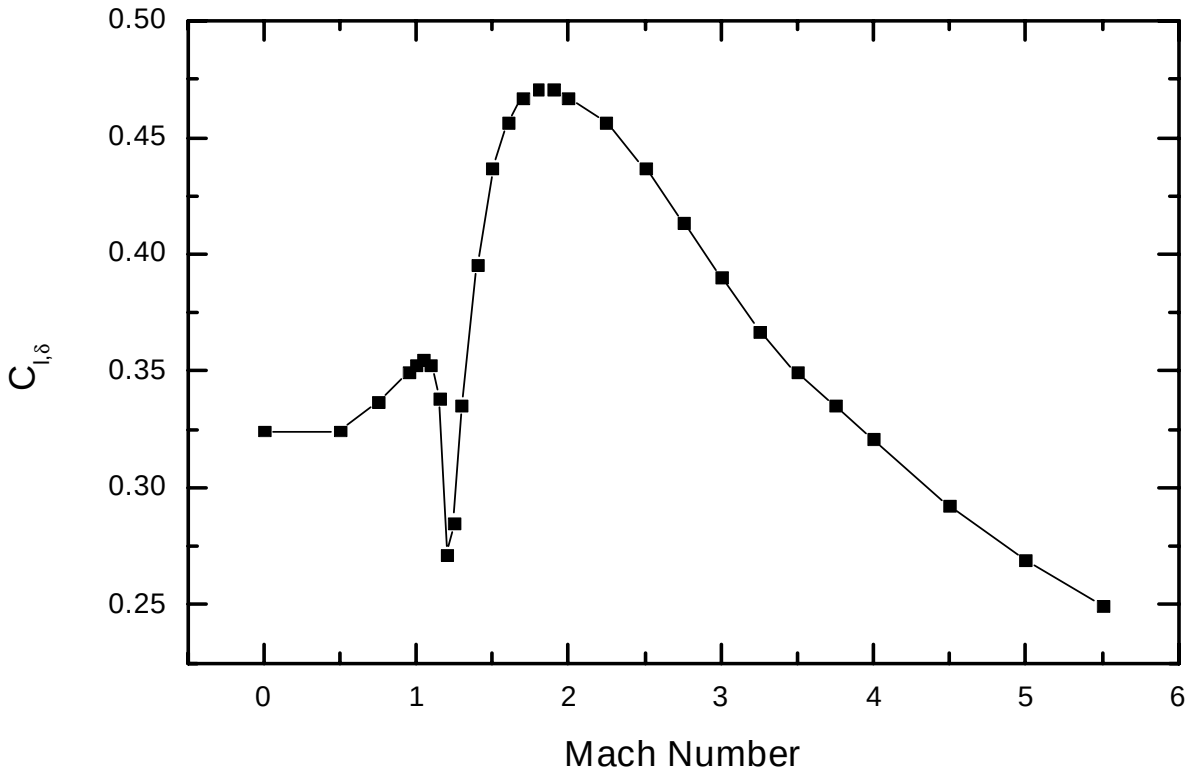

Figure C-7
C_{l_δ} as a Function of Mach Number.

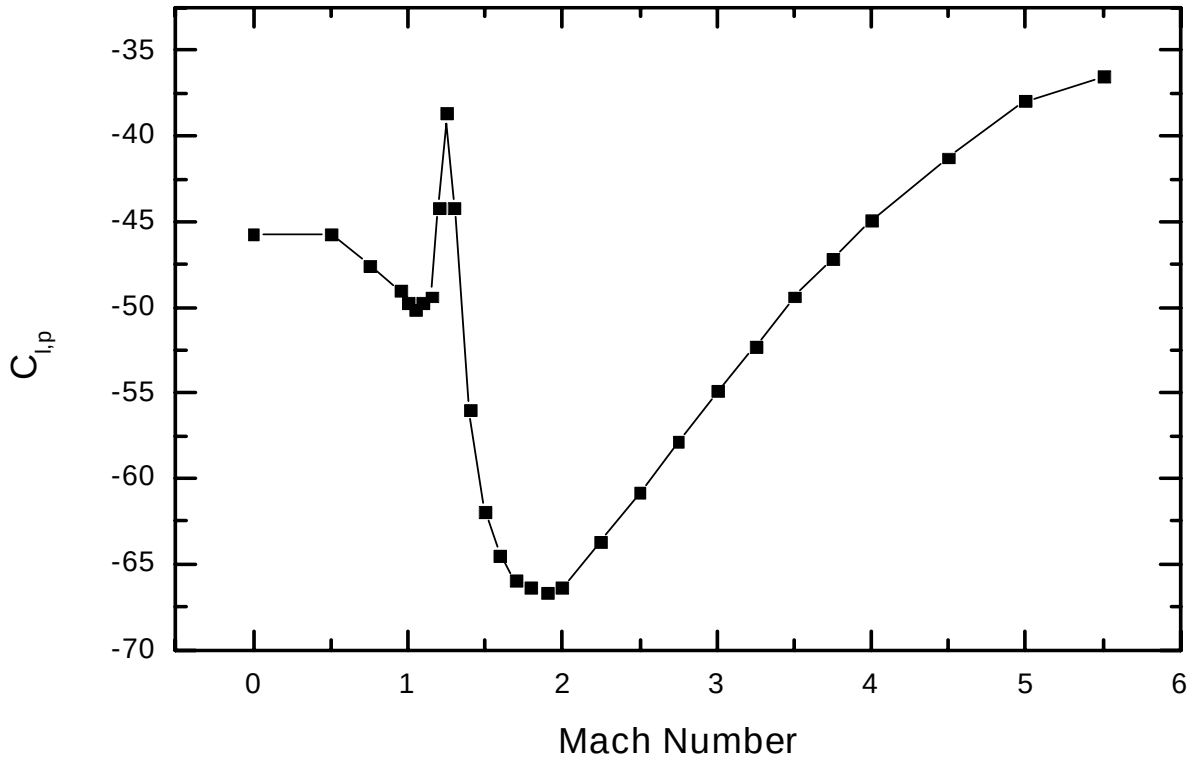

Figure C-8
C_{l_p} as a Function of Mach Number.

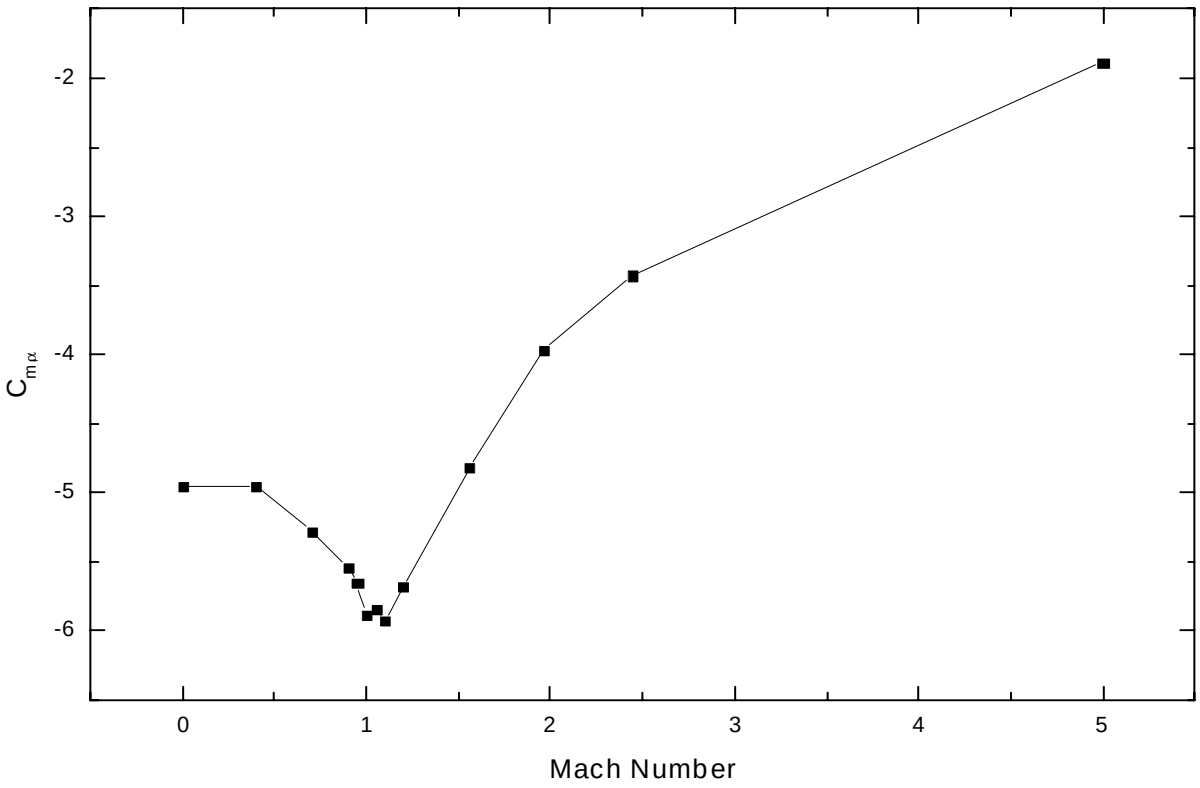

Figure C-9
C_{m_α} as a Function of Mach Number.

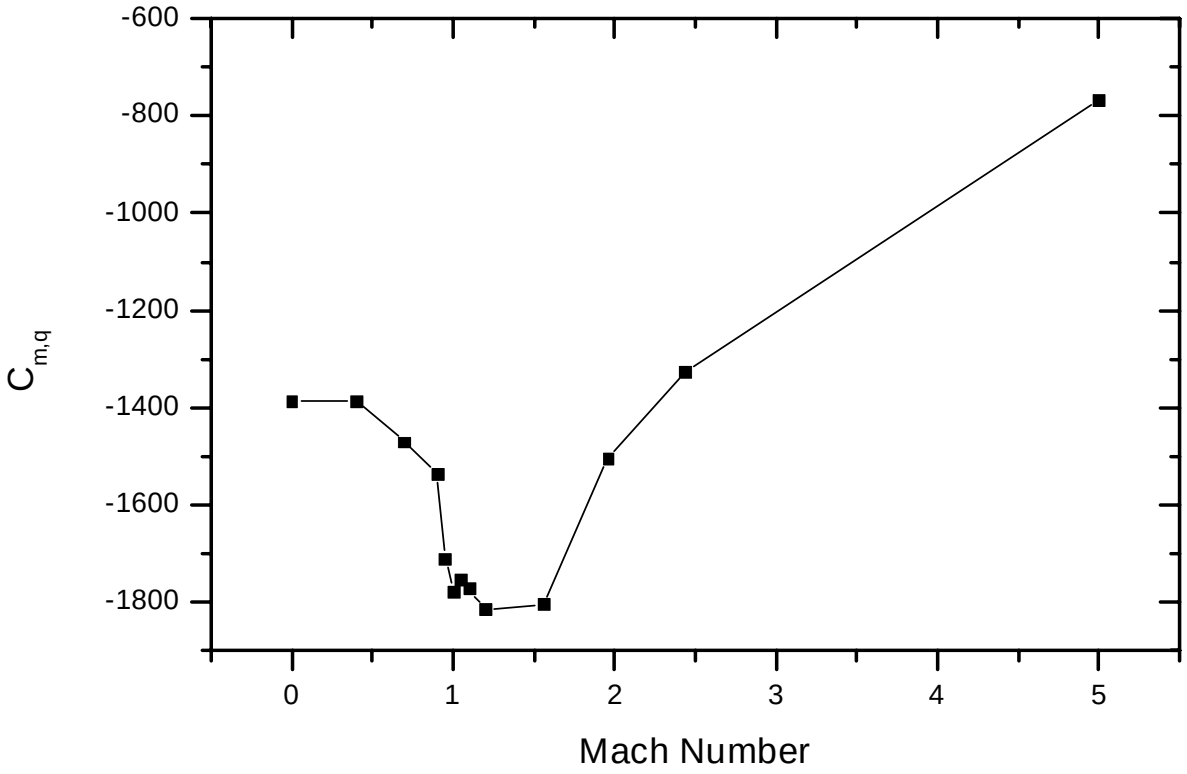

Figure C-10
C_{m_q} as a Function of Mach Number.

Appendix D: Detailed Discussion of Trajectories

D.1 Brief Description of the Problem

As described in detail in Appendix C, the flight dynamics example concerned the unguided flight of an air-launched missile. The missile is launched at 30,000 feet from an aircraft traveling straight and level at a speed of 700 ft./s. The missile has fins for flight stability and the generation of a roll rate, but no other lifting surfaces. The initial mass of the missile is 42.86 slugs and the thrust profile reflected a nominal motor temperature. In this section, we compare details of the trajectory for the 3DOF and 6DOF models for a single nominal trajectory. We refer to these two trajectories as the "baseline" trajectories.

The key difference between the 3DOF model and the 6DOF model is that the former only accounts for translational motion of the center-of-mass of the missile while the latter accounts for both translational and rotational motion. As we shall see, effects of the rotational motion are apparent in many of the output variables.

D.2 A Word About the Generation of the Plots

The graphs illustrating the detailed trajectories were generated using the results of TAOS runs for both the 3DOF and 6DOF mathematical models. The 3DOF and 6DOF computations used identical print intervals of 0.005 seconds and identical per-step truncation error criteria of $E_t = 1\times10^{-12}$. These baseline trajectories are not identical to the trajectories discussed in the body of this report, even though the same input parameters and initial conditions were used for the calculations in the body of the report and those reported in this Appendix. The reason for this, as discussed in Appendix B, is that the print interval needed here to illustrate the detailed motion of the missile affects the numerical integration very slightly.

For each output variable discussed below, we graph the result for the 6DOF model. If the result for the 3DOF model is clearly distinct from that of the 6DOF model, we show both the 6DOF and 3DOF results on the same axis. If the 6DOF and 3DOF results are not distinguishable when plotted on the same graph, we plot the 6DOF result on one graph and plot the difference between the 6DOF and 3DOF results on a second graph. Since the total flight time is different between the two models, the difference is computed through the last point in time in common between the two trajectories.

D.3 Position as a Function of Time

Although calculations were performed in the Earth-centered, fixed coordinate (ECFC) system described in Appendix A, we choose to present the position of the missile using coordinates referenced to the Earth's surface. The origin for this coordinate system is the launch point of the missile projected on to the Earth's surface. For our flight dynamics example, this point is the intersection of the Equator and the Greenwich meridian. As shown in Figure D-1, the "North" axis is aligned with lines of constant longitude and the "East" axis is aligned with lines of constant latitude. The third axis points away from the center of the Earth and is the altitude of the missile.

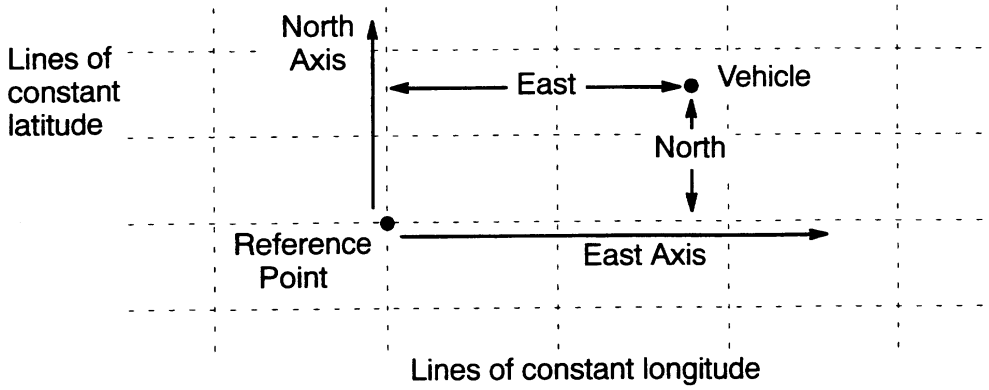

Figure D-1
East and North Distances[91]

The altitude, shown in Figure D-2, decreases smoothly from the initial launch altitude of 30,000 ft. The difference plot shows that the altitude is always higher for the 6DOF model than the 3DOF model at the same time from launch. As noted above, the 6DOF model accounts for angular motion of the missile and includes the orientation of the missile and appropriate moments of inertia. As we shall see below, the x-axis of the missile is not aligned with the velocity vector in the 6DOF model but exhibits a coupled interaction between the angle of attack, angle of side slip, roll angle, and the angular rates of each of these quantities. There is a small mean angle of attack, referred to as "yaw of repose" in flight dynamics, due to the curved trajectory of the missile. The small angle of attack in the plane of the motion generates a small lift and the missile flies further in the 6DOF model than the 3DOF model.

The North position, shown in Figure D-3, increases smoothly as a function of time. The missile flies nearly 20 nm north before the trajectory ends. The graph of the difference between the 6DOF and 3DOF results shows that the missile in the 6DOF simulation has flown approximately 91 feet further north than that in the 3DOF simulation at the time the missile in the 3DOF simulation impacts the ground. The missile in the 6DOF simulation actually impacts the ground nearly 244 feet further north because it flies for an additional 0.116 seconds.

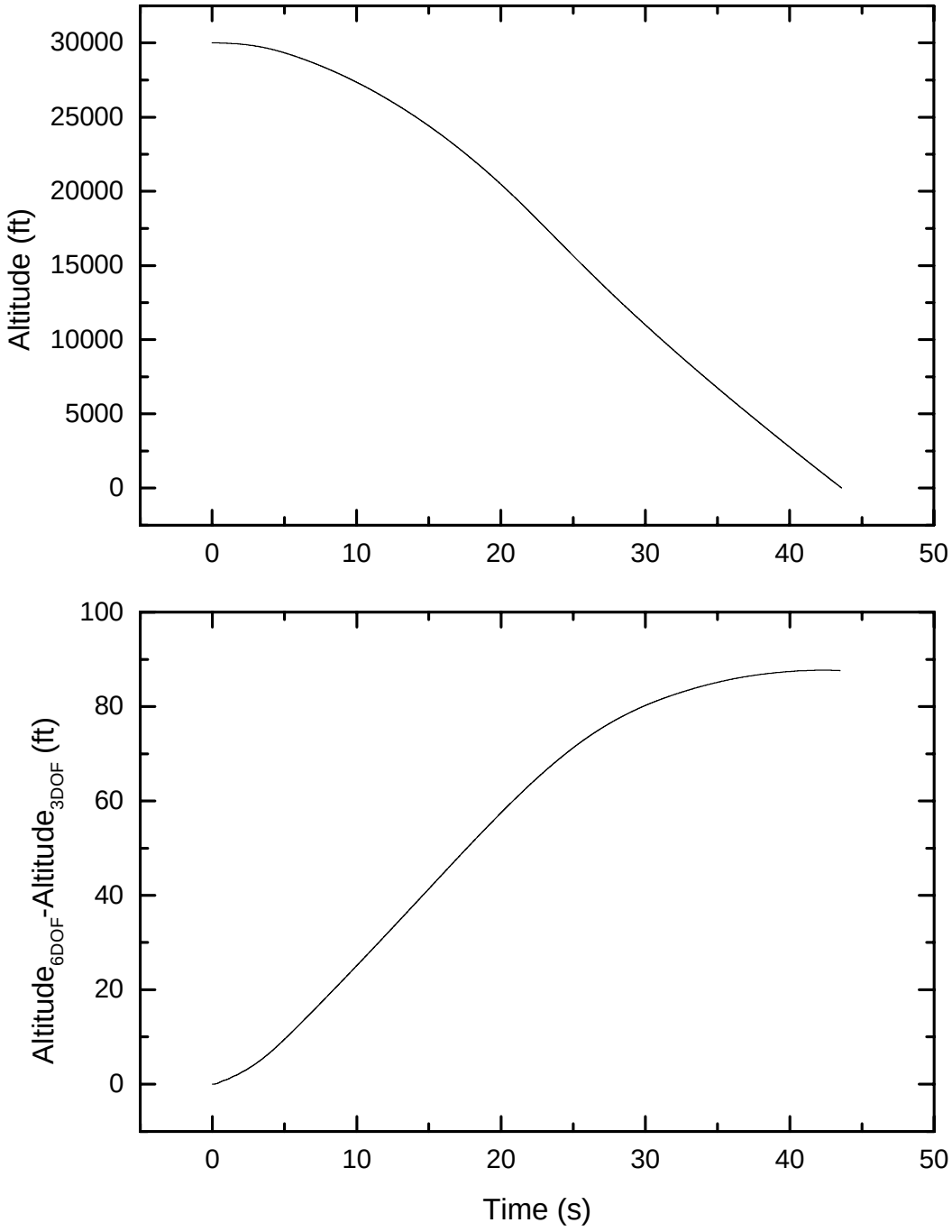

Figure D-2
Altitude as a Function of Time.

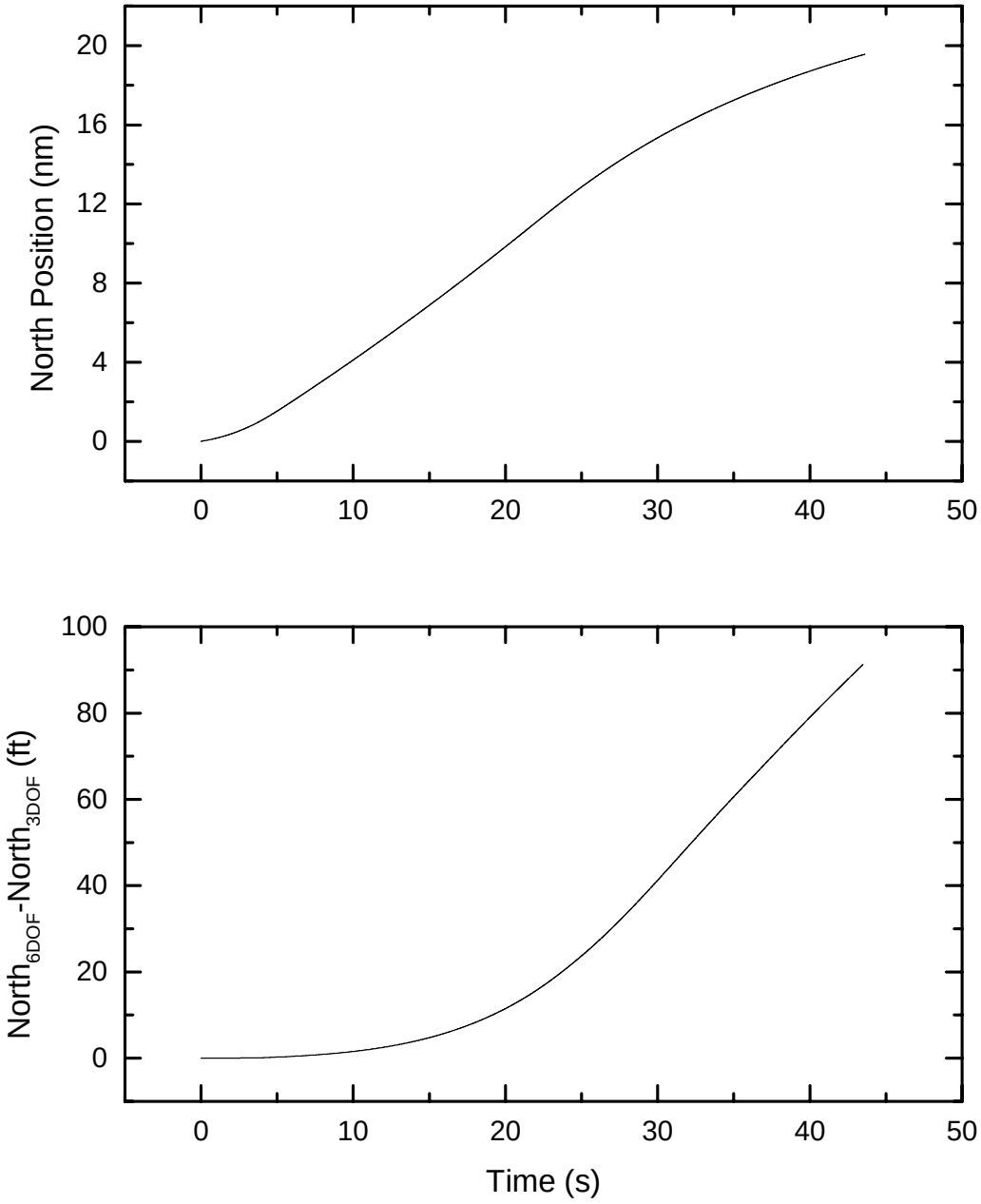

Figure D-3
North Position as a Function of Time

The East position, shown in Figure D-4, is dramatically different for the two models. The 3DOF model shows no deflection of the trajectory from the longitudinal plane, while the 6DOF models shows a deflection of a few feet. Again the rotational motion of the missile is key in explaining this difference. As we shall see below, the missile rolls clockwise about its x-axis (looking toward the nose of the missile). The pitching of the missile about the (ECFC) y-axis changes the direction of the angular momentum vector. The trajectory of the missile is deflected out of the pitch plane to the right (looking in the direction of flight) due to conservation of total angular momentum.

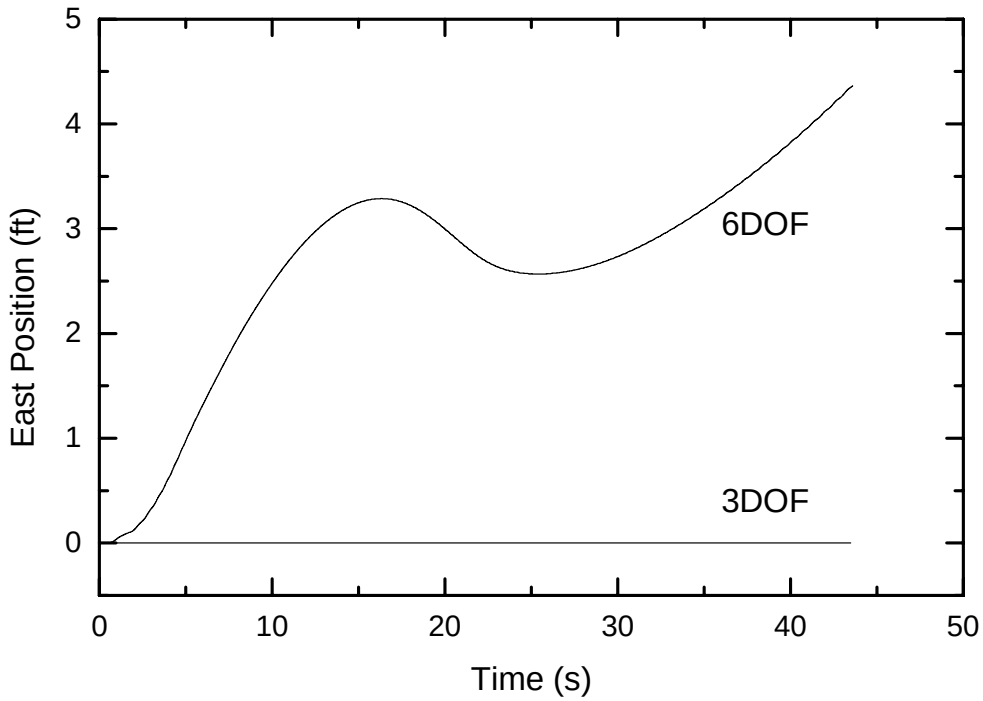

Figure D-4
East Position as a Function of Time

D.4 Velocity and Mach Number as a Function of Time

Now consider Figure D-5, a graph of the magnitude of the velocity vs. time. The velocity rises very rapidly during the first 5 seconds of the trajectory corresponding to the high initial thrust of the motor. The velocity rises more slowly between 5 seconds and 22.5 seconds corresponding to the lower average thrust during this time. There is a very small discontinuity in slope visible at 24.9 seconds. This slope discontinuity can be seen more clearly in the graph of the difference in velocity between the 6DOF and 3DOF models. The difference plot shows additional discontinuities in slope at approximately 31 seconds, 35 seconds, and 40 seconds.

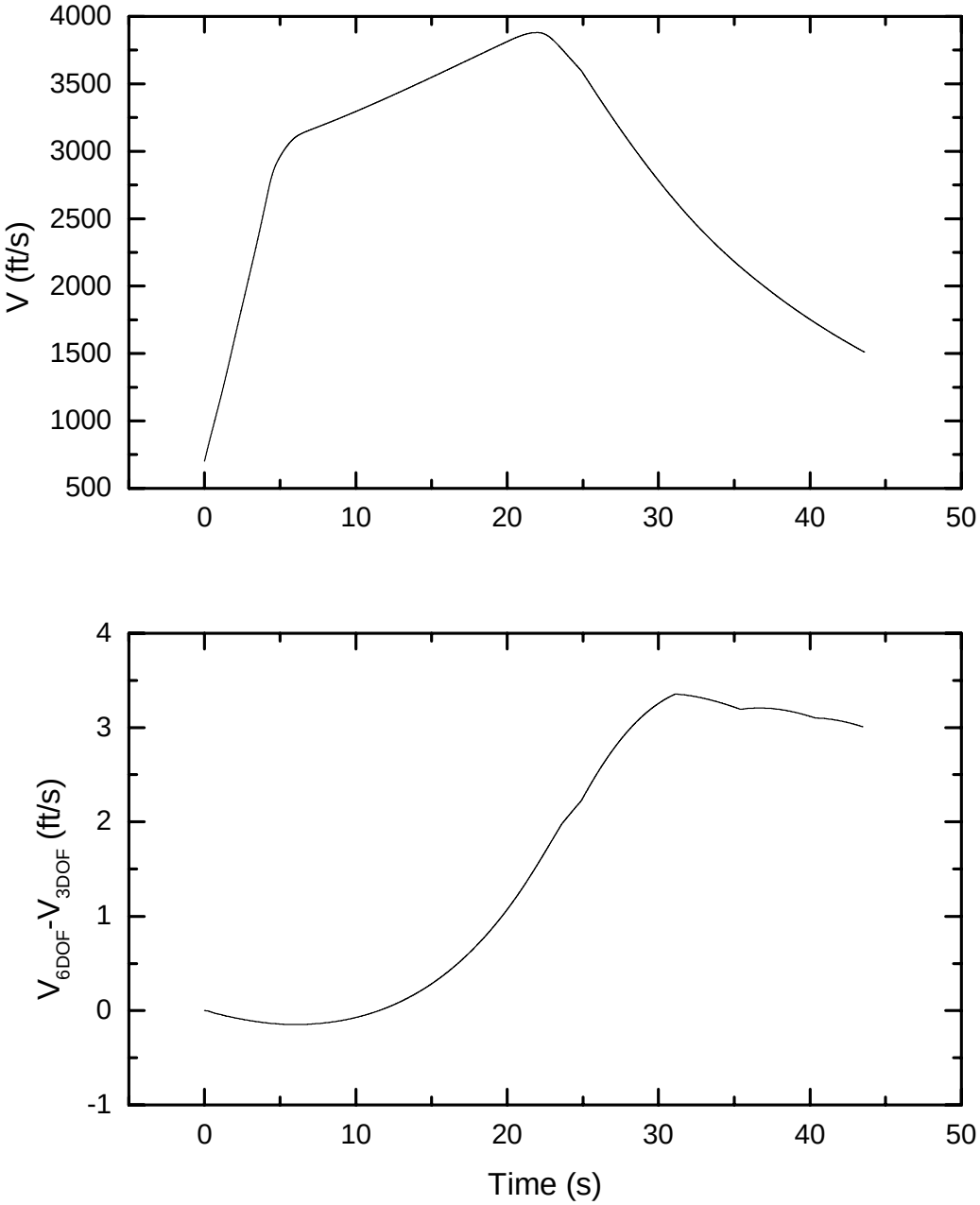

Figure D-5
Velocity as a Function of Time

The Mach number, shown in Figure D-6, shows similar features. Recall that the Mach number is the ratio of the missile velocity to the speed of sound at the altitude of the missile. The peak Mach number attained is 3.73 and the impact Mach number is 1.35. The source of the slope discontinuities will be discussed in Section D.6 below.

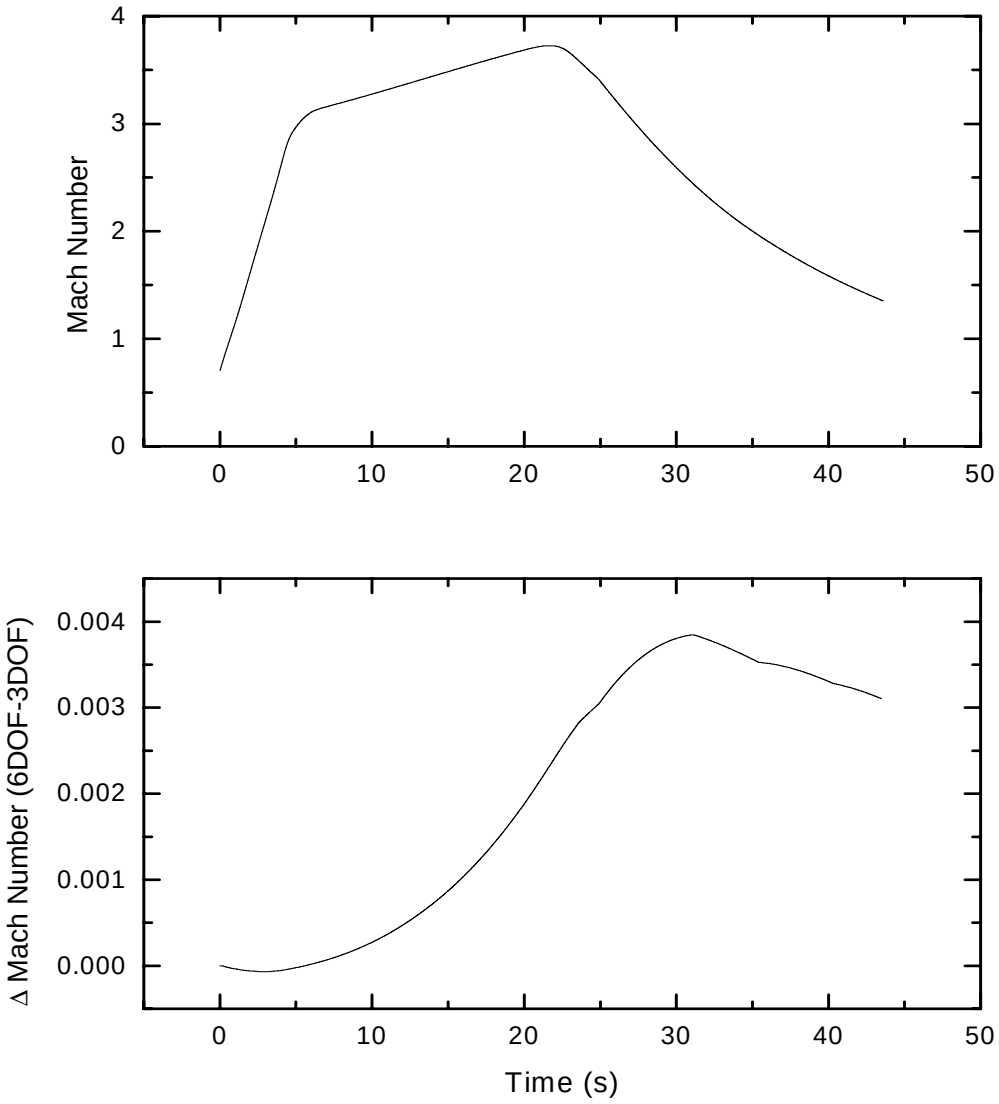

Figure D-6
Mach Number as a Function of Time

D.5 Thrust, Mass Flow Rate, and Weight

The thrust profile for a motor at nominal operating temperature is shown in Figure D-7. During the first 0.001 seconds, the thrust increases very sharply and then varies slightly near a nominal value of 19,000 lbs. The thrust then drops rapidly to a value near 3,000 lbs. and remains at this lower level until finally tapering off between 22 and 24.9 seconds. The difference plot shows that the thrust is slightly greater for the 6DOF model than the 3DOF model, with the difference rising smoothly from 0 at $t = 0$ until the thrust cut-off at 24.9 s. The slight difference in thrust (less than 2 lbs.) is due to the slight difference in altitude between the two trajectories we saw in Figure D-2. Recall that the thrust is given by

$$thrust = t_{vac} - 0.5248p \qquad \text{(EQ 0-1)}$$

where t_{vac} is the static thrust in vacuum and p is the atmospheric pressure. The atmospheric pressure is a function of altitude and increases as altitude decreases. As we saw above, the missile has a small lift in the 6DOF model that causes it to fly farther than it does in the 3DOF model. As a result, its altitude at any given time is slightly higher than that in the 3DOF model and therefore the thrust is larger.

The thrust is associated with a mass flow rate of the motor as the fuel is burned. The mass flow rate is given as a function of time as an input to the problem (see Section C.3). The graph shown in Figure D-8 shows that there is no difference in mass flow rate between the 6DOF and 3DOF models, as expected. The mass flow rate is roughly proportional to the thrust of the motor, as can be seen by the inverted profile in Figure D-8 as compared with the thrust in Figure D-7.

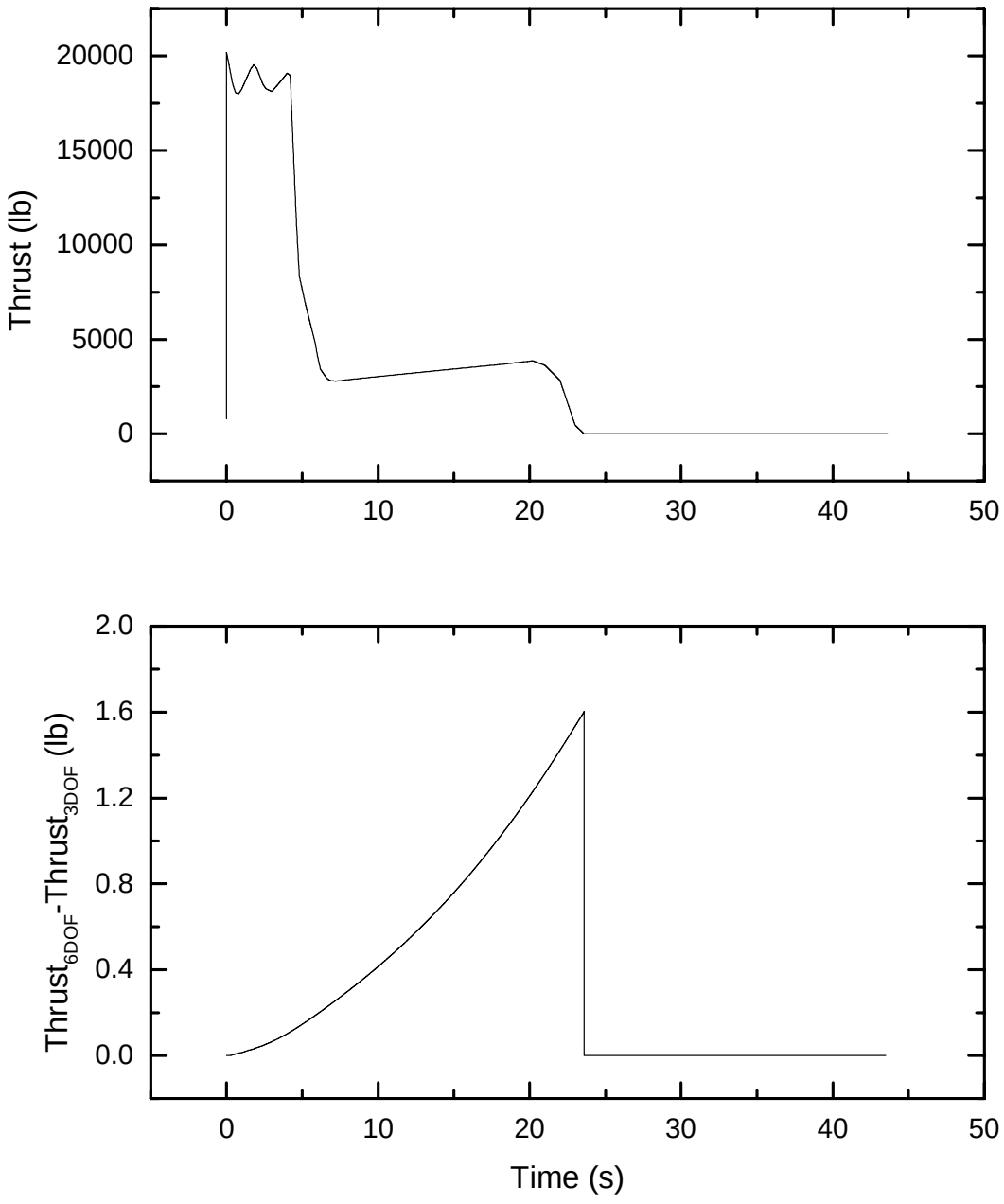

Figure D-7
Thrust as a Function of Time

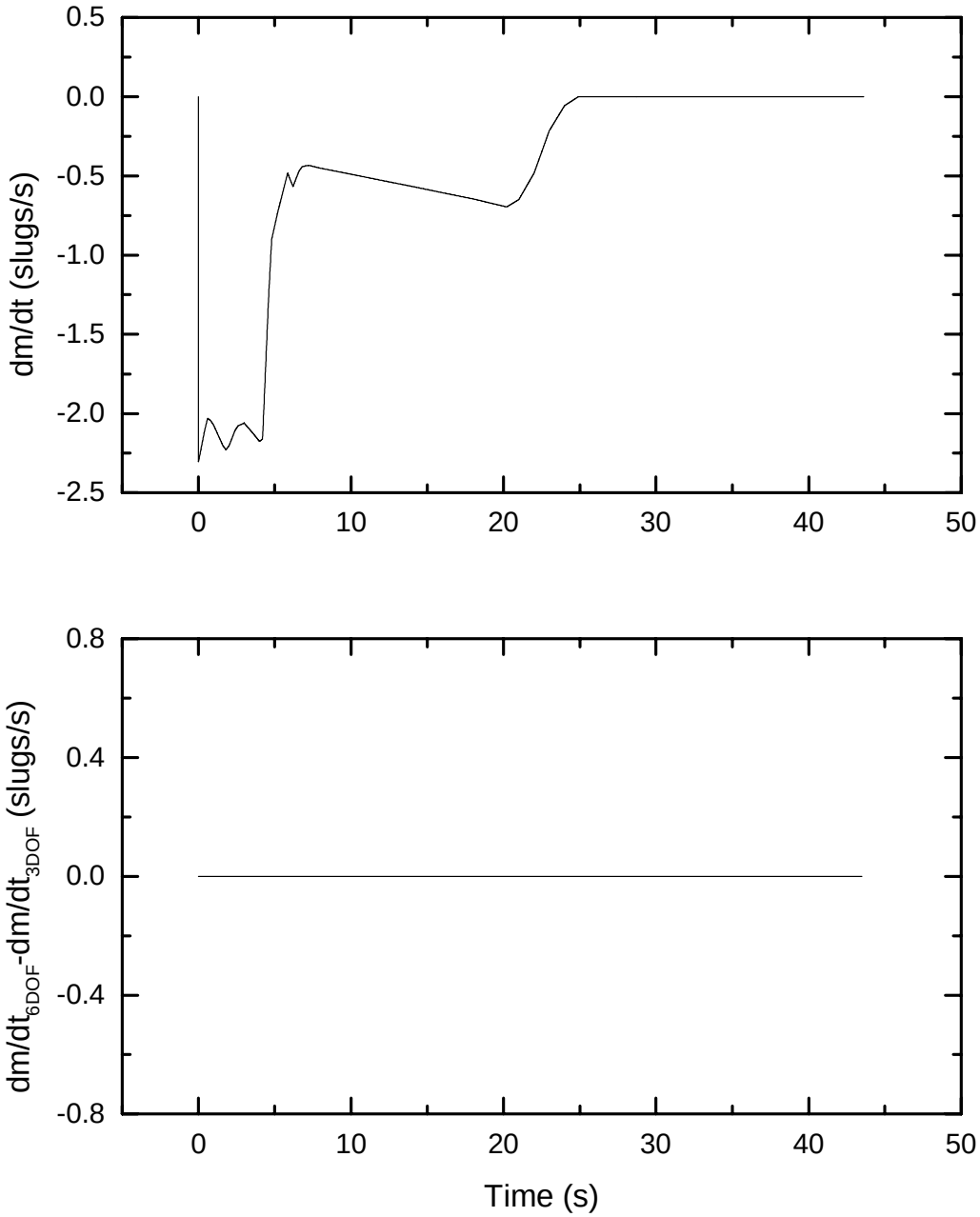

Figure D-8
Mass Flow Rate as a Function of Time

Figure D-9 shows the weight of the vehicle during the trajectory. The mass decreases at distinctly different rates during the motor operation: a high rate of decrease during the high thrust initial pulse, and a lower rate of decrease during the sustained pulse. Interestingly, there is a slight difference in the weight of the vehicle between the 6DOF and 3DOF models that occurs almost immediately in the trajectory, but does not change after that. Since both the initial mass and mass flow rate are the same in the two models it is surprising that there is a difference in the computed weight.

We investigated this slight discrepancy between the two models and determined that it is an effect of the numerical integration error. The mass flow rate drops sharply from 0 at $t = 0$ to -2.306 slugs/s at $t = 0.001\,s$ and then changes slightly to -2.208 slugs/s at $t = 0.002\,s$. (See Table C-3 in Appendix C.) This can be viewed as a discontinuity in the derivative of the mass flow rate at $t = 0.001\,s$. As we saw in Section B.2, Runge-Kutta integration estimates the value of a function using evaluations of its derivative at selected points in the time step interval. Slight differences in the time step result in the use of different points for the derivative evaluations. The use of different evaluation points in turn leads to slightly different estimates for the value of the function. In particular, the estimate of the integral across a large discontinuity in derivative will be especially sensitive to the time step used. In the calculations for the flight dynamics example, the 6DOF and 3DOF models use different time steps to cross 0.001 s. due to the differences in the physics captured by the models. The 6DOF and 3DOF simulations thus compute slightly different estimates for the mass after integrating across the discontinuity in the derivative of the mass flow rate at 0.001 s. The difference in mass is reflected in the difference in weight between the two computations.

The magnitude of the difference can be related to the per-step relative error criterion. Each integration step must satisfy this error criterion. In the worst case, one simulation will overestimate the integral across the discontinuity in the derivative of the mass flow rate by the allowed error while the other simulation will underestimate the value of the integral by the same amount. Therefore, the magnitude of the difference due to integrating across the discontinuity in the derivative of the mass flow rate is limited to twice the relative error criterion. The initial weight of the missile is 1379 lbs. The observed difference in weight should thus be less than $2 \times 1.0 \times 10^{-12} \times 1379 = 2.758 \times 10^{-9}$. The difference shown in the lower graph in Figure D-9 is $1.2 \times 10^{-9} lbs$ and is indeed consistent with a numerical integration error.

The mass flow rate does not have any other discontinuities of this magnitude in any other part of the trajectory and so there is essentially no additional accumulation of error.

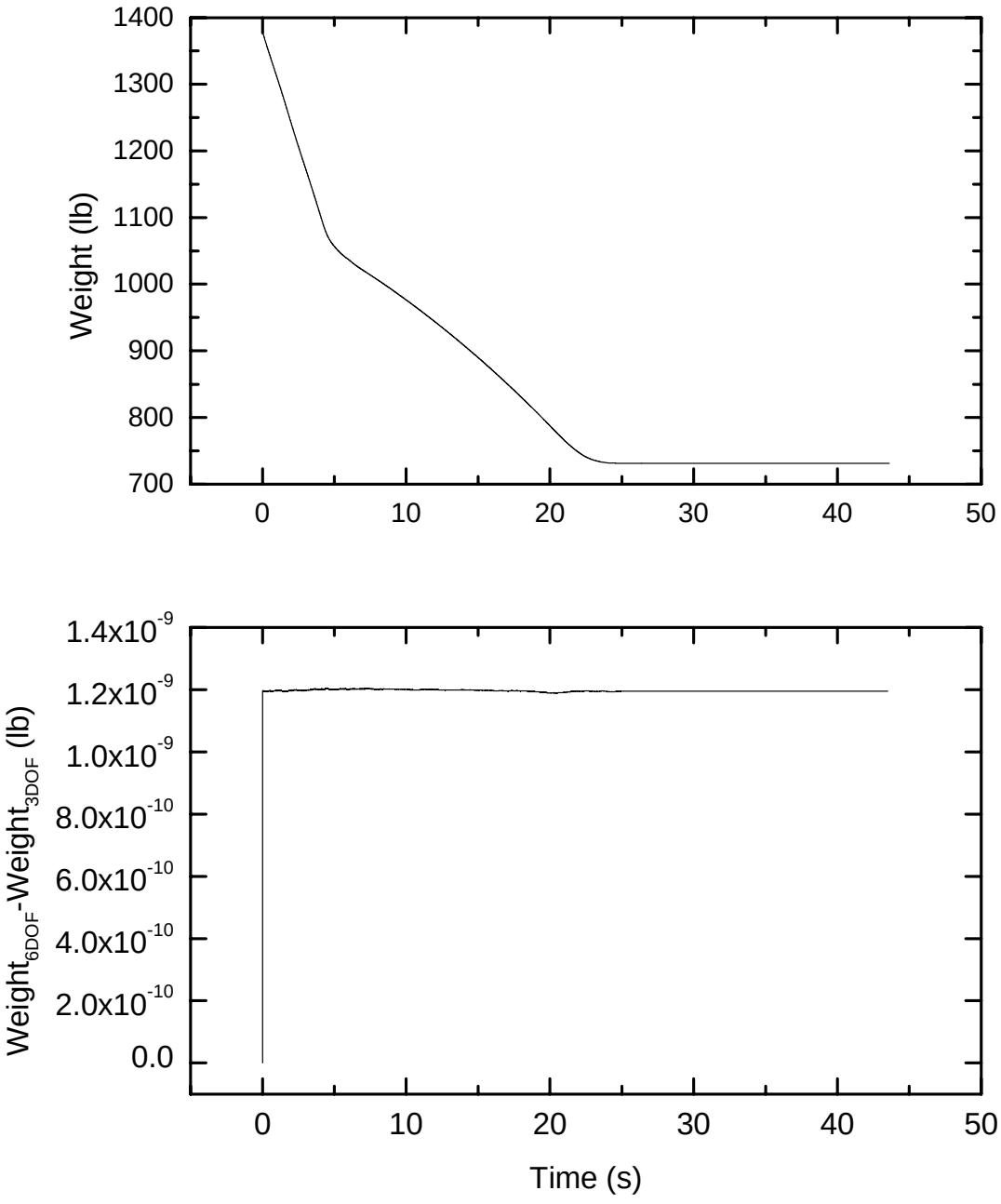

Figure D-9
Missile Weight as a Function of Time

D.6 Axial Force

The axial component of the aerodynamic force -- that is the component of the aerodynamic force along the x-axis of the missile -- is negative and is related to the aerodynamic drag on the missile. However, they are not the same. The drag is the aerodynamic force acting in the opposite direction of the total velocity of the missile. Only at zero angle of attack and zero side slip is the axial force equal to the drag force.

Figure D-10 shows the graph of axial force as a function of time and the difference plot for the two models. The key features of the axial force coefficient are a dip at around 1 second, a sudden drop at 24.9 seconds, and the apparent discontinuities in the derivative of C_x at about 31 seconds, 35 seconds, and 40 seconds. There are sharp changes in the difference plot that correspond to each of the features in the C_x curve. In order to better understand these features, let us consider the axial force coefficient as a function of Mach number as shown in Figure D-11 and the Mach number as a function of time as shown in Figure D-6. The graphs in Figure D-11 represent the input model for the axial force coefficient. As discussed in Appendix C, the axial force is larger in magnitude when the thrust is off than when the thrust is on.

Initially, the thrust is on and the upper of the two C_x curves in Figure D-11 is used to compute the trajectory for both the 6DOF and 3DOF model. As the Mach number climbs rapidly in the early part of the trajectory, we see that there should be a dip in C_x as the missile crosses Mach 1. This occurs around 1 second for each trajectory, as seen in Figure D-10.

At 24.9 seconds, C_x increases sharply in magnitude from approximately -0.30 to -0.42. This sharp increase is a result of switching from using the upper C_x curve, when the thrust is on, to using the lower C_x curve, when the thrust is off. It is well known that motor thrust alters the drag on a body in flight because of a change in the base pressure drag. Therefore, this discontinuity in C_x models a real discontinuity in the physics.

The discontinuities in the derivative of C_x for times greater than 24.9 seconds (Figure D-10) are a reflection of the piecewise linear representation of C_x as a function of Mach number used as an input to the model (Figure D-11). The Mach number is decreasing smoothly in this region (Figure D-6) and so any discontinuities in the derivative of C_x as a function of Mach number are also apparent in a plot of C_x as a function of time. We stress that these discontinuities in the derivative of C_x are a direct result of the input model for C_x and, unlike the discontinuity at 24.9 seconds, are not the result of any real physics.

The difference plot for C_x shows that the difference between the 6DOF and 3DOF models grows relatively smoothly between 0 and 24.9 seconds. This difference arises because the Mach number differs as a function of time in the two models. The difference in Mach number results in a difference in the lookup value of C_x at any given time. The difference in C_x grows as the difference in Mach number grows.

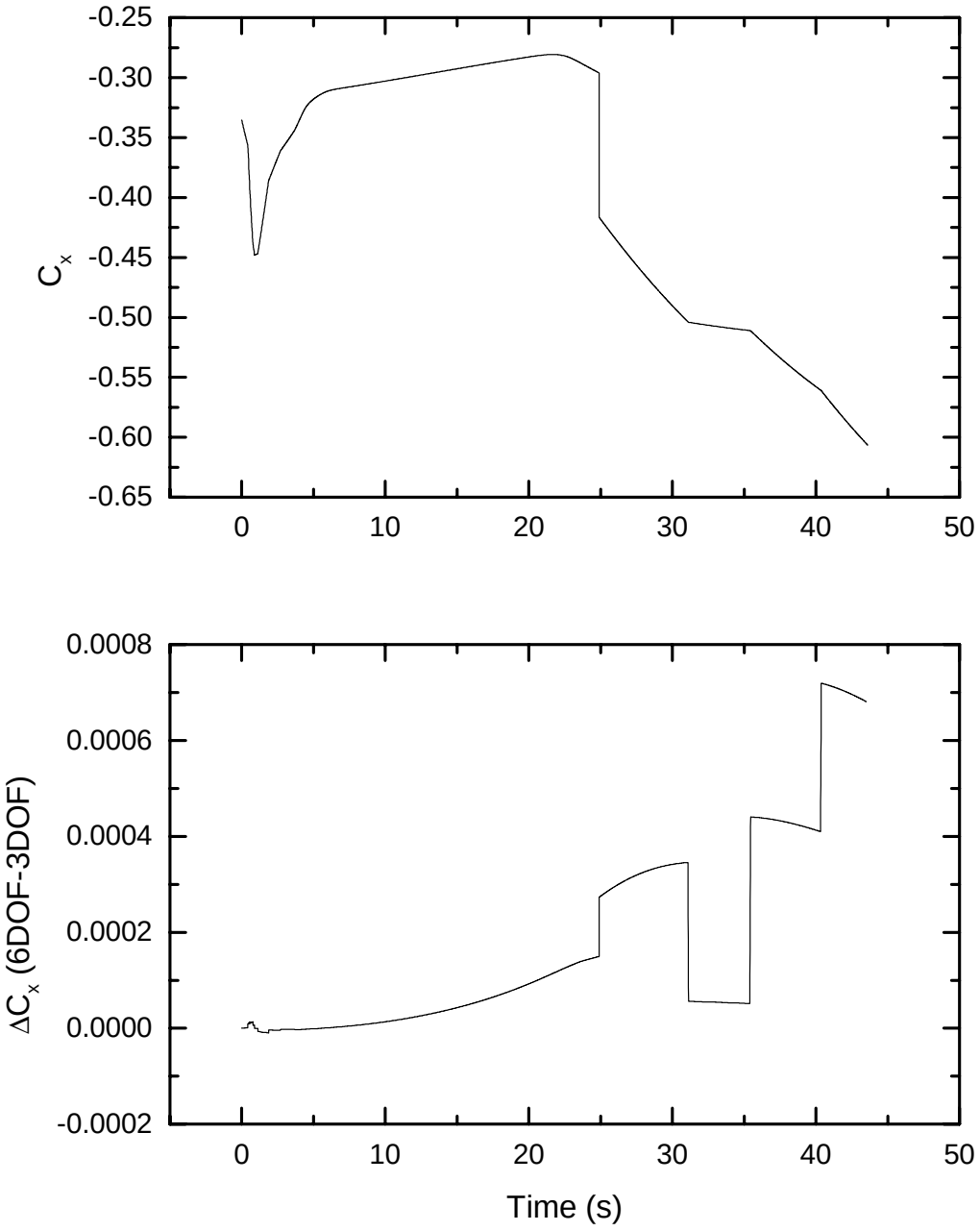

Figure D-10
Axial Force Coefficient as a Function of Time

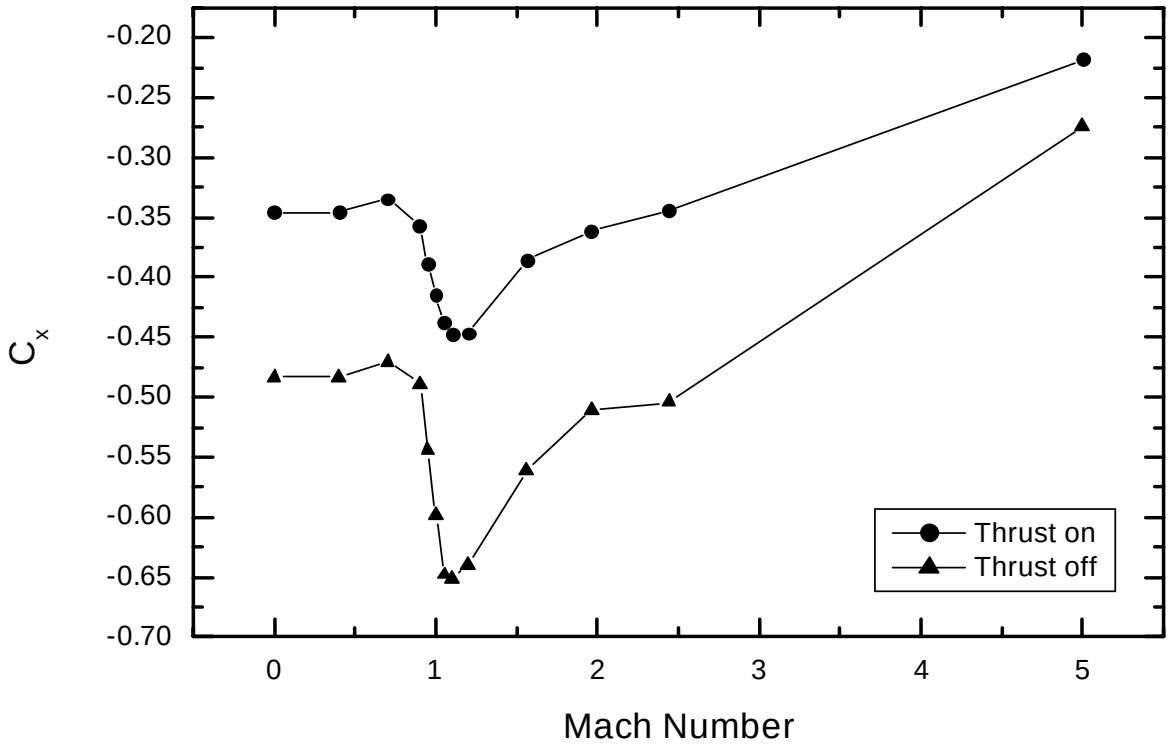

Figure D-11
Axial Force Coefficient as a Function of Mach Number

At 24.9 seconds, there is a step discontinuity of about 1×10^{-4} in the difference plot for C_x. The difference in the models to the left of the discontinuity is a result of using the upper curve (thrust on) in Figure D-11. The difference to the right of the discontinuity is a result of using the lower curve (thrust off). Note that in Figure D-11, the slope of the lower C_x curve is noticeably steeper than the upper curve when the Mach number is near 3.4, the approximate value of the Mach number when the thrust is switched off. A given difference in Mach number will therefore result in a larger difference in lookup values for C_x when determined using the lower curve than that computed using the upper curve. The magnitude of the discontinuity is thus due to the difference in slope between the two C_x curves combined with the slight difference in Mach number between the two models.

Similarly, the sharp changes in ΔC_x at approximately 31 seconds, 35 seconds, and 40 seconds shown in Figure D-10 are due to a combination of the piecewise-linear model of C_x as a function of Mach number and the difference in Mach number as a function of time for the 6DOF and 3DOF models. As the Mach number decreases for both models in this time period (Figure D-6), different segments of the piecewise-linear model are used for looking up values of C_x. Looking at the curve of C_x vs. time, discontinuities in the slope can be seen at 31, 35, and 40 seconds.

These slope discontinuities correspond to Mach numbers of 2.44, 1.96, and 1.56, respectively, in the C_x vs. Mach number table (Table C-5 in Appendix C). The near discontinuities in ΔC_x at 31, 35, and 40 seconds are due to the differences in slope on each side of these table points, combined with the difference in time when the 6DOF and 3DOF models cross these table points.

We are now in a position to understand a certain feature observed earlier in the plots of velocity and Mach number (Figure D-5 and Figure D-6). C_x is related to the axial force and from Newton's laws, we know that the axial acceleration of the body is related to the axial force. Since the velocity is the first integral of the acceleration, we might expect to see some effects in the velocity from the features we have observed in C_x. Specifically, a step discontinuity in C_x will appear as a discontinuity in the derivative of the velocity (the relative magnitude of the discontinuity will depend on the size of the step discontinuity in relation to the other forces acting in the same direction). Thus, the step discontinuity observed in C_x at 24.9 seconds results in the discontinuity in derivative in the velocity at 24.9 seconds (Figure D-5). Discontinuities in the derivative of C_x will not be apparent in the velocity since the integral of a piecewise linear function is smooth. The step discontinuities in the difference plot of C_x appear as discontinuities in the derivative in the difference plot of the velocity for each model. This explanation of features applies to the Mach number curve as well since the Mach number is simply a scaled velocity (Figure D-6).

D.7 Total Angle of Attack

In the 6DOF model, the angle between the x-axis of the missile, $\hat{x}_b$, and the wind-corrected velocity vector is defined to be the total angle of attack of the missile (Figure D-12). Note that for our calculations, the winds were assumed to be zero so the wind-corrected velocity vector is simply the velocity vector of the missile. The total angle of attack is always greater than or equal to zero. The total angle of attack is defined to be zero in the 3DOF model.

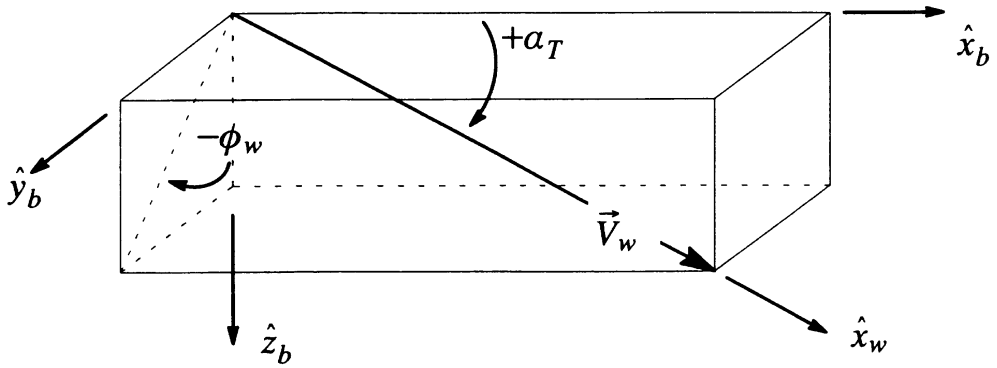

Figure D-12
Total Angle of Attack[91]

The total angle of attack as a function of time is shown in Figure D-13. It shows small (< 0.4 deg.), relatively high-frequency launch oscillations that damp down in the first several sec-

onds. These launch oscillations arise as the missile begins to pitch over after leaving the aircraft. There is a small mean total angle of attack that persists throughout the trajectory. The small mean total angle of attack results in a small lift that causes the missile to fly longer in the 6DOF simulation than the 3DOF simulation, as discussed earlier.

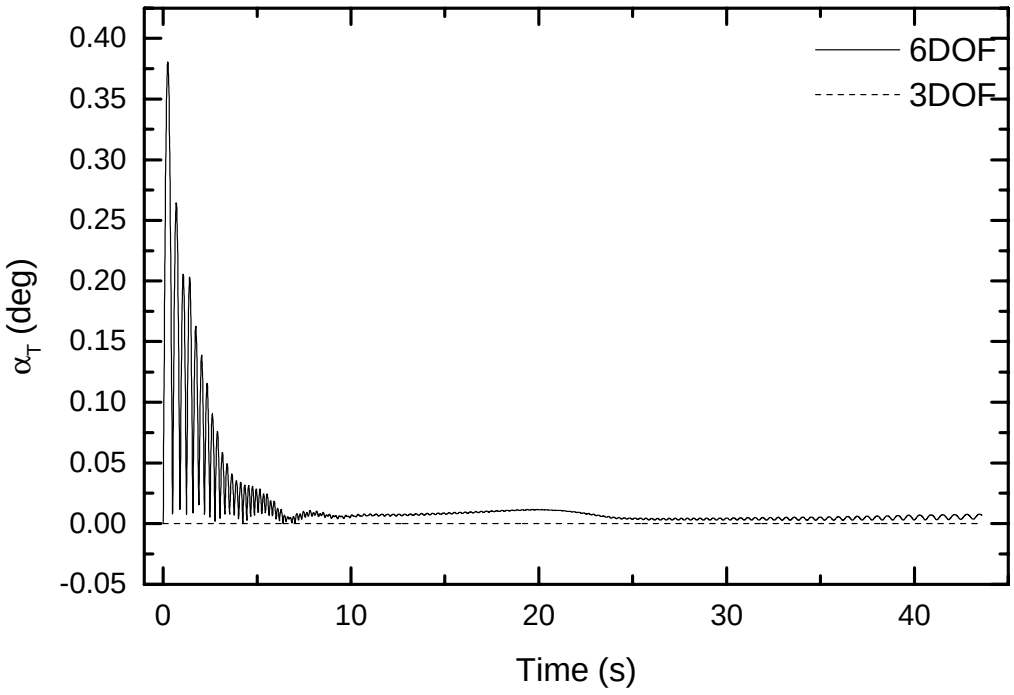

Figure D-13
Total Angle of Attack as a Function of Mach Number

D.8 Yaw, Pitch, and Roll Angles

The orientation of the missile is described by three Euler angles: yaw, pitch, and roll. A consistently defined set of yaw, pitch, and roll angles can be used to relate the orientation of the body axes to any of several coordinate systems. We chose to use the most common set, those relating the body orientation to the local geodetic horizon coordinate system with a yaw-pitch-roll sequence. Since we assumed a spherical Earth, the local geodetic horizon coordinate system (based on ellipsoidal geometry) is the same as the local geocentric horizon coordinate system (based on spherical geometry). As shown in Figure D-14, the origin of the local geocentric horizon coordinate system is located at the vehicle's center of mass. The position vector, indicated by $\vec{r}$, is the vector from the center of the Earth to the vehicle's center of mass. The z-axis is aligned with $\vec{r}$ but points in the opposite direction, i.e., toward the center of the Earth. The x and y axes are in a plane perpendicular to $\vec{r}$ and tangent to the Earth's surface at the point that $\vec{r}$ intersects the Earth's surface, *S*. The x-axis points north and the y-axis points east.

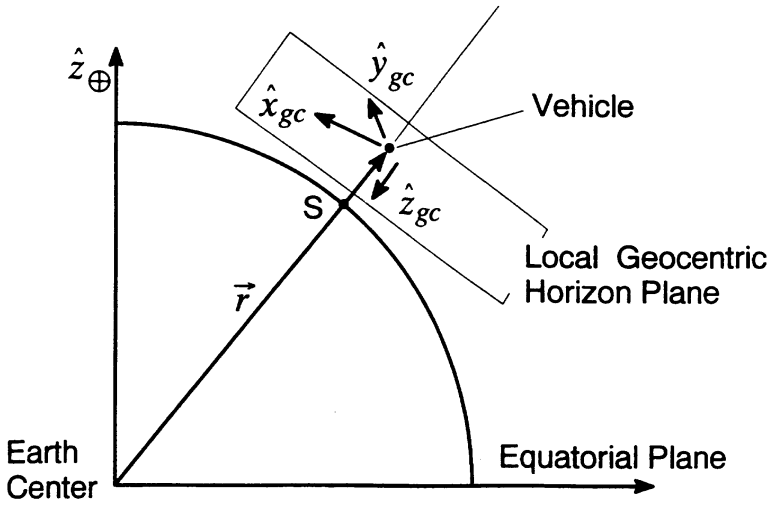

Figure D-14
Local Geocentric Horizon Coordinate System[91]

The yaw, pitch, and roll are the set of Euler angles defining the transformation from the local geodetic horizon coordinate system to the body coordinate system. The set of transformations is described in detail in Ref. 91. Figure D-15 shows the yaw and pitch angles. The yaw angle, Ψ_{gd}, is the angle between the geodetic x-axis and the projection of the body x-axis onto the x-y plane of the local geodetic horizon coordinate system. The pitch angle, Θ_{gd}, is the angle between the body x-axis and the x-y plane of the local geodetic horizon coordinate system with positive pitch measured from the x-y plane toward the negative z-axis of the geodetic system. The roll angle (not shown) is simply the rotation about the body x-axis required to align the y- and z-axes of the body coordinate system with those of the geodetic system once the yaw and pitch rotations are complete.

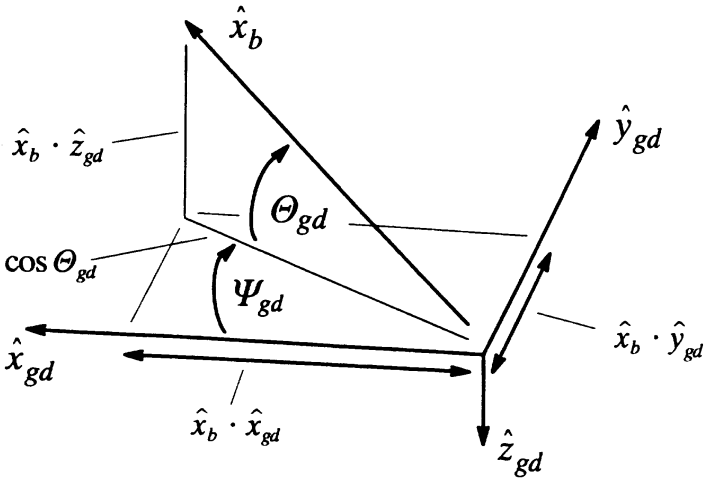

Figure D-15
Geodetic Yaw and Pitch Angles[91]

The yaw angle for the 6DOF model is shown in Figure D-16. The figure shows that the x-axis of the missile oscillates to the left and right of this plane by up to 0.03 deg. It is primarily to the right of this plane, which is consistent with the missile flying eastward a few feet (Figure D-4). The yaw angle in the 6DOF model shows oscillatory behavior with a time-varying envelope reflecting the complex, coupled angular motion of the missile. The yaw is defined to be zero in the 3DOF model.

The pitch angle for both the 6DOF and 3DOF models is shown in Figure D-17. In the 3DOF model, the body x-axis is assumed to be aligned with the velocity vector and therefore the pitch angle can be computed using the velocity vector. In the 6DOF model, the body axes can be misaligned with the velocity vector and the pitch reflects the angle of the body axes. For both models, the pitch angle shows a smooth decrease as a function of time, corresponding to the missile pitching over from horizontal as it falls to Earth. In the 6DOF model, there is a a small oscillation on top of this pitching motion. This can be seen more clearly in the difference graph between the 6DOF and 3DOF models. Since the 3DOF pitch angle must be aligned with the velocity vector, the difference graph shows the angle above and below the velocity vector in the vertical north-south plane. The mean angle of the pitch oscillation is positive. This is the angle that directly contributes to the lift on the missile, resulting in a somewhat longer range for the 6DOF model.

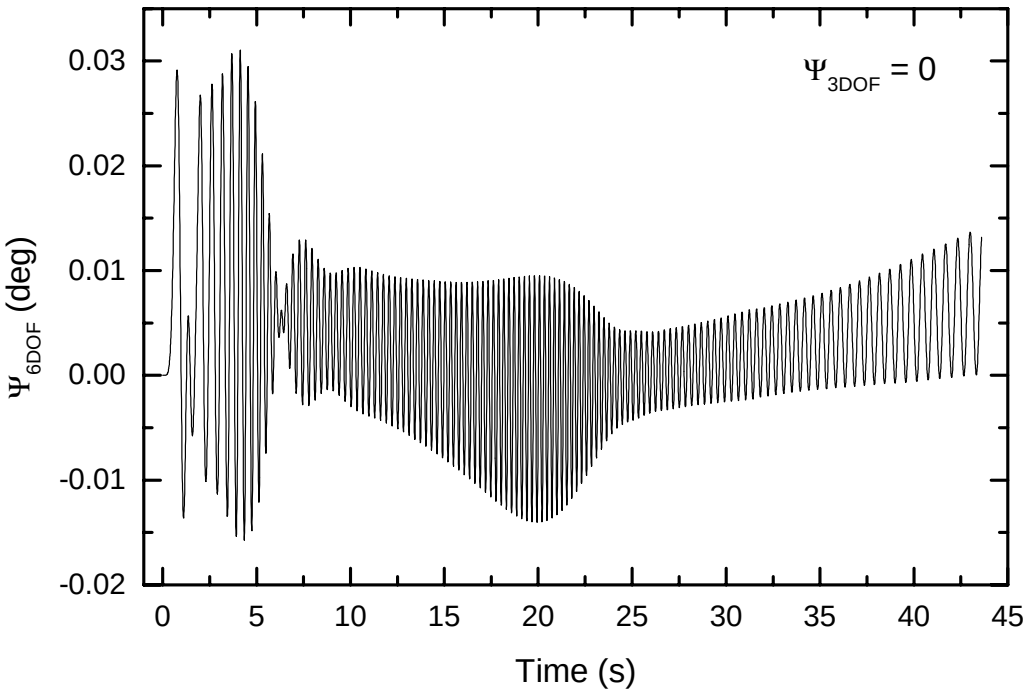

Figure D-16
Yaw Angle as a Function of Time

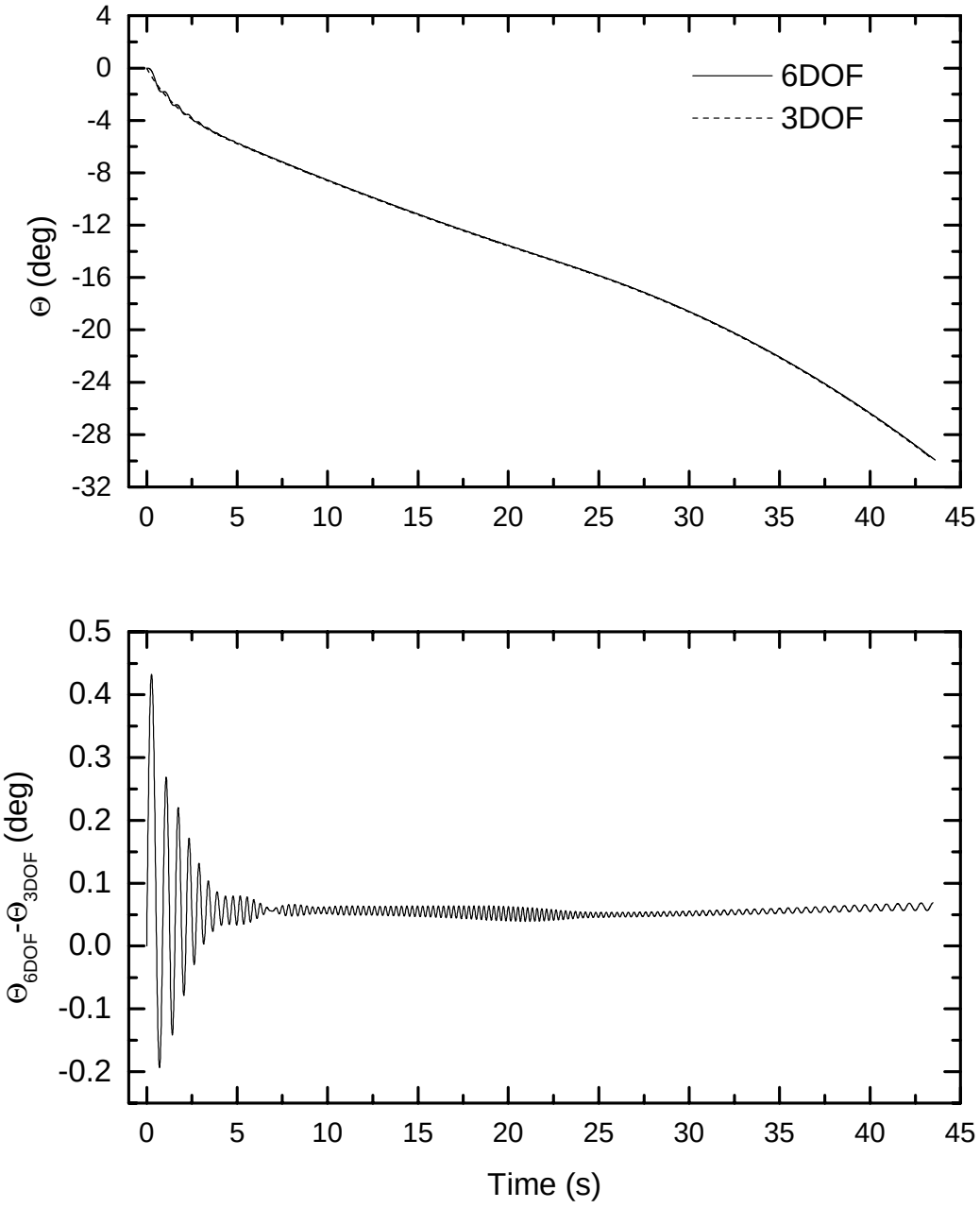

Figure D-17
Pitch Angle as a Function of Time

The roll angle for the 6DOF model is shown in Figure D-18. Although the roll angle appears to oscillate, the missile actually rolls continuously in a clockwise direction (looking in the direction of flight) throughout the entire trajectory. The calculation of the roll angle requires it to be between -180 and 180 degrees and thus there are repeated step discontinuities between 180 and -180 degrees in the plot. Note that the roll angle is undefined in the 3DOF model.

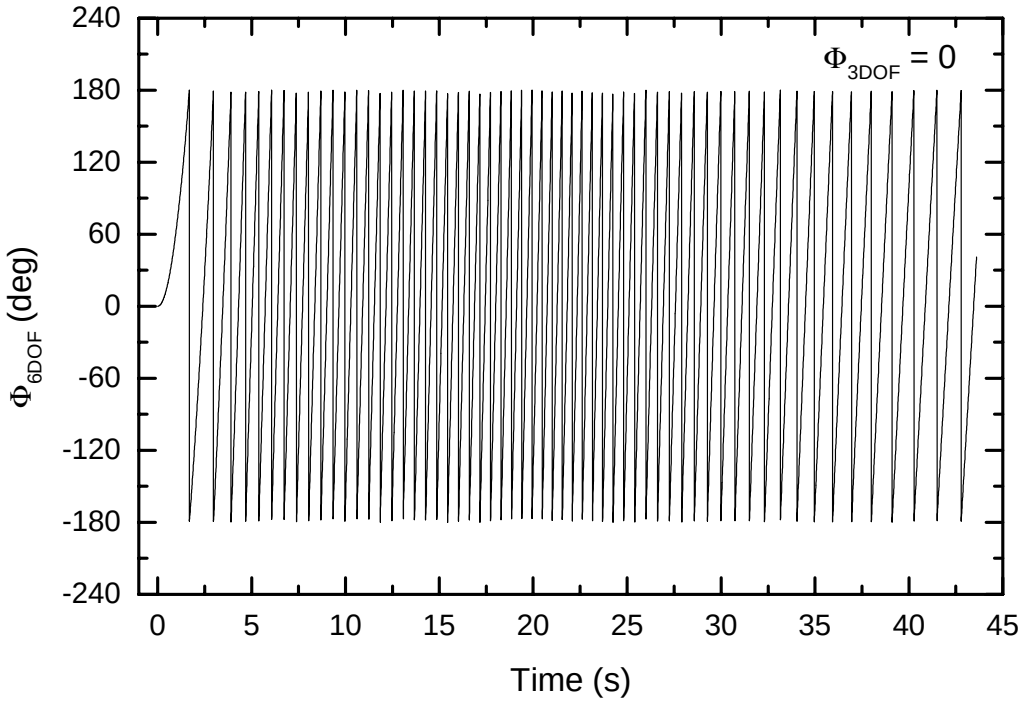

Figure D-18
Roll Angle as a Function of Time

D.9 Roll Rate

The angular rate around the x-axis in the body coordinate system, or roll rate, is shown in Figure D-19. The roll rate is undefined in the 3DOF model. For the 6DOF model, the plot shows the roll rate of the missile grows rapidly from 0 at launch, peaking at about 680 deg/sec, and decreasing to about 270 deg/sec at the end of trajectory. The roll rate is generated by the fixed deflection of each of the four fins. The deflection generates a roll torque that is proportional to the square of the missile velocity. As a result, it can be seen that Figure D-19 is very similar to the velocity as a function of time (Figure D-5).

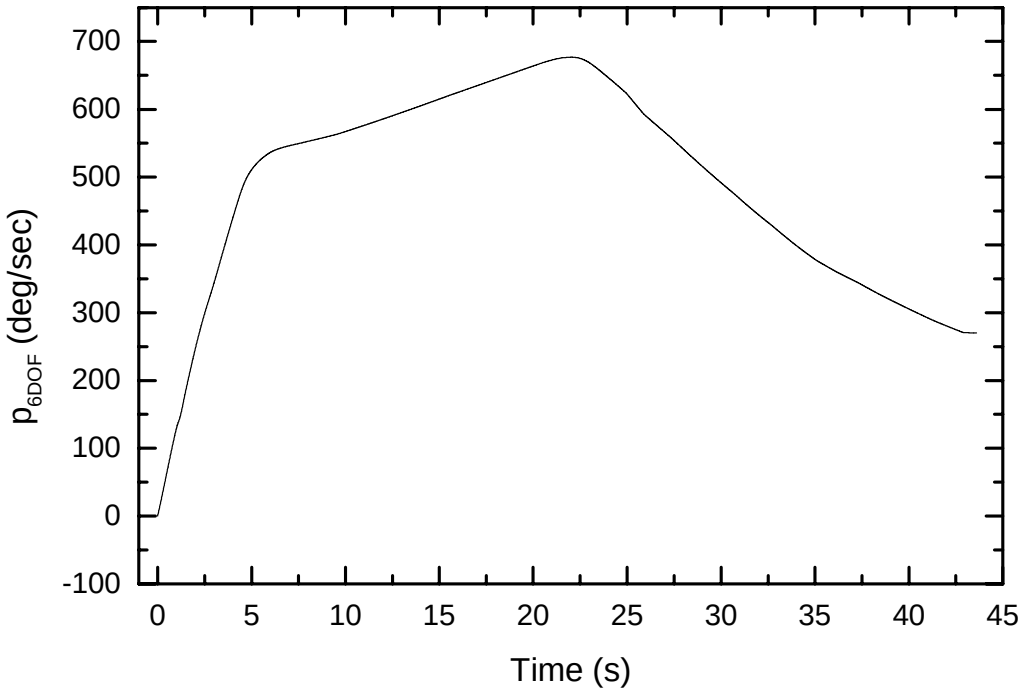

Figure D-19
Angular Rate About the Body X-Axis

D.10 Numerical Integration Time Steps

To provide an improved understanding of the numerical integration and its associated error, this section discusses the minimum and maximum time steps used during each print interval in the trajectory calculations. Figure D-20 shows plots of the largest time step, Δt_{max}, taken in each print interval as a function of time. Figure D-21 shows plots of the smallest time step, Δt_{min}, taken in each print interval as a function of time. In both figures, the top graph is a linear scale and the bottom graph is logarithmic in the dependent variable.

The two sets of figures share several interesting features. First, it is clear that the time step used in the 3DOF calculation was very often equal to the print interval, 0.005 s. As explained in Appendix B, this means that the time step needed to satisfy the per-step truncation error criterion is larger than the print interval. The 3DOF calculation therefore has a higher, but unknown, accuracy than that specified by the error criterion.

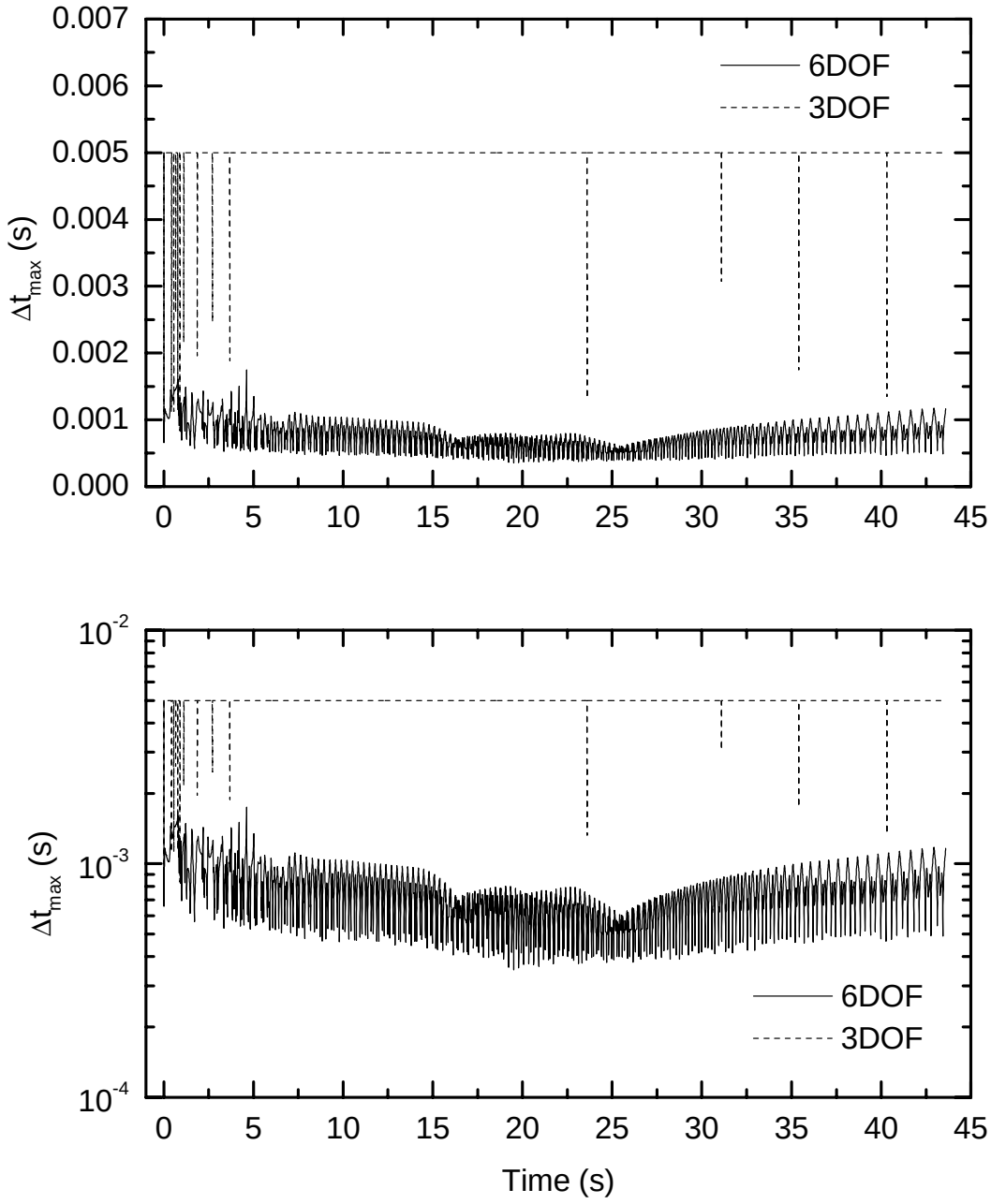

Figure D-20
Maximum Time Step as a Function of Time

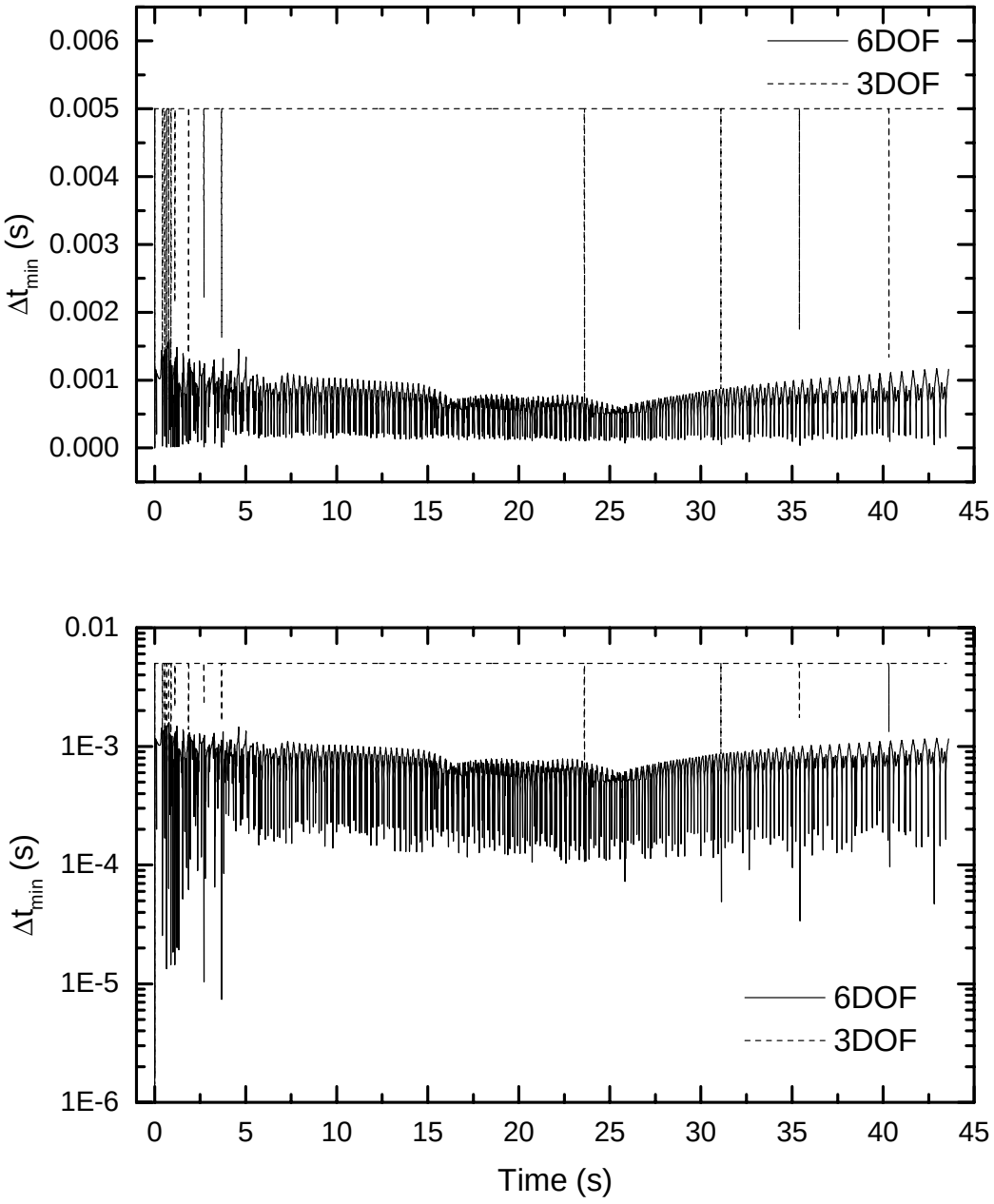

Figure D-21
Minimum Time Step as a Function of Time

Second, the time step used by the 3DOF calculation dropped by a factor of 5-10 at a small number of points. These points generally correspond to a time at which the Mach number is equal to one of the values occurring in the input tables for the aerodynamic force coefficients. As discussed earlier, the aerodynamic force coefficients are modeled as piecewise linear functions of Mach number. When the Mach number crosses one of the values used in the tables, at least one of the aerodynamic forces has a discontinuity in slope. The discontinuity in slope forces the integrator to use a smaller time step in order to satisfy the error criterion.

Third, the time step used by the 6DOF model is generally more than a factor of 5 smaller than that used by the 3DOF model. The difference would be much larger if the time step used in the 3DOF calculation had not been constrained by the print interval. The reason for the difference is that many of the state variables in the 6DOF model exhibit periodic motion, while the state variables in the 3DOF model vary quite smoothly and do not exhibit periodic motion. The integrator is a fifth-order accurate method and can use relatively large time steps in the 3DOF calculation; the time step must be much smaller in the 6DOF model to accurately capture the periodic behavior of all of the state variables.

The variables that exhibit periodic motion in the 6DOF model are those relating to the angular degrees of freedom. The state variables with the highest frequencies are the pitch rate and yaw rate. Both have frequencies of 1-2 Hz. As discussed in Appendix B, state variables that show oscillatory behavior may force the integrator to use shorter time steps in order to satisfy the per-step truncation error criterion. The time steps used in the 6DOF calculation, typically near 0.001s, are much shorter than the period of the motion, 0.5s, because of the requirement to satisfy a very stringent error criterion, $E_t = 1\times10^{-12}$.

As a final point, note that the Δt_{min} is always greater than the minimum time step input parameter of 1×10^{-8} s.

Appendix E : Sample TAOS Files

TAOS takes two types of input file: problem files containing trajectory definition parameters and table files containing tabular data such as thrust, mass flow, and aerodynamic coefficients. Sample input files representative of those used to perform the calculations in this report are given below.

Sample Problem File

```
#
(determine_error_settings)  # Problem to be solved

*title  Air-Launch Single-Stage Improved Hawk

# Uncomment one of the two lines below to switch between 6DOF and 3DOF
# Applies to all of the trajectories and segments below

*sixdof
#*ptmass

*method rk45-var relerr = 1.0e-12 abserr = 1.0e-7  # Variable step RK-45
                                                   # integrator.  Step size
                                                   # determined by relerror

*atmos standard           # 1976 US Standard Atmosphere, U.S. 75 North, January
*earth spherical omega=0  # Spherical Earth, no rotation, gravitational const.
                          # and lbm to slug conversion factor default to
                          # WGS-84 values

# No wind defined

# The missile trajectory
*trajectory  1  Improved-Hawk start on 1

  *initial      time=0.0      mass=42.86

     geodetic   alt=30000     long=0.0           lat=0.0
                vel=700       gama=0.0           psi=0.0
                          # gamma = flight path angle (deg)
                  # psi = heading angle (deg)
                          # psi = 0 ==> heading due north initially
          # gama = 0 ==> initial velocity vector is horizontal

     geodetic   yaw=0.0       pitch=0.0          roll=0.0 # Not used in 3DOF
```

```
    bodyi      p=0.0         q=0.0              r=0.0    # initial rates about
                                                         # the x, y, z axes.
                                                         # Not used in 3DOF

*segment  1  Boost

  *integ  dtprnt=.0050   dtmin=1.e-8       # print interval and minimum dt
                                           # for integration

  *aero  sref=1.0690   lrefx=1.16667   lrefy=1.16667  # ref area and lengths
                                                     # ref lengths for 6DOF
                                                     # only
         xref=0.0      yref=0.0        zref=0.0      # ref point is the nose
                                                     # 6DOF only

         #Linearized Aero Coefficients -- cyr and czq are 6 DOF only
         cx=(hawk_cx_on)      fcxon=1.0       fcxoff=1.0
         cybeta=(hawk_cyb)    cyr=(hawk_cyr)  fcyb=1.0  fcyr=1.0
         czalpha=(hawk_cza)   czq=(hawk_cza)  fcza=1.0  fczq=1.0

         #Linearized Aero Moment Coefficients -- 6DOF only
         cl=(hawk_cl)         clp=(hawk_clp)  fcl=1.0   fclp=1.0
         cmalpha=(hawk_cma)   cmq=(hawk_cmq)  fcma=1.0  fcmq=1.0
         cnbeta=(hawk_cnb)    cnr=(hawk_cnr)  fcnb=1.0  fcnr=1.0

  # No *fly block indicates alpha, betae, geocentric bank angle,
  # and power setting are all set to zero in 3DOF mode.

  # No *limits block indicates no limits have been placed on body
  # attitude angles in 3DOF


  *prop  x=-14.46       y=0.0       z=0.0        # Location used in 6DOF only
         tvc1=0.0       tvc2=0.0    fthr=1.0     fmdt=1.0
                                             # tvc1 and tvc2 used in 6DOF only
         thrust=(hawk_thr)  mdot=(hawk_mdt)

  # Mass properties -- used in 6DOF only since aero coefficients are not
  # a function of xcg.
  *mass  xcg=(hawk_xcg)   ycg=0.0          zcg=0.0
         fxcg=1.0         fixx=1.0         fiyy=1.0        fizz=1.0
         ixx=(hawk_ixx)   iyy=(hawk_iyy)   izz=(hawk_izz)
```

```
           ixy=0.0          iyz=0.0          ixz=0.0

    *when  time>24.5  goto 2

  *segment  2  Coast

    *integ  dtprnt=.0050   dtmin=1.e-8    # Want dtprint to be an even divisor
                                          # of 24.5 -- time at which thrust
                                          # switches from on to off

    *aero  sref=1.0690          lrefx=1.16667   lrefy=1.16667
           xref=0.0             yref=0.0        zref=0.0
           cx=(hawk_cx_off)     fcxon=1.0       fcxoff=1.0
           cybeta=(hawk_cyb)    cyr=(hawk_cyr)  fcyb=1.0  fcyr=1.0
           czalpha=(hawk_cza)   czq=(hawk_cza)  fcza=1.0  fczq=1.0
           cl=(hawk_cl)         clp=(hawk_clp)  fcl=1.0   fclp=1.0
           cmalpha=(hawk_cma)   cmq=(hawk_cmq)  fcma=1.0  fcmq=1.0
           cnbeta=(hawk_cnb)    cnr=(hawk_cnr)  fcnb=1.0  fcnr=1.0

    *mass  xcg=(hawk_xcg)   ycg=0.0          zcg=0.0
           fxcg=1.0         fixx=1.0         fiyy=1.0        fizz=1.0
           ixx=(hawk_ixx)   iyy=(hawk_iyy)   izz=(hawk_izz)
           ixy=0.0          iyz=0.0          ixz=0.0

    *when  alt<0  stop

*units/fmt time    e22.15
   alt     e22.15                      # Set precision of output
          east   e22.15# to show 15 digits
   north   e22.15
   long    e22.15
           latgd   e22.15
   xecfc   e22.15
   yecfc   e22.15
           zecfc   e22.15

   vel     e22.15
   mach    e22.15
           xecfcdt e22.15
           yecfcdt e22.15
           zecfcdt e22.15
   gamgd   e22.15
```

```
           psigd   e22.15

   p   e22.15
     q    e22.15
   r    e22.15
   yawgd   e22.15
   pitchgd e22.15
           rollgd  e22.15

           wt      e22.15
   mass    e22.15
           mdt     e22.15
   thrst   e22.15

   alphat  e22.15
           alpha   e22.15
           beta    e22.15
           cx      e22.15
           cy      e22.15
           cz      e22.15
   dynprs  e22.15

   plength e22.15
   range   e22.15
   dtmin   e22.15
   dtmax   e22.15


# output desired data to a file.  These variables are common between 3DOF
# and 6DOF
*file hawk6_9.dat
   time alt east north long latgd xecfc yecfc zecfc
vel mach xecfcdt yecfcdt zecfcdt gamgd psigd
p q r yawgd pitchgd rollgd
wt mass mdt thrst
alphat alpha beta cx cy cz dynprs
    plength
range
time dtmin dtmax

*end
```

Sample Table File

```
#
#   Aerodynamics => Improved Hawk (virtual rail configuration)
#
#   Ref:  L. Rollstin
#
#   Sref = 1.0690 ft**2
#   Lrefx = 1.16667 ft
#   Lrefy = 1.16667 ft
#
#   Xref = 0.0 ft (from nose)
#   Yref = 0.0 ft
#   Zref = 0.0 ft


#
#  Axial Force Coefficient - Coast
# --------------------------------

(hawk_cx_off)
table   aero_force

   start

      add   cx(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                     1.00,     1.05,     1.10,     1.20,     1.56,
                     1.96,     2.44,     5.00

         cx     =   0.483,    0.483,    0.470,    0.489,    0.544,
                    0.598,    0.648,    0.651,    0.640,    0.561,
                    0.511,    0.504,    0.274,
      neg
      mult  fcxoff
   end


#
#  Axial Force (cx) Power On
```

Sample Table File

```
# -------------------------

(hawk_cx_on)
table   aero_force

   start

      add   cx(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                      1.00,     1.05,     1.10,     1.20,     1.56,
                      1.96,     2.44,     5.00

         cx     =   0.346,    0.346,    0.335,    0.357,    0.389,
                    0.415,    0.438,    0.448,    0.447,    0.386,
                    0.361,    0.344,    0.218,
      neg
      mult fcxon
   end


#
#  Normal Force Coefficient Slope
# ------------------------------

(hawk_cza)
table   aero_force

   start

      add   cza(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                      1.00,     1.05,     1.10,     1.20,     1.56,
                      1.96,     2.44,     5.00

         cza    =  -0.411,   -0.411,   -0.436,   -0.456,   -0.464,
                   -0.482,   -0.475,   -0.480,   -0.492,   -0.422,
                   -0.352,   -0.311,   -0.178
      mult fcza
   end
```

```
#
#  Normal Force Damping
# ---------------------

(hawk_czq)
table   aero_force

   start

      add   czq(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                     1.00,     1.05,     1.10,     1.20,     1.56,
                     1.96,     2.44,     5.00

         czq    =   255.4,    255.4,    271.1,    283.1,    301.6,
                    313.4,    309.1,    312.4,    319.7,    295.5,
                    246.6,    217.3,    125.2

      div  57.2958   # 180 degrees / pi radians
      mult fczq
   end


#
#  Side Force Coefficient Slope
# -----------------------------

(hawk_cyb)
table   aero_force

   start

      add   cyb(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                     1.00,     1.05,     1.10,     1.20,     1.56,
                     1.96,     2.44,     5.00

         cyb    =  -0.411,   -0.411,   -0.436,   -0.456,   -0.464,
                   -0.482,   -0.475,   -0.480,   -0.492,   -0.422,
```

```
                    -0.352,    -0.311,    -0.178
      mult fcyb
   end


#
#  Side Force Damping
# -------------------

(hawk_cyr)
table   aero_force

   start

      add   cyr(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                     1.00,     1.05,     1.10,     1.20,     1.56,
                     1.96,     2.44,     5.00

         cyr    =   255.4,    255.4,    271.1,    283.1,    301.6,
                    313.4,    309.1,    312.4,    319.7,    295.5,
                    246.6,    217.3,    125.2

      div  57.2958
      mult fcyr
   end


#
#  Pitching Moment Coefficient Slope
# ----------------------------------

(hawk_cma)
table   aero_moment

   start

      add   cma(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                     1.00,     1.05,     1.10,     1.20,     1.56,
```

```
                    1.96,     2.44,     5.00

         cma    =  -4.953,   -4.953,   -5.281,   -5.542,   -5.657,
                   -5.882,   -5.847,   -5.928,   -5.680,   -4.817,
                   -3.967,   -3.432,   -1.889
      mult fcma
   end


#
#  Pitching Moment Damping
# ------------------------

(hawk_cmq)
table   aero_moment

   start

      add   cmq(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                     1.00,     1.05,     1.10,     1.20,     1.56,
                     1.96,     2.44,     5.00

         cmq    = -1385.3,  -1385.3,  -1470.2,  -1535.6,  -1711.6,
                   -1778.8,  -1754.4,  -1772.9,  -1814.2,  -1804.3,
                   -1505.3,  -1326.8,   -767.6

      div  57.2958
      mult fcmq
   end


#
#  Yawing moment coefficient slope
# --------------------------------

(hawk_cnb)
table   aero_moment

   start
```

```
      add   cnb(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                      1.00,     1.05,     1.10,     1.20,     1.56,
                      1.96,     2.44,     5.00

         cnb    =   4.953,    4.953,    5.281,    5.542,    5.657,
                    5.882,    5.847,    5.928,    5.680,    4.817,
                    3.967,    3.432,    1.889
      mult fcnb
   end


#
#  Yawing moment damping
# ----------------------

(hawk_cnr)
table   aero_moment

   start

      add   cnr(mach)

         mach   =    0.00,     0.40,     0.70,     0.90,     0.95,
                      1.00,     1.05,     1.10,     1.20,     1.56,
                      1.96,     2.44,     5.00

         cnr    = -1385.3,  -1385.3,  -1470.2,  -1535.6,  -1711.6,
                  -1778.8,  -1754.4,  -1772.9,  -1814.2,  -1804.3,
                  -1505.3,  -1326.8,   -767.6

      div  57.2958
      mult fcnr
   end


#
#  Rolling moment coefficient
# ---------------------------

(hawk_cl)
```

Sample Table File

```
table   aero_moment

   start

      add   cldel(mach)      #  cl-delta-fin-deflection

         mach   =    0.00,    0.50,    0.75,    0.95,    1.00,
                      1.05,    1.10,    1.15,    1.20,    1.25,
                      1.30,    1.40,    1.50,    1.60,    1.70,
                      1.80,    1.90,    2.00,    2.25,    2.50,
                      2.75,    3.00,    3.25,    3.50,    3.75,
                      4.00,    4.50,    5.00,    5.50

         cldel  =  0.3244,  0.3244,  0.3369,  0.3493,  0.3529,
                   0.3547,  0.3529,  0.3386,  0.2709,  0.2852,
                   0.3351,  0.3957,  0.4367,  0.4563,  0.4670,
                   0.4705,  0.4705,  0.4670,  0.4563,  0.4367,
                   0.4135,  0.3903,  0.3672,  0.3493,  0.3351,
                   0.3208,  0.2923,  0.2691,  0.2495

      mult  0.25  #  Fin deflection (deg)
      mult  fcl
   end


#
#  Rolling moment damping
# -----------------------

(hawk_clp)
table   aero_moment

   start

      add   clp(mach)

         mach   =     0.00,     0.50,     0.75,     0.95,     1.00,
                      1.05,     1.10,     1.15,     1.20,     1.25,
                      1.30,     1.40,     1.50,     1.60,     1.70,
                      1.80,     1.90,     2.00,     2.25,     2.50,
                      2.75,     3.00,     3.25,     3.50,     3.75,
                      4.00,     4.50,     5.00,     5.50
```

```
         clp    =   -45.69,   -45.69,   -47.53,   -49.00,   -49.74,
                    -50.11,   -49.74,   -49.37,   -44.21,   -38.69,
                    -44.21,   -56.00,   -61.90,   -64.48,   -65.95,
                    -66.32,   -66.69,   -66.32,   -63.74,   -60.79,
                    -57.85,   -54.90,   -52.32,   -49.37,   -47.16,
                    -44.95,   -41.27,   -37.95,   -36.48

      div  57.2958
      mult fclp
   end


#
#   Propulsion => Improved Hawk
#
#  Imp Hawk Thrust Curve from L. Rollstin, 1551
#  From Ameer file -- starts at t = 0 sec
#

(hawk_thr)
  table  thrust
  start
     sub  pres
     mult 0.5248
     add  tvac(time)

        time =   0.000,    0.001,    0.200,    0.400,    0.600,
                  0.800,    1.000,    1.200,    1.400,    1.600,
                  1.800,    2.000,    2.200,    2.400,    2.600,
                  2.800,    3.000,    3.200,    3.400,    3.600,
                  3.800,    4.000,    4.200,    4.400,    4.600,
                  4.800,    5.000,    5.200,    5.400,    5.600,
                  5.800,    5.850,    6.000,    6.200,    6.400,
                  6.600,    6.800,    7.000,    7.200,    7.400,
                  8.000,   10.000,   12.000,   14.000,   16.000,
                 18.000,   20.000,   20.200,   21.000,   22.000,
                 23.000,   24.000,   24.900,   99.000

        tvac  =  1110.6,  20525.9,  19696.5,  18867.2,  18379.6,
                18331.3,  18574.6,  18932.2,  19290.7,  19648.3,
                19867.9,  19671.9,  19258.2,  18843.5,  18599.3,
```

```
                18526.4,  18452.5,  18644.6,  18836.6,  19027.7,
                19225.7,  19428.6,  19306.5,  15515.2,  11724.0,
                 8697.1,   7965.2,   7233.3,   6596.0,   5957.8,
                 5320.5,   5160.9,   4480.3,   3766.1,   3533.7,
                 3304.2,   3180.1,   3159.4,   3138.7,   3159.4,
                 3221.4,   3402.7,   3581.9,   3763.2,   3946.4,
                 4131.6,   4347.3,   4369.9,   4150.3,   3369.2,
                 1020.0,    280.2,     17.2,     17.2
    mult fthr
    min  0.0
  end

(hawk_mdt)
  table  mdot  units=slugs/sec
  start

    add  mdt(time)

       time =   0.000,    0.001,    0.200,    0.400,    0.600,
                0.800,    1.000,    1.200,    1.400,    1.600,
                1.800,    2.000,    2.200,    2.400,    2.600,
                2.800,    3.000,    3.200,    3.400,    3.600,
                3.800,    4.000,    4.200,    4.400,    4.600,
                4.800,    5.000,    5.200,    5.400,    5.600,
                5.800,    5.850,    6.000,    6.200,    6.400,
                6.600,    6.800,    7.000,    7.200,    7.400,
                8.000,   10.000,   12.000,   14.000,   16.000,
               18.000,   20.000,   20.200,   21.000,   22.000,
               23.000,   24.000,   24.900,   99.000

       mdt   =   0.000,    2.306,    2.208,    2.109,    2.030,
                2.045,    2.074,    2.117,    2.159,    2.202,
                2.228,    2.205,    2.156,    2.106,    2.077,
                2.069,    2.060,    2.083,    2.105,    2.128,
                2.152,    2.176,    2.161,    1.711,    1.261,
                0.901,    0.814,    0.727,    0.652,    0.576,
                0.500,    0.481,    0.520,    0.567,    0.517,
                0.468,    0.442,    0.437,    0.433,    0.437,
                0.451,    0.489,    0.528,    0.566,    0.606,
                0.645,    0.691,    0.696,    0.649,    0.482,
                0.214,    0.056,    0.000,    0.000
     mult fmdt
```

```
   end



#
#   Mass Properties => Single-Stage Improved Hawk
#   (Winker Payload)

(hawk_xcg)
table   cg
   start
      add xcg(mass)
        mass    =  0.00,  22.75,  23.25,  32.36,  42.86,  99.99
        xcg     = -9.021, -9.021, -9.051, -9.552, -9.847, -9.847
      mult fxcg
   end

(hawk_ixx)
table   inertia
   start
      add ixx(mass)
        mass    =   0.00,  22.75,  23.25,  32.36,  42.86,  99.99
        ixx     =  7.097,  7.097,  7.267,  9.537, 10.667, 10.667
      mult fixx
   end

(hawk_iyy)
table   inertia
   start
      add iyy(mass)
        mass    =   0.00,  22.75,  23.25,  32.36,  42.86,  99.99
        iyy     =  884.3,  884.3,  884.9,  914.5,  934.3,  934.3
      mult fiyy
   end

(hawk_izz)
table   inertia
   start
      add izz(mass)
        mass    =   0.00,  22.75,  23.25,  32.36,  42.86,  99.99
        izz     =  884.3,  884.3,  884.9,  914.5,  934.3,  934.3
      mult fizz
   end
```

External Distribution

2 Tim Hasselman
ACTA
2790 Skypark Dr, Suite 310
Torrance, CA 90505-5345

1 David Belk
WL/MNAA
101 W. Eglin Blvd, Suite 219
Eglin AFB, FL 32542-6810

2 Michael Mendenhall
Nielsen Engr. and Research, Inc.
510 Clyde Ave.
Mountain View, CA 94043

1 Robert Nelson
Dept. of Aerospace and Mechanical Engr.
University of Notre Dame
Notre Dame, IN 46556

2 NASA/Langley Research Center
Attn: Michael Hemsch, MS 280
Jim Weilmuenster, MS 408A
Jim Luckring, MS 280
Hampton, VA 23681-0001

1 Kevin Greenaugh
DP 51
Dept. of Energy
1000 Independence Ave., SW
Washington, DC 20585

2 Paul Messina
DP 51
Dept. of Energy
1000 Independence Ave., SW
Washington, DC 20585

2 Kenneth Tatum
Sverdrup Tech. Inc./AEDC Group
740 Fourth Ave.
Arnold AFB, 37389-6001

1 Jack Benek
Micro Craft Inc.
P. O. Box 370
207 Big Springs Ave.
Tullahoma, TN 37388

1 NASA/Glen Research Center
Attn: Chris Steffen, MS 5-11
Cleveland, OH 44135

2 Hans Mair
Institute for Defense Analysis
Operational Evaluation Div.
1801 North Beauregard St.
Alexandria, VA 22311-1772

2 Achintya Haldar
Dept of Civil Engr. and Engr. Mech.
University of Arizona
Tucson, AZ 85721

2 William Reed
DP 51
Dept. of Energy
1000 Independence Ave., SW
Washington, DC 20585

2 Juan Meza
DP 51
Dept. of Energy
1000 Independence Ave., SW
Washington, DC 20585

2 Bilal Ayyub
Dept. of Civil Engineering
University of Maryland
College Park, MD 20742

1 Hugh Coleman
Dept. of Mech. and Aero. Engr.
Univ. of Alabama/Huntsville
Huntsville, AL 35899

2 Osman Balci
Dept. of Computer Science
Virginia Tech.
Blacksburg, VA 24061

2 Chun-Hung Chen
Dept. of Systems Engineering
University of Pennsylvania
220 South 33rd St.
Philadelphia, PA 19104-6315

2 T. P. Shivananda
Bldg. SB2/Rm. 1011
TRW/Ballistic Missiles Division
P. O. Box 1310
San Bernardino, CA 92402-1310

1 David Dolling
Dept. of Aerospace Engr. and
Engineering Mechanics
University of Texas at Austin
Austin, TX 78712-1085

2 Michael Ortiz
Mail Code 105-50
Graduate Aeronautical Laboratories
California Institute of Technology
Pasadena, CA 91125

2 Mike Giltrud
Defense Threat Reduction Agency
DTRA/CPWS
6801 Telegraph Road
Alexandria, VA 22310-3398

2 Isaac Elishakoff
Dept. of Mechanical Engineering
Florida Atlantic Univ.
777 Glades Rd.
Boca Raton, FL 33431-0991

2 Dale Pace
The Johns Hopkins University
111000 Johns Hopkins Road
Laurel, MD 20723-6099

2 Sankaran Mahadevan
Dept of Civil and Environmental Engr.
Vanderbilt University
Box 6077, Station B
Nashville, TN 37235

1 Steven Batill
Dept. of Aerospace and Mech. Engr.
University of Notre Dame
Notre Dame, IN 46556

1 Len Schwer
Schwer Engineering & Consulting
6122 Aaron Court
Windsor, CA 95492

2 Simone Youngblood
DMSO
Technical Director for VV&A
1901 N. Beauregard St., Suite 504
Alexandria, VA 22311

2 Ashok Singhal
CFD Research Corp.
Cummings Research Park
215 Wynn Drive
Huntsville, AL 35805

2 Raymond Cosner
Boeing-Phantom Works
MC S106-7126
PO Box 516
St. Louis, MO 63166-0516

1 Munir Sindir
MC IB39
Boeing-Rockwell Intl. Corp.
P. O. Box 7922
Canoga Park, CA 91309-7922

10 Stategic Systems Programs
Attn: Frank Dean
Nebraska Avenue Complex
287 Somers Court NW Suite 10041
Washington, DC 20393-5446

2 Richard Hills
College of Engineering, MSC 3449
New Mexico State University
P. O. Box 30001
Las Cruces, NM 88003

1 George Hazelrigg
Div. of Design, Manuf. and Innovation
Rm. 508N
4201 Wilson Blvd.
Arlington, VA 22230

2 Bernard Grossman
Dept. of Aerospace and Ocean Engr.
Mail Stop 0203
215 Randolph Hall
Blacksburg, VA 24061

1 Thomas Chwastyk
U. S. Naval Research Lab.
Code 6304
4555 Overlook Ave. SW
Washington, DC 20375-5343

2 Pradeep Raj
Computational Fluid Dynamics
Lockheed Martin Aeronautical Sys.
86 South Cobb Drive
Marietta, GA 30063-0685

1 James Glimm
Dept. of Applied Math. and Statistics
P138A
State Univ. of New York
Stony Brook, NY 11794-3600

2 Raphael Haftka
Dept. of AeMES
231 Aerospace Building
P.O. Box 116250
University of Florida
Gainesville, Fl 32611-6250

2 Efstratios Nikolaidis
Dept. of Aerospace and Ocean Engr.
Mail Stop 0203
215 Randolph Hall
Blacksburg, VA 24061

2 F. Owen Hoffman
SENES
102 Donner Drive
Oak Ridge, TN 37830

2 Leo Kadanoff
Research Institutes Building
University of Chicago
5640 South Ellis Ave.
Chicago, IL 60637

Foreign Distribution

2 Yakov Ben-Haim
Dept. of Mechanical Engineering
Technion-Israel Institute of Technology
Haifa 32000
ISRAEL

1 Igor Kozin
Systems Analysis Dept.
Riso National Laboratory
P. O. Box 49
DK-4000 Roskilde
DENMARK

1 Malcolm Wallace
National Engineering Laboratory
East Kilbride
Glasgow
G75 0QU
UNITED KINGDOM

2 Charles Hirsch
Dept. of Fluid Mechanics
Vrije Universiteit Brussel
Pleinlaan, 2
1050 Brussels
BELGUIM

Department of Energy Laboratories

29 Los Alamos National Laboratory
Mail Station 5000
P. O. Box 1663
Los Alamos, NM 87545
ATTENTION:
Peter Adams, MS B220
Thomas Bement, MS F600
Scott Doebling, MS P946
Dawn Flicker, MS F664
Sallie Keller-McNulty, MS F600
Ken Koch, MS F652
Harry Martz, MS F600
Mike McKay, MS F600
Tom Seed, MS F663
David Sharp, MS B213
Ronald E. Smith, MS J576
Ken Hanson, MS P940
Maysa Peterson-Schnell, MS B295
Dominic Cagliostro, MS F645
David Crane, MS P946
John F. Davis, MS B295
Rudolph Henninger, MS D413
Brad Holian, MS B268
Kathleen Holian, MS B295
Darryl Holm, MS B284
James Hyman, MS B284
Michael E. Jones, MS B259
James Kamm, MS D413
Elizabeth Kelly, MS F600
Len Margolin, MS D413
Mark R. Miller, MS P946
William Rider, MS D413
Daniel Weeks, MS B295
Morgan White, MS F663

10 Lawrence Livermore National Lab.
7000 East Ave.
P. O. Box 808
Livermore, CA 94550
ATTENTION:
Peter Brown, MS L-561
T. Scott Carman, MS L-031
Randy Christensen, MS L-160
Richard Klein, MS L-023
Roger Logan, MS L-125
G. Michael Murphy, MS L-650
Cynthia Nitta, MS L-096
Douglas Post, MS L-038
Peter Terrill, MS L-125
Kirk Levedahl, MS L-016

Internal Distribution

1	MS 0457	2001	W. J. Tedeschi
1	MS 0429	2100	J. S. Rottler
1	MS 0453	2103	M. W. Callahan
1	MS 0453	2104	D. L. McCoy
1	MS 0475	2105	R. C. Hartwig
1	MS 1393	2106	F. F. Dean
1	MS 0447	2111	J. O. Harrison
1	MS 0447	2111	P. D. Hoover
1	MS 0479	2151	P. A. Sena
1	MS 0479	2151	M. H. Abt
1	MS 0482	2161	V. J. Johnson
1	MS 0481	2167	M. A. Rosenthal
1	MS 0481	2167	W. C. Moffatt
1	MS 0481	2168	K. D. Meeks
1	MS 0481	2168	K. Ortiz
1	MS 0769	5800	D. S. Miyoshi
1	MS 0759	5845	I. V. Waddoups
1	MS 0759	5845	M. S. Tierney
1	MS 0782	5861	R. V. Matalucci
1	MS 0737	6114	P. A. Davis
1	MS 0751	6117	L. S. Costin
1	MS 0718	6141	C. D. Massey
1	MS 0708	6214	P. S. Veers
1	MS 0747	6410	R. L. Camp
1	MS 0747	6410	G. D. Wyss
1	MS 0746	6411	R. M. Cranwell
1	MS 0746	6411	D. J. Anderson
1	MS 0746	6411	J. E. Campbell
1	MS 0746	6411	D. G. Robinson
1	MS 0746	6411	L. P. Swiler
1	MS 1140	6500	J. K. Rice
1	MS 0977	6524	W. R. Cook
3	MS 0977	6524	S. M. DeLand
1	MS 1137	6534	A. L. Hodges
1	MS 1137	6535	G. K. Froehlich
1	MS 0771	6805	P. G. Kaplan
1	MS 1395	6821	J. W. Garner
1	MS 1395	6821	P. Vaughn
1	MS 0779	6849	J. C. Helton
1	MS 0779	6849	R. P. Rechard
1	MS 0779	6849	M. J. Shortencarier
1	MS 0778	6851	R. J. MacKinnon
1	MS 9405	8700	T. M. Dyer

1	MS 9042	8725	W. A. Kawahara
1	MS 9161	8726	E. P. Chen
1	MS 9003	8900	K. E. Washington
1	MS 9012	8920	P. E. Nielan
1	MS 9217	8950	M. L. Koszykowski
1	MS 1110	8950	L. J. Lehoucq
1	MS 0835	9101	T. C. Bickel
1	MS 0828	9102	R. K. Thomas
1	MS 0841	9102	J. A. Fernandez
1	MS 0835	9111	S. N. Kempka
1	MS 0835	9111	S. P. Burns
1	MS 0835	9111	R. J. Cochran
1	MS 0835	9111	B. Hassan
1	MS 0825	9111	W. P. Wolfe
1	MS 0834	9112	A. C. Ratzel
1	MS 0826	9113	W. Hermina
1	MS 0827	9114	J. E. Johannes
3	MS 0825	9115	W. H. Rutledge
1	MS 0825	9115	M. A. McWherter-Payne
1	MS 0825	9115	K. Salari
1	MS 0825	9115	L. W. Young
1	MS 0836	9116	E. S. Hertel
1	MS 0836	9116	L. A. Gritzo
1	MS 0835	9121	J. S. Peery
1	MS 0555	9122	M. S. Garrett
1	MS 0847	9123	H. S. Morgan
1	MS 0847	9123	A. F. Fossum
1	MS 0847	9124	D. R. Martinez
3	MS 0847	9124	K. F. Alvin
1	MS 0847	9124	R. V. Field
1	MS 0557	9125	T. J. Baca
1	MS 0553	9126	R. A. May
1	MS 0827	9131	J. D. Zepper
1	MS 0828	9132	J. L. Moya
1	MS 0828	9132	K. V. Chavez
1	MS 0828	9132	T. Y. Chu
3	MS 0828	9133	M. Pilch
1	MS 0828	9133	R. S. Baty
1	MS 0828	9133	B. F. Blackwell
1	MS 0828	9133	K. J. Dowding
1	MS 0828	9133	A. R. Lopez
1	MS 0828	9133	K. E. Metzinger
25	MS 0828	9133	W. L. Oberkampf
1	MS 0828	9133	T. L. Paez
1	MS 0828	9133	C. Romero
1	MS 0828	9133	V. J. Romero
1	MS 0828	9133	A. Urbina

1	MS 0828	9133	W. R. Witkowski
1	MS 1135	9134	D. B. Davis
1	MS 1135	9134	J. T. Nakos
1	MS 0321	9200	W. J. Camp
1	MS 0847	9211	M. S. Eldred
1	MS 0847	9211	J. R. Red-Horse
3	MS 0819	9211	T. G. Trucano
1	MS 1110	9222	D. E. Womble
1	MS 0847	9226	P. Knupp
1	MS 0820	9232	P. Yarrington
1	MS 0419	9800	R. G. Easterling
1	MS 0421	9814	J. M. Sjulin
1	MS 0423	9817	R. A. Paulsen
1	MS 0423	9817	S. E. Dingman
1	MS 9003	9900	D. L. Crawford
1	MS 0428	12000	D. D. Carlson
1	MS 0490	12331	J. A. Cooper
1	MS 0829	12323	F. W. Spencer
1	MS 0829	12323	M. L. Abate
3	MS 0829	12323	B. M. Rutherford
1	MS 0638	12326	D. E. Peercy
1	MS 0638	12326	D. L. Knirk
1	MS 0490	12331	P. E. D'Antonio
1	MS 0492	12332	D. R. Olson
1	MS 0405	12333	T. R. Jones
1	MS 0434	12334	R. J. Breeding
3	MS 0829	12335	K. V. Diegert
1	MS 1221	15002	R. D. Skocypec
1	MS 1179	15340	J. R. Lee
1	MS 0301	15400	J. L. McDowell
1	MS 9018	8940-2	Central Technical Files
2	MS 0899	9616	Technical Library
1	MS 0612	9612	Review & Approval Desk For DOE/OSTI