1
Introduction

This chapter briefly sketches the historical beginnings of modeling and simulation (M&S). Although claiming the beginning of anything is simply a matter of convenience, we will start with the stunning invention of calculus. We then discuss how the steadily increasing performance and decreasing costs of computing have been another critical driver in advancing M&S. Contributors to the credibility of M&S are discussed, and the preliminary concepts of verification and validation are mentioned. We close the chapter with an outline of the book and suggest how the book might be used by students and professionals.

1.1 Historical and modern role of modeling and simulation

1.1.1 Historical role of modeling and simulation

For centuries, the primary method for designing an engineered system has been to improve the successful design of an existing system incrementally. During and after the system was built, it would be gradually tested in a number of ways. The first tests would usually be done during the building process in order to begin to understand the characteristics and responses of the new system. This new system was commonly a change in the old system's geometrical character, materials, fastening techniques, or assembly techniques, or a combination of all of these changes. If the system was intended to be used in some new environment such as a longer bridge span, a taller structure, or propelled at higher speeds, the system was always tested first in environments where the experience base already existed. Often, during the building and testing process, design or assembly weaknesses and flaws were discovered and modifications to the system were made. Sometimes a catastrophic failure of a monumental project would occur and the process would start over: occasionally after attending the funeral of the previous chief designer and his apprentices (DeCamp, 1995). In ancient times, chief designers understood the consequences of a major design failure; they had skin in the game.

After the invention of calculus by Newton and Leibniz around 1700, the mathematical modeling of physics slowly began to have an impact on concepts for the understanding of nature and the design of engineered systems. The second key ingredient to have an impact on mathematical physics was the invention of logarithms by John Napier about 1594 (Kirby *et al.*, 1956). A mathematical model is of little practical use until it is exercised, which today is referred to as obtaining a *simulation result*. Until the existence and use of logarithms, it was not practical to conduct simulations on a routine basis. Then, not long after the invention of logarithms, the slide rule was invented by William Oughtred. This device provided a mechanical method for adding and subtracting logarithms and enabling rapid multiplication and division of numbers. The slide rule and mechanical calculators revolutionized not only simulation, but also such fields as surveying, navigation, and astronomy. Even though by today's standards the combination of mathematical theory and computing machines would be called "Before Computers," it provided the opportunity for the beginning of massive changes in science, engineering, and technology.

Starting with the Industrial Revolution, roughly around 1800 in England, the impact of modeling and simulation on engineering and design began to grow rapidly. However, during the Industrial Revolution, M&S was always an adjunct to experimentation and testing of engineered systems, always playing a minor support role. The primary reason for this was that computations were typically done by hand on a slide rule or mechanical calculator. By the early 1960s, programmable digital computers began to appear in a wide number of industrial, academic, and governmental organizations. During this time period, the number of arithmetic calculations commonly done for a simulation grew from hundreds or thousands to millions of calculations. It would be reasonable to identify the 1960s as the beginning of widespread scientific computing. In this book, we restrict the term *scientific computing* to the numerical solution of models given by partial differential equations (PDEs) or integro-differential equations. During the 1960s, computer power reached the level where scientific computing began to have a significant effect on the design and decision making of engineered systems, particularly aerospace and military systems. It is appropriate to view scientific computing as a field within the broader topic of M&S, which today includes systems that would have, for example, fundamental involvement with human behavior, such as economic and investment modeling, and individual and social modeling.

There were a few important exceptions, such as nuclear weapons design in the US, where scientific computing began to significantly influence designs in the 1940s and 1950s. The initial impetus for building much faster computers was the Cold War between the US and the Soviet Union. (See Edwards, 1997 for a perspective of the early history of electronic computing and their influence.) M&S activities were primarily modeling activities in the sense that models were simplified until it was realistic to obtain simulation results in an acceptable time period so as to have an impact on the design of a system or research activity. Relative to today's standards, these were extremely simplified models because there was relatively minimal computing power. This in no way denigrates the M&S conducted during the 1940s or the century before. Indeed, one could convincingly argue that the M&S conducted before the 1960s was more creative and insightful than present day scientific computing because the modeler had to sort carefully through what was physically and mathematically important to decide what could be ignored. This took great understanding, skill, and experience regarding the physics involved in the system of interest.

One of the most stunning scientific computing articles to appear during the 1960s was "Computer Experiments in Fluid Dynamics" by Harlow and Fromm (1965). This article, probably more than any other, planted the seed that scientific computing should be thought of as the third pillar of science, along with theory and experiment. During the 1970s and 80s, many traditionalists strongly resisted this suggestion, but that resistance faded as the power of scientific computing became dominant in advancing science and engineering. It is now widely accepted that scientific computing does indeed provide the third pillar of science and engineering and that it has its own unique strengths and weaknesses.

From a historical perspective, it should be recognized that we are *only beginning* to build this third pillar. One could argue that the pillar of experiment and measurement has been built, tested, and continually refined since the beginning of the Italian Renaissance in the 1400s. One could also argue that this pillar has much earlier historical roots with the Mesopotamian, Egyptian, Babylonian, and Indus Valley civilizations. The pillar of theory, i.e., theoretical physics, has been built, tested, and refined since the late 1700s. Understanding the strengths and weaknesses of each of these pillars has not come without major controversies. For example, the importance of uncertainty estimation in experimental measurements, particularly the importance of using different measurement techniques, is well understood and documented. History has shown, even in modern times, the bitter and sometimes destructive debates that occur when there is a paradigm shift, e.g., the shift from Newtonian mechanics to relativistic mechanics. In a century or so, when present day human egos and organizational and national agendas have faded, science and engineering will admit that the pillar of scientific computing is just now beginning to be constructed. By this we mean that the weaknesses and failings of all the elements contributing to scientific computing are beginning to be better understood. More importantly, the weaknesses and failings are often simply ignored in the quest for publicity and grabbing media headlines. However, we must learn to balance this youthful enthusiasm and naiveté with the centuries of experience and errors encountered during the building of the pillars of experiment and theory.

1.1.2 Changing role of scientific computing in engineering

1.1.2.1 Changing role of scientific computing in design, performance and safety of engineering systems

The capability and impact of scientific computing has increased at an astounding pace. For example, scientific simulations that were published in research journals in the 1990s are now given as homework problems in graduate courses. In a similar vein, what was at the competitive leading edge in scientific computing applied to engineering system design in the 1990s is now common design practice in industry. The impact of scientific computing has also increased with regard to helping designers and project managers improve their decision making, as well as in the assessment of the safety and reliability of manufactured products and public works projects. During most of this scientific computing revolution, system design and development were based primarily on testing and experience in the operating environment of the system, while scientific computing was commonly a secondary contributor in both preliminary and final design. For example, if there was some type of system failure, malfunction, or manufacturing issue that could not be solved quickly by testing, scientific computing was frequently called on for assistance and insight. Another common mode for the use of scientific computing was to reduce the number of design-then-test-then-redesign iterations that were needed for a product to perform better than competing products or to meet reliability or safety requirements. Specialized mathematical models for components or features of components were commonly constructed to better understand specific performance issues, flaws, or sensitivities of the components. For example, models were made to study the effect of joint stiffness and damping on structural response modes. Similarly, specialized mathematical models were built so that certain impractical, expensive, or restricted tests could be eliminated. Some examples were tests of high-speed entry of a space probe into the atmosphere of another planet or the structural failure of a full-scale containment vessel of a nuclear power plant.

As scientific computing steadily moves from a supporting role to a leading role in engineering system design and evaluation, new terminology has been introduced. Terminology such as *virtual prototyping* and *virtual testing* is now being used in engineering development to describe scientific computing used in the evaluation and "testing" of new components and subsystems, and even entire systems. As is common in the marketing of anything new, there is a modicum of truth to this terminology. For relatively simple components, manufacturing processes, or low-consequence systems, such as many consumer products, virtual prototyping can greatly reduce the time to market of new products. However, for complex, high-performance systems, such as gas turbine engines, commercial and military aircraft, and rocket engines, these systems continue to go through a long and careful development process based on testing, modification, and retesting. For these complex systems it would be fair to say that scientific computing plays a supporting role.

The trend toward using scientific computing more substantially in engineering systems is driven by increased competition in many markets, particularly aircraft, automobiles, propulsion systems, military systems, and systems for the exploration for oil and gas deposits. The need to decrease the time and cost of bringing products to market is intense. For example, scientific computing is relied on to reduce the high cost and time required to test components, subsystems, and complete systems. In addition, scientific computing is used in the highly industrialized nations of the world, e.g., the US, European Union, and Japan, to improve automated manufacturing processes. The industrialized nations increasingly rely on scientific computing to improve their competitiveness against nations that have much lower labor costs.

The safety aspects of products or systems also represent an important, sometimes dominant, element of both scientific computing and testing. The potential legal and liability costs of hardware failures can be staggering to a company, the environment, or the public.

This is especially true in the litigious culture of the US. The engineering systems of interest are both existing or proposed systems that operate, for example, at design conditions, off-design conditions, misuse conditions, and failure-mode conditions in accident scenarios. In addition, after the terrorist attacks on September 11, 2001, scientific computing is now being used to analyze and improve the safety of a wide range of civil systems that may need to function in hostile environments.

Scientific computing is used in assessing the reliability, robustness, and safety systems in two rather different situations. The first situation, which is by far the most common, is to *supplement* test-based engineering; for example, to supplement crash worthiness testing of automobiles to meet federal safety regulations. In fact, crash worthiness has become so important to some customers that automobile manufactures now use this feature in marketing their products. The second situation is to depend almost entirely on scientific computing for reliability, robustness, and safety assessment of high-consequence systems that cannot be tested in fully representative environments and scenarios; for example, failure of a large-scale dam due to an earthquake, explosive failure of the containment building of a nuclear power plant, underground storage of nuclear waste, and a nuclear weapon in a transportation accident. These types of high-consequence system analyses attempt to predict events that very rarely, if ever, occur subject to the design and intent of the system. That is, scientific computing is used to assess the reliability, robustness, and safety of systems where little or no direct experimental data exists.

For these types of situation, the burden of credibility and confidence that is required of scientific computing is dramatically higher than when scientific computing supplements test-based engineering. However, at this relatively early stage in the development of scientific computing, the methodologies and techniques for attaining this high level of credibility are not well developed, nor well implemented in engineering and risk assessment practice. Major improvements need to be made in the transparency, understandability, and maturity of all of the elements of scientific computing so that risk-informed decision making can be improved. Stated differently, decision makers and stakeholders need to be informed of the limitations, weaknesses, and uncertainties of M&S, as well as the strengths. The needed improvements are not just technical, but also cultural.

1.1.2.2 Interaction of scientific computing and experimental investigations

Interactions of scientific computing and experimental investigations have traditionally been very much one-way; from experiments to scientific computing. For example, experimental measurements were made and then mathematical models of physics were formulated, or experimental measurements were used to assess the accuracy of a simulation result. Given the limited capabilities of scientific computing until recently, this was an appropriate relationship. With the dramatic improvements in computing capabilities, however, the relationship between scientific computing and experiment is in the midst of change, although the changes have been slow and sometimes painful. When viewed from a historical as well as human perspective, the slow rate of change is perhaps understandable. Building the third pillar of science and engineering is viewed by some with vested interests in the established pillars of theory and experiment as a competitor, or sometimes a threat for resources and prestige. Sometimes the building of the scientific computing pillar is simply ignored by those who believe in the validity and preeminence of the established pillars. This view could be summarized as “Stay out of my way and don’t expect me to change the way that I have been conducting my research activities.” This attitude seriously undermines and retards the growth of scientific computing and its positive impact on science, engineering, and technology.

The fields of computational fluid dynamics (CFD) and computational solid mechanics (CSM) have pioneered many of the theoretical, practical, and methodological developments in scientific computing. The relationship between experiment and scientific computing in each of these fields, however, has been quite different. In CSM, there has been a long term and consistent tradition of a constructive and symbiotic relationship. Because of the nature of the physics modeled, CSM is fundamentally and critically dependent on experimental results for the construction of the physics models being used. To give a simple example, suppose one is interested in predicting the linear elastic modes of a built-up structure, e.g., a structure that is constructed from a number of individual beams that are fastened together by bolts. A mathematical model is formulated for the elastic beams in the structure and the joints between the beams are simply modeled as torsional springs and dampers. The stiffness and damping of the joints are treated as calibrated model parameters, along with the fluid dynamic and internal damping of the structure. The physical structure is built and then tested by excitation of many of the modes of the structure. Using the results of the experimental measurements, the stiffness and damping parameters in the mathematical model are optimized (calibrated) so that the results of the model best match the measurements of the experiment. It is seen in this example that the computational model could not be completed, in a practical way, without the experimental testing.

The relationship between experiment and CFD has not always been as collegial. Very early in the development of CFD, an article was published entitled “Computers vs. Wind Tunnels” (Chapman *et al.*, 1975). This article by influential leaders in CFD set a very negative and competitive tone early on in the relationship. One could certainly argue that the authors of this article simply gave voice to the brash claims of some CFD practitioners in the 1970s and 80s, such as “Wind tunnels will be used to store the output from CFD simulations.” These attitudes often set a competitive and frequently adversarial relationship between experimentalists and CFD practitioners, which has led to a lack of cooperation between the two groups. Where cooperation has occurred, it seems as often as not to have been due to small research teams forming voluntarily or in industrial settings where engineering project needs were paramount. There were several early researchers and technology leaders who properly recognized that such competition does not best serve the interests of either CFD practitioners or experimentalists (Bradley, 1988; Marvin, 1988; Neumann, 1990; Mehta, 1991; Dwoyer, 1992; Oberkampf and Aeschliman, 1992; Ashill, 1993; Lynch *et al.*, 1993; Cosner, 1994; Oberkampf, 1994).

As will be discussed at length in this book, the most efficient and rapid progress in scientific computing and experiment is obtained through a synergistic and cooperative environment. Although this may seem obvious to proponents in this viewpoint, there have been, and will remain, human and organizational attitudes that will work against this type of environment. There will also be practical issues that will hinder progress in both simulation and experiment. Here, we mention two examples of practical difficulties: one related to simulation and one related to experiment.

It is a commonly held view among scientific computing practitioners that comparison of computational results and experimental results, commonly referred to as the *validation* step, can be accomplished through comparison to existing data. These data are normally documented in corporate or institute reports, conference papers, and archival journal articles. Our experience, and that of many others, has shown that this approach is commonly less quantitative and precise than desired. Almost invariably, critical details are missing from published data, particularly for journal articles where discussion is limited in the interest of reducing article length. When important details, such as precise boundary conditions and initial conditions, are missing, the scientific computing practitioner commonly uses this lack of knowledge as freedom to adjust unknown quantities to obtain the best agreement with experimental data. That is, the comparison of computational results with experimental data begins to take on the character of a *calibration* of a model, as opposed to the evaluation of the predictive accuracy of the model. Many scientific computing practitioners will argue that this is unavoidable. We disagree. Although this calibration mentality is prevalent, an alternative methodology can be used which directly addresses the uncertainties in the simulation.

An important practical difficulty for experimentalists, particularly in the US, is that, with the rapid increase in the visibility and importance of simulation, many funding sources for experimental activities have evaporated. In addition, the attitude of many funding sources, both governmental and industrial, is that simulation will provide all of the important breakthroughs in research and technology, *not* experimental activities. This attitude over the last two decades has produced a decrease in the number of experimental research projects, including funding for graduate students, and a dramatic decrease in the number of experimental facilities. Also, with restricted funding for experimental activities, there is less research into the development of new experimental diagnostic techniques. We believe this has had an unintended detrimental effect on the growth of simulation. That is, with less high-quality experimental data available for validation activities, the ability to critically assess our computational results will decrease, or worse, we will have a false sense of confidence in our simulations. For example, major efforts are being initiated in multi-scale and multi-physics modeling. This type of modeling commonly bridges at least two spatial scales. Spatial scales are usually divided into the macro-scale (e.g., meter scale), the meso-scale (e.g., millimeter scale), the micro-scale (e.g., the micrometer scale), and the nano-scale (e.g., nanometer scale). The question that arises in mathematical model building or validation is: what new diagnostic techniques must be developed to obtain experimental data at multiple scales, particularly the micro- and nano-scales?

1.1.3 Changing role of scientific computing in various fields of science

Beginning around the 1960s, scientific computing has had an ever-increasing impact on a wide number of fields in science. The first that should be mentioned is computational physics. Although there is significant overlap between computational physics and computational engineering, there are certain areas of physics that are now dominated by simulation. Some examples are nuclear physics, solid state physics, quantum mechanics, high energy/particle physics, condensed matter physics, and astrophysics.

A second major area where simulation has become a major factor is environmental science. Some of the environmental areas where simulation is having a dominant impact are atmospheric science, ecology, oceanography, hydrology, and environmental assessment. Atmospheric science has received worldwide attention with the debate over global warming. Environmental assessment, particularly when it deals with long-term, underground storage of nuclear waste, has also achieved very high visibility. The predictions in fields such as global warming and underground storage of nuclear waste are extremely challenging because large uncertainties are present, and because the prediction time scales are on the order of tens of centuries. The accuracy of these predictions cannot be confirmed or falsified for many generations. Because of the widespread, potentially catastrophic effects studied in environmental science, the credibility of computational results is being scrutinized far more closely than in the past. Computational results can affect public policy, the well-being of entire industries, and the determination of legal liability in the event of loss of life or environmental damage. With this major level of impact of computational results, the credibility and uncertainty quantification in these areas must be greatly improved and standardized. If this is not done, hubris and the political and personal agendas of the participants will take precedence.

1.2 Credibility of scientific computing

1.2.1 Computer speed and maturity of scientific computing

The speed of computers over the last 50 years has consistently increased at a rate that can only be described as *stunning*. Figure 1.1 shows the increase in computing speed of the fastest computer in the world, the 500th fastest computer, and the sum of computing speed of the 500 fastest computers in the world as of November 2008. As can be seen, the speed of the fastest computer has consistently increased by roughly a factor of 10 every four years. Over the last few decades, many predicted that this rate of increase could not be maintained because of physics and technology constraints. However, the computer industry has creatively and consistently found ways around these hurdles and the steady increase in computing speed has been the real engine behind the increasing impact of computational simulation in science and engineering.

Measuring computer speed on the highest performance computers is done with a very carefully crafted set of rules, benchmark calculations, and performance measurements. Many people, particularly non-technically trained individuals, feel that computer speed

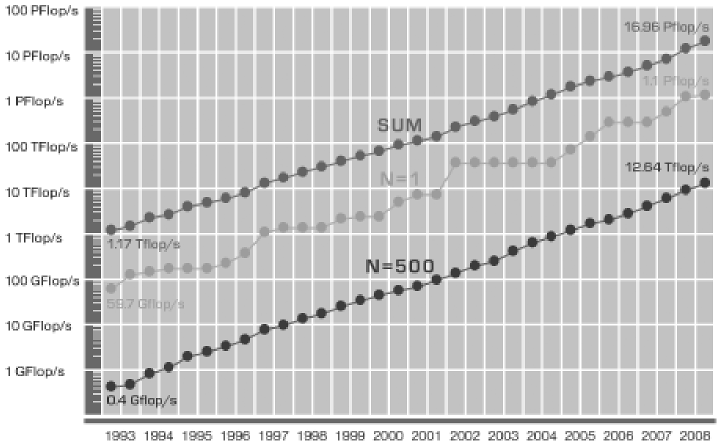

Figure 1.1 Computing speed of the 500 fastest computers in the world (Top500, 2008). See color plate section.

is directly related to maturity and impact of scientific computing. There is a relationship between computer speed and maturity and impact, but it is far from direct. Maturity of scientific computing clearly relates to issues such as credibility, trust, and reliability of the computational results. Impact of scientific computing relies directly on its trustworthiness, in addition to many other issues that depend on how the computational results are used. In industry, for example, some of the key perspectives are how scientific computing can reduce costs of product development, its ability to bring new products to market more quickly and improve profitability or market share, and the usability of results to improve decision making. In government, impact might be measured more in terms of how scientific computing improves risk assessment and the understanding of possible alternatives and unintended consequences. In academia, impact is measured in terms of new understanding and knowledge created from computational results.

In 1986, the US National Aeronautics and Space Administration (NASA) requested and funded a study conducted by the National Research Council to study the maturity and potential impact of CFD (NaRC, 1986). This study, chaired by Richard Bradley, was one of the first to examine broadly the field of CFD from a business and economic competitiveness perspective. They sent questionnaires to a wide range of individuals in industry and government to evaluate the maturity of CFD. Although they specifically examined CFD, we believe their high-level analysis is equally applicable to *any* field in scientific computing. In this study, the committee identified five stages of maturity of predictive capability. These stages, along with their descriptive characteristics, are:

- Stage 1: Developing enabling technology – scientific papers published, know-how developed.
- Stage 2: Demonstration and confidence building – limited pioneering applications, subject to surprise.
- Stage 3: Putting it all together – major commitment made, gradual evolution of capability.

- Stage 4: Learning to use effectively – changes the engineering process, value exceeds expectations, skilled user groups exist.
- Stage 5: Mature capability – dependable capability, cost effective for design applications, most analyses done without supporting experimental comparisons.

Using these descriptors for the various stages, the individuals ranked the maturity according to a matrix of elements. The matrix was formed on one side by increasing levels of complexity of the modeling approach to CFD, and on the other by increasing levels of complexity of engineering systems that would be of interest. A score of 0 meant that scientific papers have not been published and know-how has not been developed for that particular element in the matrix. A score of 5 meant that a mature capability existed – most analyses done without supporting experimental comparisons. What they found was, rather surprisingly, that depending on the model complexity and on the system complexity, the scores ranged from 0 to 5 over the matrix.

One would imagine that if the survey were conducted today, 20 years after the original survey, the maturity levels would be higher for essentially all of the elements in the matrix. However, there would still be a very wide range of scores in the matrix. Our point is that even within a well-developed field of scientific computing, such as CFD, the range of maturity varies greatly, depending on the modeling approach and the application of interest. Claims of high maturity in CFD for complex systems, whether from commercial software companies or any other organization, are, we believe, unfounded. Companies and agencies that sell programs primarily based on colorful graphics and flashy video animations have no skin in the game. We also claim that this is the case in essentially all fields of scientific computing.

1.2.2 Perspectives on credibility of scientific computing

People tend to think that their perspective on what is required for credibility or believability of an event or situation is similar to most other individuals. Broader experience, however, shows that this view is fundamentally mistaken. With regard to the present topic of scientific computing, there exists a wide range of perspectives regarding what is required for various individuals to say, “I believe this simulation is credible and I am comfortable making the needed decision.” In human nature, a key factor in decision making is the heavier weighting on downside risk as opposed to upside gain (Tversky and Kahneman, 1992; Kahneman and Tversky, 2000); that is, a person’s loss, pain, or embarrassment from a decision is weighed much more heavily than the expected gain. For example, when a decision must be made based on the results from a simulation, the individual’s perspective is weighted more heavily toward “What are the personal consequences of a poor decision because of a deficient or misleading simulation?” as opposed to “What are the personal gains that may result from a successful simulation?” If there is little downside risk, however, individuals and organizations can more easily convince themselves of the strengths of a simulation than its weaknesses. When an analyst is conducting the simulation, they will normally work toward attaining *their* required level of confidence in the simulation, given the time and resources available. If they are working in a team environment, each member of the team will have to make their own value judgment concerning their contribution to the combined result.

If the computational results are to be submitted to an archival journal for publication, the individual(s) authoring the work will ask themselves a slightly more demanding question: "Will other people (editor, reviewers, readers) believe the results?" This is usually a more demanding test of credibility of the work than the judgment of most individuals. Within the group of editor, reviewers, and readers, there will certainly be a wide range of perspectives toward the credibility of the work presented. However, the final decision maker concerning credibility and whether the article is published is the editor. In this regard, several well-known journals, such as the ASME Journal of Fluids Engineering, and all of the AIAA journals, have been increasing the requirements for credibility of computational results.

If the computational results are to be used as an important element in the decision making regarding some type of engineering project, then the issue of credibility of the computational results becomes more important. For this situation, presume that the individual computational analysts have satisfied themselves as to the credibility of their results. The engineering project manager must assess the credibility of the computational results based on their own personal requirements for credibility. This judgment will involve not only the computational results themselves, but also any knowledge he/she might have concerning the individuals involved in the analysis. Personal knowledge of the individuals involved is a great advantage to the project manager. However, for large-scale projects, or projects that have contributions from individuals or teams from around the country or around the world, this type of information is very rare. To judge the credibility of the computational results, some of the questions the project manager might ask are: "Am I willing to bet the success of my project (my career or my company) on these results?" These kinds of perspectives of the project manager are rarely appreciated by all of the contributors to the project.

Certain projects are of such magnitude that the effects of the success or failure of the project, or decisions made on the project, have major consequences beyond the project itself. For this type of situation, we refer to these as *high-consequence systems*. Two examples of these situations and the types of decisions made are the following.

- NASA uses scientific computing as part of it day-to-day activities in preparation for each of its Space Shuttle launches, as well as the safety assessment of major systems during each Space Shuttle flight. Individual analysts through high-level managers commonly ask themselves: "Am I willing the bet the lives of the flight crew on the results of this analysis?"
- The US Nuclear Regulatory Commission and the Environmental Protection Agency use scientific computing for analyzing the safety of nuclear power reactors and the underground storage of nuclear waste. These analyses commonly deal with the immediate effects of a possible accident, as well as the effects on the environment for thousands of years. For these situations, analysts and managers commonly ask themselves: "Am I willing to bet the public's safety and catastrophic damage to the environment for possibly centuries to come based on the results of this analysis?"

High-consequence systems are not mentioned to dramatize the importance of scientific computing in decision making, but to point out that there is a very wide range of impact of scientific computing. Typically, scientists and engineers work in one or two technical fields, e.g., research or system design. Rarely do individuals, especially those involved in academic research, consider the wide range of impact that scientific computing is having on real systems.

1.2.3 How is credibility built in scientific computing?

By *credibility* of computational results we mean that the results of an analysis are worthy of belief or confidence. The fundamental elements that build credibility in computational results are (a) quality of the analysts conducting the work, (b) quality of the physics modeling, (c) verification and validation activities, and (d) uncertainty quantification and sensitivity analyses. We believe that all of these elements are necessary for credibility, and more importantly accuracy, but none is sufficient in itself. Our perspective in discussing these elements is that *scientific computing is a tool for generating information about some physical situation, process, or system*. This information could be used in a wide variety of ways, some of which were discussed in the previous section. The quality of the information depends on how it was developed, but the quality of the decisions made based on the information depends on many other factors. Two key factors are the user's depth of understanding of the information produced and the appropriateness of the information for its intended use. Although it is beyond the scope of this book to discuss how the information might be used, methods for clarifying how the information should be used and methods to reduce the possible misuse of the information will be discussed later in the text.

1.2.3.1 Quality of the analysts conducting the scientific computing

When we refer to *analysts*, we mean the group of individuals that: (a) construct the conceptual model for the problem of interest, (b) formulate the mathematical model, (c) choose the discretization and numerical solution algorithms, (d) program the software to compute the numerical solution, (e) compute the simulation on a digital computer, and (f) analyze and prepare the results from the simulation. On small-scale analyses of subsystems or components, or on research projects, a single individual may conduct all of these tasks. On any significantly sized effort, a team of individuals conducts all of these tasks.

The quality of the analysts encompasses their training, experience, sound technical judgment, and understanding of the needs of the customer of the computational information. Some have expressed the view that the quality of a computational effort should be entirely centered on the quality of the analysts involved. For example, it has been said, "I have such confidence in this analyst that whatever simulation he/she produces, I'll believe it." No one would argue against the extraordinary value added by the quality and experience of the analysts involved. However, many large projects and organizations have learned, often painfully, that they cannot completely depend on a few extraordinarily talented individuals for long-term success. Large organizations must develop and put into place business, technology, training, and management processes for all the elements that contribute to the quality and on-time delivery of their product. In addition, many modern large-scale projects will typically involve groups that are physically and culturally separated, often around the nation or around the world. For these situations, users of the computational information will have minimal personal knowledge of the individual contributors, their backgrounds, or value systems.

1.2.3.2 Quality of the physics modeling

By *quality of the physics modeling*, we mean the fidelity and comprehensiveness of physical detail embodied in the mathematical model representing the relevant physics taking place in the system of interest. These modeling decisions are made in the formulation of the conceptual and mathematical model of the system of interest. Two contrasting levels of physics model fidelity are (a) at the low end, fully empirical models that are entirely built on statistical fits of experimental data with no fundamental relationship to physics-based principles; and (b) at the high end, physics-based models that are reliant on PDEs or integro-differential equations that represent conservation of mass, momentum, and energy in the system. Comprehensiveness of the modeling refers to the number of different types of physics modeled in the system, the level of coupling of the various physical processes, and the extent of possible environments and scenarios that are considered for the system.

We are *not* saying that the highest possible level of quality of physics modeling should be used for every computational activity. Efficiency, cost effectiveness, and schedule should dictate the appropriate level of physics modeling to meet the information needs of the scientific computing customer. Stated differently, the analysts should understand the needs of the scientific computing customer and then decide on the simplest level of physics modeling fidelity that is needed to meet those needs. To accomplish this requires significant experience on the part of the analysts for the specific problem at hand, very clear communication with the customer of what they think they need, and how they intend to use the results of the computational effort. Quality of the physics modeling is a very problem-specific judgment. It is *not* one size fits all.

1.2.3.3 Verification and validation activities

Verification is the process of assessing software correctness and numerical accuracy of the solution to a given mathematical model. *Validation* is the process of assessing the physical accuracy of a mathematical model based on comparisons between computational results and experimental data. Verification and validation (V&V) are the primary processes for assessing and quantifying the accuracy of computational results. The perspective of V&V is distinctly on the side of skepticism, sometimes to the degree of being radical (Tetlock, 2005). In verification, the association or relationship of the simulation to the real world is *not* an issue. In validation, the relationship between the mathematical model and the real world (experimental data) *is* the issue. Blottner (1990) captured the essence of each in the compact expressions: "Verification is solving the equations right;" "Validation is solving the right equations." These expressions follow the similar definitions of Boehm (1981).

The pragmatic philosophy of V&V is fundamentally built on the concept of *accuracy assessment*. This may sound obvious, but in Chapter 2, Fundamental Concepts and Terminology, it will become clear that there are wide variations on the fundamental concepts of V&V. In our present context, it is clear how accuracy assessment is a necessary building block of "How is credibility built in scientific computing?" V&V do not answer the entire question of simulation credibility, but they are the key contributors. V&V could be described as processes that develop and present evidence of the accuracy of computational results. To measure accuracy, one must have accurate benchmarks or reference values. In verification, the primary benchmarks are highly accurate solutions to specific mathematical models. In validation, the benchmarks are high-quality experimental measurements. Given this perspective of V&V, it should be pointed out that a critical additional element is needed: estimation of accuracy when no benchmark is available.

The pivotal importance of V&V in the credibility of scientific computing was captured by Roache (2004) when he said

> In an age of spreading pseudoscience and anti-rationalism, it behooves those of us who believe in the good of science and engineering to be above reproach whenever possible. Public confidence is further eroded with every error we make. Although many of society's problems can be solved with a simple change of values, major issues such as radioactive waste disposal and environmental modeling require technological solutions that necessarily involve computational physics. As Robert Laughlin noted in this magazine, "there is a serious danger of this power [of simulations] being misused, either by accident or through deliberate deception." Our intellectual and moral traditions will be served well by conscientious attention to verification of codes, verification of calculations, and validation, including the attention given to building new codes or modifying existing codes with specific features that enable these activities.

1.2.3.4 Uncertainty quantification and sensitivity analyses

Uncertainty quantification is the process of identifying, characterizing, and quantifying those factors in an analysis that could affect the accuracy of the computational results. Uncertainties can arise from many sources, but they are commonly addressed in three steps of the modeling and simulation process: (a) construction of the conceptual model, (b) formulation of the mathematical model, and (c) computation of the simulation results. Some common sources of uncertainty are in the assumptions or mathematical form of either the conceptual or mathematical model, the initial conditions or boundary conditions for the PDEs, and the parameters occurring in the mathematical model chosen. Using the computational model, these sources of uncertainty are propagated to uncertainties in the simulation results. By *propagated* we mean that the sources of uncertainty, wherever they originate, are mathematically mapped to the uncertainties in the simulation results. The primary responsibility for identifying, characterizing and quantifying the uncertainties in a simulation is the team of analysts involved in conjunction with the customer for the simulation results.

Sensitivity analysis is the process of determining how the simulation results, i.e., the outputs, depend on all of the factors that make up the model. These are usually referred to as *inputs* to the simulation, but one must recognize that sensitivity analysis also deals with the question of how outputs depend on assumptions, or mathematical models, in the analysis. Uncertainties due to assumptions or choice of mathematical models in an analysis are typically referred to as model form, or model structure, uncertainties. Uncertainty quantification and sensitivity analysis critically contribute to credibility by informing the user of the simulation results how uncertain the results are and what factors are the most important in uncertain results.

1.3 Outline and use of the book

1.3.1 Structure of the book

The book is structured into five parts. Part I: Fundamental concepts (Chapters 1–3) deals with the development of the foundational concepts of verification and validation (V&V), the meaning of V&V that has been used by different communities, fundamentals of modeling and simulation, and the six phases of computational simulation. Part II: Code verification (Chapters 4–6) deals with how code verification is closely related to software quality assurance, different methodological approaches to code verification, traditional methods of code verification, and the method of manufactured solutions. Part III: Solution verification (Chapters 7–9) covers fundamental concepts of solution verification, iterative convergence error, finite-element-based error estimation procedures, extrapolation-based error estimation procedures, and practical aspects of mesh refinement. Part IV: Model validation and prediction (Chapters 10–13) addresses the fundamental concepts of model validation, the design and execution of validation experiments, quantitative assessment of model accuracy using experimental data, and discusses the six steps of a nondeterministic model prediction. Part V: Planning, management, and implementation issues (Chapters 14–16) discusses planning and prioritization of modeling activities and V&V, maturity assessment of modeling and simulation, and finally, development and responsibilities of V&V and uncertainty quantification.

The book covers the fundamental issues of V&V as well as their practical aspects. The theoretical issues are discussed only in as far as they are needed to implement the practical procedures. V&V commonly deals with quality control concepts, procedures, and best practices, as opposed to mathematics and physics issues. Our emphasis is on how V&V activities can improve the quality of simulations and, as a result, the decisions based on those simulations. Since V&V is still a relatively new field of formal technology and practice, there are commonly various methods and divergent opinions on many of the topics discussed. This book does not cover every approach to a topic, but attempts to mention and reference most approaches. Typically, one or two approaches are discussed that have proven to be effective in certain situations. One of our goals is to provide readers with enough detail on a few methods so they can be used in practical applications of scientific computing. Strengths and weaknesses of methods are pointed out and cautions are given where methods should not be used. Most chapters discuss an example application of the principles discussed in the chapter. Some of the examples are continually developed throughout different chapters of the book as new concepts are introduced.

The field of V&V is not associated with specific application areas, such as physics, chemistry, or mechanical engineering. It can be applied to essentially any application domain where M&S is used, including modeling of human behavior and financial modeling. V&V is a fascinating mixture of computer science, numerical solution of PDEs, probability, statistics, and uncertainty estimation. Knowledge of the application domain clearly influences what V&V procedures might be used on a particular problem, but the application domain is not considered part of the field of V&V. It is presumed that the practitioners of the application domain bring the needed technical knowledge with them.

The book is written so that it can be used either as a textbook in a university semester course, or by professionals working in their discipline. The emphasis of the book is directed toward models that are represented by partial differential equations or integro-differential equations. Readers who are only interested in models that are represented by ordinary differential equations (ODEs) can use all of the fundamental principles discussed in the book, but many of the methods, particularly in code and solution verification, will not be applicable. Most parts of the book require some knowledge of probability and statistics. The book does not require that any particular computer software or programming language be used. To complete some of the examples, however, it is beneficial to have general purpose software packages, such as MATLAB or Mathematica. In addition, general-purpose software packages that solve PDEs would also be helpful for either completing some of the examples or for the reader to generate his/her own example problems in their application domain.

1.3.2 Use of the book in undergraduate and graduate courses

For senior-level undergraduates to get the most out of the book, they should have completed courses in introduction to numerical methods, probability and statistics, and analytical solution methods for PDEs. Chapters of interest, at least in part, are Chapters 1–4 and 10–13. Depending on the background of the students, these chapters could be supplemented with the needed background material. Although some elements of these chapters may not be covered in depth, the students would learn many of the fundamentals of V&V, and, more generally, what the primary issues are in assessing the credibility of computational results. Ideally, homework problems or semester projects could be given to teams of individuals working together in some application area, e.g., fluid mechanics or solid mechanics. Instead of working with PDEs, simulations can be assigned that only require solution of ODEs.

For a graduate course, all the chapters in the book could be covered. In addition to the courses just mentioned, it is recommended that students have completed a graduate course in the numerical solution of PDEs. If the students have not had a course in probability and statistics, they may need supplementary material in this area. Assigned homework problems or semester projects are, again, ideally suited to teams of individuals. A more flexible alternative is for each team to pick the application area for their project, with the approval of the instructor. Our view is that teams of individuals are very beneficial because other team members experienced in one area can assist any individual lacking in knowledge in those areas. Also, learning to work in a team environment is exceptionally important in any science or engineering field. The semester projects could be defined such that each element of the project builds on the previous elements completed. Each element of the project could deal with specific topics in various chapters of the book, and each could be graded separately. This approach would be similar in structure to that commonly used in engineering fields for the senior design project.

1.3.3 Use of the book by professionals

Use of the book by professionals working in their particular application area would be quite different than a classroom environment. Professionals typically scan through an entire book, and then concentrate on particular topics of interest at the moment. In the following list, five groups of professionals are identified and chapters that may be of particular interest are suggested:

- code builders and software developers: Chapters 1–9;
- builders of mathematical models of physical processes: Chapters 1–3 and 10–13;
- computational analysts: Chapters 1–16;
- experimentalists: Chapters 1–3 and 10–13;
- project managers and decision makers: Chapters 1–3, 5, 7, 10, and 13–16.

1.4 References

Ashill, P. R. (1993). Boundary flow measurement methods for wall interference assessment and correction: classification and review. *Fluid Dynamics Panel Symposium: Wall Interference, Support Interference, and Flow Field Measurements, AGARD-CP-535*, Brussels, Belgium, AGARD, 12.1–12.21.

Blottner, F. G. (1990). Accurate Navier–Stokes results for the hypersonic flow over a spherical nosetip. *Journal of Spacecraft and Rockets*. **27**(2), 113–122.

Boehm, B. W. (1981). *Software Engineering Economics*, Saddle River, NJ, Prentice-Hall.

Bradley, R. G. (1988). CFD validation philosophy. *Fluid Dynamics Panel Symposium: Validation of Computational Fluid Dynamics, AGARD-CP-437*, Lisbon, Portugal, North Atlantic Treaty Organization.

Chapman, D. R., H. Mark, and M. W. Pirtle (1975). Computer vs. wind tunnels. *Astronautics & Aeronautics*. **13**(4), 22–30.

Cosner, R. R. (1994). Issues in aerospace application of CFD analysis. *32nd Aerospace Sciences Meeting & Exhibit, AIAA Paper 94–0464*, Reno, NV, American Institute of Aeronautics and Astronautics.

DeCamp, L. S. (1995). *The Ancient Engineers*, New York, Ballantine Books.

Dwoyer, D. (1992). The relation between computational fluid dynamics and experiment. *AIAA 17th Ground Testing Conference*, Nashville, TN, American Institute of Aeronautics and Astronautics.
Edwards, P. N. (1997). *The Closed World: Computers and the Politics of Discourse in Cold War America*, Cambridge, MA, The MIT Press.
Harlow, F. H. and J. E. Fromm (1965). Computer experiments in fluid dynamics. *Scientific American*. **212**(3), 104–110.
Kahneman, D. and A. Tversky, Eds. (2000). *Choices, Values, and Frames*. Cambridge, UK, Cambridge University Press.
Kirby, R. S., S. Withington, A. B. Darling, and F. G. Kilgour (1956). *Engineering in History*, New York, NY, McGraw-Hill.
Lynch, F. T., R. C. Crites, and F. W. Spaid (1993). The crucial role of wall interference, support interference, and flow field measurements in the development of advanced aircraft configurations. *Fluid Dynamics Panel Symposium: Wall Interference, Support Interference, and Flow Field Measurements, AGARD-CP-535*, Brussels, Belgium, AGARD, 1.1–1.38.
Marvin, J. G. (1988). Accuracy requirements and benchmark experiments for CFD validation. *Fluid Dynamics Panel Symposium: Validation of Computational Fluid Dynamics, AGARD-CP-437*, Lisbon, Portugal, AGARD.
Mehta, U. B. (1991). Some aspects of uncertainty in computational fluid dynamics results. *Journal of Fluids Engineering*. **113**(4), 538–543.
NaRC (1986). *Current Capabilities and Future Directions in Computational Fluid Dynamics*, Washington, DC, National Research Council.
Neumann, R. D. (1990). CFD validation – the interaction of experimental capabilities and numerical computations, *16th Aerodynamic Ground Testing Conference, AIAA Paper 90–3030*, Portland, OR, American Institute of Aeronautics and Astronautics.
Oberkampf, W. L. (1994). A proposed framework for computational fluid dynamics code calibration/validation. *18th AIAA Aerospace Ground Testing Conference, AIAA Paper 94–2540*, Colorado Springs, CO, American Institute of Aeronautics and Astronautics.
Oberkampf, W. L. and D. P. Aeschliman (1992). Joint computational/experimental aerodynamics research on a hypersonic vehicle: Part 1, experimental results. *AIAA Journal*. **30**(8), 2000–2009.
Roache, P. J. (2004). Building PDE codes to be verifiable and validatable. *Computing in Science and Engineering*. **6**(5), 30–38.
Tetlock, P. E. (2005). *Expert Political Judgment: How good is it? How can we know?*, Princeton, NJ, Princeton University Press.
Top500 (2008). *32nd Edition of TOP500 Supercomputers*. www.top500.org/.
Tversky, A. and D. Kahneman (1992). Advances in prospect theory: cumulative representation of uncertainty. *Journal of Risk and Uncertainty*. **5**(4), 297–323.