14

Planning and prioritization in modeling and simulation

In Chapter 2, Fundamental concepts and terminology, a summary discussion was given concerning an integrated view of verification, validation, and prediction. This chapter will discuss in detail Element 2, Planning and prioritization of activities, as shown in Figure 14.1. The topic of this chapter could be summarized as: given the wide range of activities dealing with modeling and simulation, how does one allocate resources to best achieve the goals of the project? Here, we are interested in dealing with the perspective of the management responsibilities for a large-scale, project-oriented activity as opposed to a research effort or general capability development of a commercial software package. Our discussion applies to projects within both industry and government. We emphasize engineering system projects, but the discussion also applies to the analysis of natural systems, e.g., underground storage of radioactive wastes, global climate change, and transport of contaminants or chemical agents due to atmospheric winds. The system of interest could be a new or proposed system in the design phase, an existing system that is being considered for modification or upgrade, or analysis of a system as it presently exists.

14.1 Methodology for planning and prioritization

For large-scale projects, planning and prioritization (P&P) are major activities that require a significant investment in time, money, and specialized personnel talent. Although we are primarily concerned with P&P in V&V and prediction, we comment here on the importance of P&P in large-scale projects for the development of an M&S capability. Depending on the type of organization developing the capability, it can be challenging for management to assemble the personnel with the needed talent and interest in P&P. For example, if a large-scale M&S project arrives in the midst of ongoing computational development activities, it can be difficult for management and staff to change pre-existing viewpoints, traditions and ingrained habits that are no longer appropriate, and possibly even very detrimental, to the goals of the new project. Needed changes can be particularly onerous to affect in a research or government organization.

It is a widely held view that the crucial importance of P&P is commonly ignored, almost always with dire consequences (Dorner, 1989; Flowers, 1996; Stepanek, 2005; Vose, 2008).

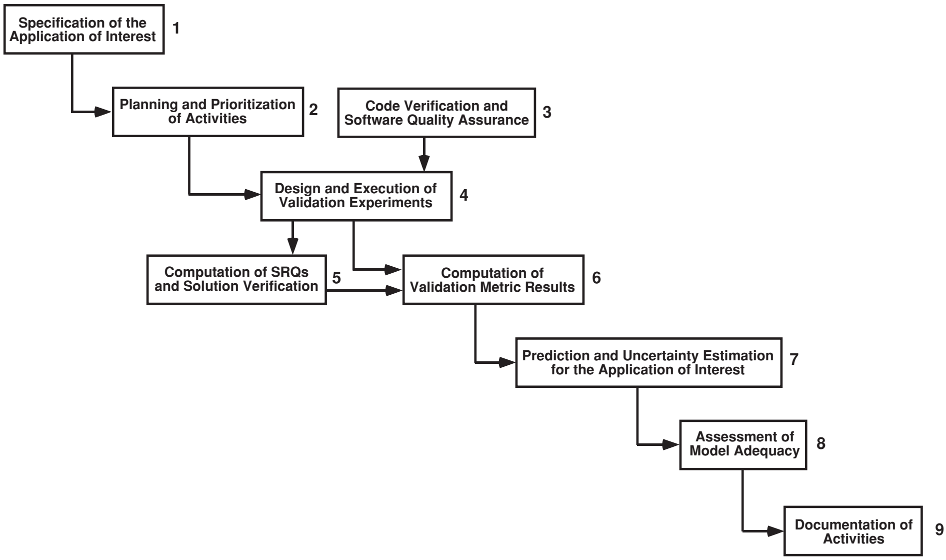


Figure 14.1 Integrated view of the elements of verification, validation, and prediction (adapted from Trucano *et al.*, 2002).

In an article in *IEEE Spectrum* in 2005, Charette (2005) states, "This year, organizations and governments will spend an estimated $1 trillion on IT hardware, software, and services worldwide. Of the IT projects that are initiated, from 5 to 15 percent will be abandoned before or shortly after delivery as hopelessly inadequate. Many others will arrive late and over budget or require massive reworking. Few IT projects, in other words, truly succeed." He lists a dozen of the most common reasons for massive software failures, but several of these can be grouped as a lack of P&P in the software project.

A number of very helpful texts are available that deal with P&P of software projects (Flowers, 1996; Karolak, 1996; Galin, 2003; Wiegers, 2003). Here, we concentrate on the broader aspects of projects that have been discussed, e.g., modeling and experimental activities, not just the software aspects. Another well-developed approach used in P&P and decision making in complex situations is the Analytical Hierarchy Process developed primarily by Saaty (2001). The method has been developed over the last three decades and has been used in a wide variety of applications in business, government, policy decisions, economic planning, military operations, service delivery, and environmental impact. The Analytical Hierarchy Process, however, has not yet been applied within the relatively new framework for V&V.

14.1.1 Planning for a modeling and simulation project

Planning in a large-scale M&S project begins by specifying the application of interest, the first element shown in Figure 14.1. Sometimes the activities in this element are referred to as development and specification of customer requirements. These activities should involve significant interaction between the customer and the development team. The following summarizes the most important topics to be addressed *before* prioritization of activities can begin.

- Specification of the application of interest. The key physical process, engineering system, or event of interest should be well defined. The expected informational-contributions or value added from the computational effort should also be specified.
- Specification of the application domain of the system that the computational effort is expected to address. The application domain could be restricted to the normal environment, e.g., the operating envelope of the system, but it could also include certain abnormal and hostile environments of the system.
- Identification of important scenarios for each of the specified environments. The scenarios correspond to conditions that the system could be exposed to within a specified environment, various event sequences, event trees, or fault trees.
- Specification of the system and it surroundings. Different specifications for the system and the surroundings could be given, depending on the environments and scenarios of interest.
- Specification of the system response quantities of interest. This should include all SRQs for each of the environments and scenarios of interest. For the normal environment, typical SRQs of interest are related to system performance and reliability; for abnormal and hostile environments, SRQs are usually related to some aspect of safety or security of the system.
- Specification of the accuracy requirements by the customer or stakeholder for each SRQ of interest. If the system specification is at an early stage, or the environments and scenarios are not well defined, then these requirements may be very poorly known. For example, accuracy requirements of some SRQs may only be known to within an order of magnitude, or just a qualitative feature of the SRQ may be specified. Even though the customer may poorly know the accuracy requirements, recognizing that they may be modified later because of the cost and schedule involved, it is of utmost importance that initial expectations be discussed.

Completing these tasks will require a significant effort by a diverse group of customer representatives and project managers and staff. It will require in-depth, and often difficult, discussions between the project managers, the customer, and stakeholders in the effort. By *customers* we mean both those who will use the software produced by the project and those who will directly use the results of the analyses, e.g., decision makers. By *stakeholders* we mean those who have any type of vested interest in the development of the simulation capability, those who use the simulation results as secondary information in their activities, and those whose success is dependent on the success of the simulation capability or the customer's success. The funding source for the simulation capability is usually the customer, but in certain situations, e.g., governmental funding, this is not always the case. When the funding source is *not* the customer, the likelihood of confusion, misunderstandings, delivering the wrong product, and failure increases dramatically.

The discussions between the project managers, customer, and stakeholders will involve a great deal of clarification of issues, negotiation of trade-offs, and compromise. Sometimes the customer only has a vague idea of needed requirements because either he has not carefully thought about his requirements, or because the application of interest has not yet been fully defined. Many times the customer will request more than what is realistic, given the time and money available for the effort. The manager of the effort must therefore be realistic in what can be delivered, given the resources available. The cause of many software project failures is that the project manager either (a) is initially unrealistic in what could be accomplished; or (b) allows significant changes or increases in requirements, features, or capabilities to be imposed during software development. During the negotiation of the specifications given in the bullet list above, there must be flexibility by the customer and the project management, an open exchange of information concerning costs for various capabilities desired, and the customer clearly expressing his value system concerning trade-offs of deliverables. For example, the project manager must try to assess the time and costs required to achieve certain customer accuracy requirements. On the other hand, the customer must be flexible in trading-off one requirement of lesser value for requirements that are more important to his goals for the performance of the system.

For some projects, it will already be specified which computer code, or group of codes, will be used in the various analyses of the system of interest. So that, in addition to the activities listed above, there will also be significant specificity of many of the additional elements listed for the verification, validation, and prediction shown in Figure 14.1, as well as the additional phases of project discussed in Chapter 3, Modeling and computational simulation. For example, if the project is using commercial software, then the planning phase will deal much more about comparing customer requirements with options available in the software. In commercial software there would typically be more detailed documentation available concerning available user options, compared to software written and supported by a corporate or government organization.

14.1.2 Value systems for prioritization

Here we are interested in value systems for prioritizing future work on activities such as physics modeling, software development, software quality assurance (SQA), code verification, solution verification, experiments associated with validation, construction of validation metrics, and uncertainty quantification (UQ) associated with predictions. Our interest is in attempting to best allocate resources between various activities to achieve the highest possible level of success of a project-oriented M&S effort, given the constraints on resources. It should be recognized that the success of the M&S effort is not synonymous with success of the engineering system project. The success of the engineering project typically depends on many other factors not of interest here. By *success of the M&S effort* we mean an effort that best contributes to assisting (a) the engineering system to attain its performance goals, or (b) the customer to attain his informational needs.

By *resources* we primarily mean the time, money, personnel expertise, and facilities required to complete an activity. These will be grouped together for convenience, because we will not deal with individual types of resource in any detail. It should be recognized throughout this discussion that there are strong dependencies between each of these resources. For example, there is typically a strong connection between the personnel that conduct SQA activities and those that conduct code verification activities, i.e., they may be the same people. Likewise, there is commonly overlapping expertise between physics model builders and software developers. Here, we will not deal with these types of dependency in resources types.

There are various value systems that can be used to prioritize project-oriented activities. In some projects, however, the prioritization of activities has little to do with project-oriented goals, but with the organizational power base of groups or the physics interests of groups within the organization conducting the M&S effort. Since there are many different types of activities that are conducted throughout the elements shown in Figure 14.1, one needs to map all of the activities back to one common feature that links all of the elements. The most logical feature to use is physics modeling, i.e., mathematical modeling of the various physical processes occurring in the system of interest. Without this common feature, the remainder of the activities would have little meaning or effect on the goals of the M&S effort.

Here we will discuss three of the more common value systems and where each may be appropriate in optimizing resource allocation.

1 Rank the modeling of individual physical processes occurring in the system according to the expected level of *inaccuracy*. In this approach, the highest rank would be given to those physical processes that are expected to be modeled the *least* accurately. This prioritization scheme is based on the anticipated level of understanding of the physics processes and their interactions within the system. This approach would be appropriate for improving the understanding of complex systems and multi-physics processes, as well as systems exposed to abnormal or hostile environments, such as severe accident conditions, highly damaged systems, electromagnetic pulse attack on a computer controlled system, and a terrorist attack on a public transportation system. This approach would *not* be appropriate for engineering systems in normal environments, such as well-understood operating conditions.
2 Rank according to the expected impact of individual physical processes on SRQs of interest. In this approach, the highest rank would be given to those physical processes that have the *largest expected effect* on specific SRQs related to system performance, safety, or reliability. This approach would also be appropriate for improving the understanding of complex physics processes in abnormal and hostile environments. Here, however, the focus is ranking the impact of the physics processes on specific system responses, not just their impact on understanding the physics of the system. The most well-developed method using this prioritization scheme is the phenomena identification and ranking table (PIRT). It was developed by several organizations for the US Nuclear Regulatory Commission (NRC) to improve assessment of nuclear reactor safety in abnormal, i.e., accident, environments. The PIRT approach will be discussed in Section 14.2.
3 Rank according to the expected ability to predict the impact of individual physical processes on SRQs of interest. This is a two-step method where one would first conduct approach 2 just mentioned, and then focus on the more important physical processes to determine if the modeling effort can adequately address these phenomena. This method concentrates on the *weaknesses* in the modeling effort to predict the important SRQs related to system performance, safety, or

reliability. As a result, this prioritization scheme is commonly referred to as the *gap analysis method*. Although this approach was initiated by several organizations for the NRC, it has recently been extended as part of the Advanced Simulation and Computing (ASC) Program (previously called the Accelerated Strategic Computing Initiative) sponsored by the US Department of Energy. It has been used for normal, abnormal, and hostile environments by Sandia National Laboratories. This method will be discussed in Section 14.3.

14.2 Phenomena identification and ranking table (PIRT)

The nuclear reactor safety community, as it has done in many areas of risk assessment, developed a new process for improving the understanding of nuclear plants in accident scenarios. In 1988, the NRC issued a revised emergency core cooling system rule for light water reactors that allows, as an option, the use of best estimate plus uncertainty (BE+U) methods in safety analyses (NRC, 1988). To support the licensing revision, the NRC and its contractors developed the code scaling, applicability, and uncertainty (CSAU) evaluation methodology to demonstrate the feasibility of the BE+U approach. The phenomena identification and ranking table was developed in support of the CSAU methodology. The PIRT process was initially developed by Shaw *et al.* (1988) to help analyze the safety of pressurized water nuclear reactors during a loss of coolant accident. Since its initial development, the process has been developed further and applied many times to various nuclear reactor designs. For a detailed discussion of PIRTs, see Boyack *et al.* (1990); Wilson *et al.* (1990); Wulff *et al.* (1990); Hanson *et al.* (1992); Rohatgi *et al.* (1997); Kroeger *et al.* (1998); and Wilson and Boyack (1998).

During the 1990s, it was found that the PIRT process was a much more powerful tool than originally conceived (Wilson and Boyack, 1998). It was found that it could also be used to aid in identifying needed future experiments, physics model development, and needed improvements in UQ. The additional objectives, however, were still focused on improving the physical understanding of the plant behavior and the interactions of systems and subsystems during a specific accident scenario. The generalized PIRT process, as described by Wilson and Boyack (1998), is a 15-step process that requires significant resources, if completed in detail.

As part of the ASC program, various researchers and projects at Sandia National Laboratories modified the PIRT process so that it was more oriented towards modeling activities (Pilch *et al.*, 2001; Tieszen *et al.*, 2002; Trucano *et al.*, 2002; Boughton *et al.*, 2003). In addition, they simplified the PIRT process to five steps so that it could be used by individual code development projects. The PIRT process has focused on abnormal and hostile environments because these environments typically involve large uncertainties in (a) knowing the condition, state, or geometry of the system; (b) the surroundings as they affect the system through the BCs or the excitation function; (c) strongly coupled physics that commonly occur; and (d) important human or computer controlled intervention that that could occur as part of the functioning of the system, the BCs, or the excitation function. For the design of systems in abnormal or hostile environments, the primary goal is typically *not* to analyze the performance of the system, but rather to seek system designs that will behave in predictable ways.

14.2.1 Steps in the PIRT process for modeling and simulation

The five steps for a simplified PIRT process are:

- assembly of the team,
- definition of the objectives of the PIRT process,
- specification of environments and scenarios,
- identification of plausible physical phenomena,
- construction of the PIRT.

The following description combines the work and recommendations of Wilson and Boyack (1998), the Sandia researchers referenced above, and the present authors.

14.2.1.1 Assembly of the team

PIRT development is best accomplished using a team that has a broad base of knowledge and expertise in a number of areas. The most important areas for individual knowledge and representation are (a) goals of the M&S effort, (b) needs of the customer, (c) operation of the system of interest or closely related systems, (d) environments and scenarios of interest, (e) different physical processes and phenomena occurring in the system as well as how they are modeled, and (f) analysts with experience in simulating the system of interest or closely related systems. Additional expertise should be included, depending on the nature of the system of interest and the modeling effort. The number of people on the team should be roughly five to ten. Teams any larger than this become inefficient, burdensome to keep on track, and meetings become difficult to arrange so that all team members can attend every meeting. The team members should have a demonstrated capability to work in a team environment, particularly the ability to control their individual agendas in the interest of the team effort. The assistance of technical and administrative support staff significantly improves the efficiency and productivity of the team. Finally, and most importantly, the team should have a clearly defined leader who has the recognized authority and is ultimately responsible for the effort and its documentation. This person should also have participated in previous PIRT efforts and be familiar with all aspects of the PIRT process.

14.2.1.2 Definition of the objectives of the PIRT process

The objectives of the PIRT process should be narrowly focused, as opposed to the objectives of the development of a M&S capability. Typically, the objectives of the PIRT process are oriented toward the performance, safety, or reliability of a system or subsystem. If the objectives are too broad, then the results of the process tend to be of limited value for prioritizing future efforts. For example, if the scenarios of interest are too broad or ill defined, then the prioritization result commonly is: everything needs work. Prioritization is fundamentally about ordering needs: some things will get done and some will not. As part of the definition of objectives, the team should specify what predictions are needed from the simulation capability. These can almost always be categorized as one or more SRQs. If multiple physics codes are coupled together to produce the SRQ of ultimate interest, there may be several SRQs from intermediate codes that are needed to predict the ultimate SRQ. Also, the SRQs of interest can be quantitative or qualitative in nature. For example, suppose one were interested in an accident scenario concerned with the separation or fracturing of a fan blade attached to the rotor disk on a gas turbine engine. One may be only interested in predicting if any fragments of the fan penetrate the engine cowling after the blade separates from the rotor; not in the detailed prediction of the deformation of the cowling.

It should be stressed that in the PIRT process there should be *no connection* to existing computer codes and needed simulation capabilities. The PIRT process is focused on discovering what are the most important physical phenomena that affect a system and its response, *not* on whether an existing computer code can simulate the phenomena or how well it can be simulated. These latter issues derail the discussion into expected performance or promotion of existing codes.

During formulation of the objectives of the PIRT process, the team must estimate the resources required to complete the identified objectives, compared with the resources available for the effort. The team should be fairly cautious and restrictive in its objectives, because of the time required to conduct the effort, as well as document the results.

14.2.1.3 Specification of environments and scenarios

Typically only one environment is specified for analyzing in the PIRT process; normal, abnormal, or hostile. For the environment chosen, one or more scenarios are usually specified, but they should be closely related. The scenarios specified should match the technical expertise and background of the team. If the scenarios are unrelated, then separate PIRT process teams should be formed. In the specification of the environment and scenarios, key parameters characterizing the system and surroundings should be specified, and, if possible, a range of parameter values should be specified. For example, these could be (a) the quantity and characteristics of coolant available in an emergency cooling system, (b) type and range of damage to a system or subsystem, (c) quantity of fuel available in a fuel fire, and (d) level of access to a computer control system by an attacker. In certain situations, the team may be given freedom to identify scenarios that have never been considered, but may also be pursued. The possibility of this approach should be specified in setting the objectives in Step 1 because, as will be seen in Section 14.2.1.5, a PIRT will be constructed for each scenario.

Figure 14.2 depicts the given environment chosen, the scenarios of interest, and the SRQs of interest for each scenario. As can be seen from this tree structure, the total number of quantities of interest that will be analyzed in the PIRT process can become quite large, if the scope of the effort is not limited. It should also be noted that, at the beginning of a PIRT process, not all of the scenarios of interest and the SRQs are necessarily recognized. For example, as the PIRT process evolves, new failure modes or coupled physics interactions are commonly discovered, that were not recognized initially.

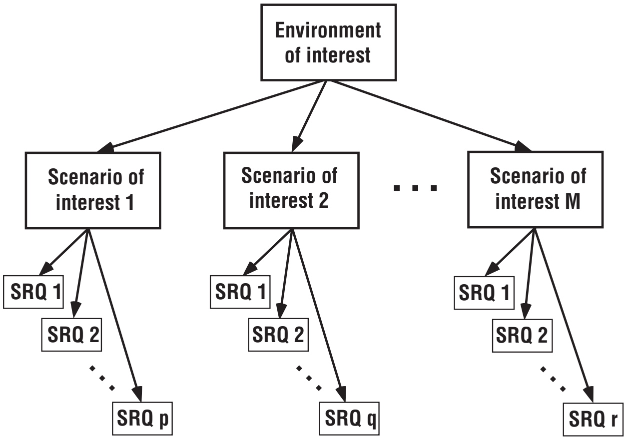


Figure 14.2 Environment–scenario–SRQ tree.

14.2.1.4 Identification of plausible physical phenomena

All plausible physical phenomena and interactions of phenomena should be considered for each scenario of interest, given that the phenomena could have an impact on the SRQs of interest. In many scenarios, it has proven beneficial to divide the scenario into time phases, since different physical phenomena tend to be important in different phases. For example, in loss of coolant accidents in nuclear reactors, the time phases could be divided into (a) initial rapid loss of coolant, (b) high heating of subsystems before emergency coolant is introduced, (c) initial introduction of emergency coolant onto high temperature subsystems, and (d) flooding of the system with emergency coolant. In some analyses, it is also useful to spatially divide the system or subsystem into regions where plausible phenomena can be better identified. For example, in a terrorist attack of a facility, it is useful to consider varying levels of physical access and control of a facility, as well as levels of armament and technical capability of the attackers.

Trying to discover all of the plausible phenomena that may occur in the system is a difficult task. The team can never be sure that they have discovered all of them; which is referred to as the problem of incompleteness. Creative, *thinking out of the box* type individuals on the team can significantly aid in this step. Different approaches can be used to generate ideas about plausible phenomena. Depending on the makeup of the team, one approach that has proven beneficial is to use some type of brainstorming. In this approach, many different, apparently unrelated, ideas are suggested. Sometimes these ideas are fruitful in themselves and sometimes they lead to other creative ideas that may be fruitful. Regardless of whether a team-brainstorming approach is used, or a more separate individual approach, the key is that no ideas are criticized or ridiculed. The key element in phenomena identification is *discovery*, *not* evaluation, appraisal, or ranking of importance of phenomena or ideas. The team leader plays an important role here to be certain that a proper and constructive atmosphere is maintained during the discussions of the team. Simplified simulations of the scenarios, or order of magnitude analyses, can be used to

Table 14.1 *Example of initial PIRT for a given environment and scenario.*

SRQ / Phenomena	SRQ 1	SRQ 2	•••	SRQ *p*
Physical phenomenon 1				
Physical phenomenon 2				
Physical phenomenon 3				
Physical phenomenon 4				
⋮				
Physical phenomenon *n*				

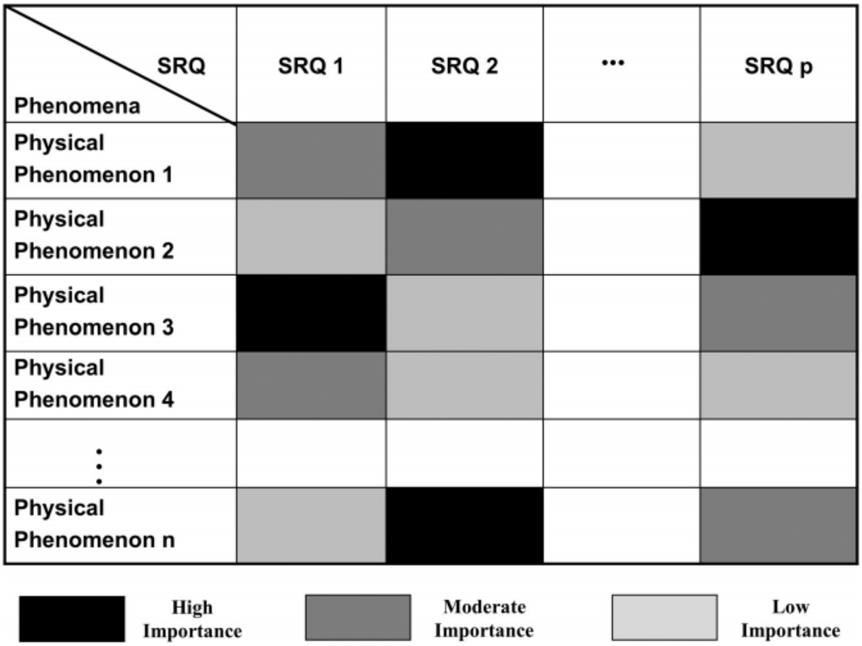
aid in the discussions. Use of more complex simulations is not recommended because they begin to focus the discussion on details of the analysis and existing modeling capabilities, instead of what is plausible. Most importantly, the M&S capability that will be evaluated later in the gap analysis, Section 11.3, should *never* be used during the PIRT process.

14.2.1.5 Construction of the PIRT

A PIRT is constructed for each scenario of interest by listing all of the plausible phenomena identified in the previous step (see Table 14.1). The phenomena can be listed in arbitrary order in the table, but it is recommended that they be grouped into general classes of phenomena or processes. For example in a heat transfer problem, they would be listed as conduction, convection, and radiation. All of the single-physics processes should be listed together. Physical processes that produce important effects because of strong coupling between single-physics processes should be listed as separate entries. If there are multiple SRQs for a scenario, then they can be listed across the top of the table. If a scenario were divided into multiple time phases or spatial regions, then a separate PIRT would be given for each phase or region.

After the initial PIRT is constructed, then the heart of the PIRT process begins: the discussion and debate to rank order the list of physical phenomena in terms of their importance on a particular SRQ of interest. Various methods of ranking the phenomena can be used. The simplest, and usually the most effective, is to define three relative ranks of importance: high, moderate, and low. A five-level scale can also be used: very high, high,

Table 14.2 *Example of a completed PIRT for a given environment and scenario.*



moderate, low, and very low. No more than a five-level scale should be used because a higher level of precision is not necessary for a qualitative effort such as this. In addition, the more levels of importance the more difficult it is to obtain agreement by a group of individuals.

Depending on the composition of the team and the resources set aside for the effort, the discussion and debate on ranking can be quite lengthy and energetic. The team leader should show a great deal of discretion and patience in the debates. Unless the team leader has a very strong opinion on a certain ranking, he should avoid taking sides in a debate concerning a particular ranking. Taking sides in a debate can lead team members to concluding he is aligning with certain factions within the team. The team leader should serve more in the role of an unbiased, rational observer and referee in the debate. After an appropriate amount of time debating the issues, the leader should call for a vote on the issue at hand. Any team member should be allowed *not* to vote, if they so choose.

After agreement has been reached on what level each phenomenon should be placed, the completed PIRT table would appear similar to that shown in Table 14.2. This table shows three levels: high is black, moderate is gray, and low is light-gray. Note that if a particular physical phenomenon is a combination of two other single-physics phenomena, it could be ranked at a higher level than the each of the single-physics phenomena. This could occur because of a strong nonlinear interaction of each of the contributing physical phenomena. One of the most common examples in fluid dynamics is the combination of chemical diffusion reaction and fluid dynamic turbulence. When multiple SRQs are listed in a PIRT table, it should be stressed that the ranking must be consistent across all of the SRQs. Stated differently, the level of ranking is a function of the individual phenomena, *not* the importance of one SRQ relative to another. If ranking of the importance of the individual SRQs is needed, based on the objectives of the PIRT process, this ranking should be done separately.

14.3 Gap analysis process

Because the ASC program was oriented toward developing new mathematical models and codes directed toward project needs, the Sandia researchers significantly extended the concept of a *gap analysis* (Pilch *et al.*, 2001; Tieszen *et al.*, 2002; Trucano *et al.*, 2002; Boughton *et al.*, 2003). In the gap analysis, the emphasis shifts from developing knowledge and prioritizing physical phenomena to an in-depth knowledge of the capabilities *available* in existing software. The key question that is answered in the gap analysis is: *Where does the existing capability presently stand, relative to the phenomena and SRQs that have been identified as important?* To answer this question, different expertise is needed on the team compared to the expertise needed in the construction of the PIRT. For the gap analysis, individuals are needed with knowledge of (a) physics models available in the code, (b) geometry and mesh generation options available in the code, (c) code and solution verification status of the code, (d) validation of the models for the system of interest or related systems, and (e) UQ of predictions for the system of interest or related systems. This is a wide range of expertise needed, but most of these areas could be filled by developers of the code in question, as well as analysts experienced with the system, or similar systems, of interest. New team members with the needed expertise should be added to the existing PIRT team, since the expertise now needed would probably not be represented on the original PIRT team. Some of the existing PIRT team members may choose to drop from the team for the gap analysis portion.

By having code developers or any other proponents of the existing code on the team, a formidable dilemma is posed. The code developers, or their proponents, have a very real vested interest in espousing the capabilities of the code, instead of critically assessing its capabilities. There is no single method of dealing with this situation, because it depends on the individuals and organization involved. The managers responsible for the formation of the gap analysis team, as well as the team leader, must be cognizant of the dilemma posed to the individuals involved. If the code developers or proponents are unrealistic in evaluating the existing capability, or are always on the defensive of criticisms of the code, it can greatly hinder or mislead the gap analysis process. If the code developers or proponents are critical of the existing capability, then they risk severe criticism from their peers or their management. If the existing code being evaluated is from an external organization, for example, commercial software, different issues come into play. These will be briefly discussed in a later section.

Table 14.3 *Example of an initial gap analysis table for a given environment, scenario, and SRQ of interest.*

Phenomena \ Gap Areas	Physics Modeling	Code and Solution Verification	Model Validation	Uncertainty Quantification
Physical Phenomenon 1				
Physical Phenomenon 2				
Physical Phenomenon 3				
Physical Phenomenon 1				
Physical Phenomenon 2				
Physical Phenomenon 3				
Physical Phenomenon 4				

14.3.1 Construct the gap analysis table

The gap analysis begins with the information generated in the completed PIRT, Table 14.2. For each SRQ of interest, the most important one or two levels of physical phenomena are singled out for a gap analysis. The most common groups of activities for assessment of gaps are physics modeling, code verification, model validation, and uncertainty quantification. Table 14.3 shows the initial table for a gap analysis for these groups for the top two levels of physical phenomena identified in the PIRT process. In this table it is assumed that there are three phenomena in the top level (indicated by the dark stripe), and four in the second level (indicated by the gray stripe). The order of listing of the phenomena within a level is usually not important at this point.

The purpose of the assessment is to determine if each of the four activities listed across the top of the gap analysis table has been adequately addressed for the physical phenomena so that the SRQ of interest can be adequately predicted. The meaning of *adequacy* depends on the specific activity. In the following, the general question that should be addressed concerning adequacy is listed for each activity, as well as detailed examples of questions for each activity.

1 Physics modeling: does the code have the capability to adequately address the phenomena in question? Examples of detailed questions in this activity are
   - Does it have the ability to model the phenomena, and any physics coupling, over some or all of the physical parameter space of interest?

- Are all of the code options needed to compute the SRQ of interest operational?
- Does the code have alternative models that may be used to deal with the phenomena of interest?
- Are the models physics-based; or are they empirical, data-fit models?
- Does the model have the capability to address the needed spatial and temporal scales in the phenomena?
- Does the model have the flexibility to deal with the geometries of interest, as well as geometry details that may be needed?
- Does the code have all of the material models needed and are they operational over the needed range of parameters?

2 Code and solution verification: has the code undergone adequate code verification testing for the code options that would be used, and can it adequately estimate numerical solution error in the SRQs of interest? Examples of detailed questions in this activity are
   - Is there a regression test suite that adequately tests the options that would be used and is it run routinely?
   - Have adequate test problems, such as manufactured solutions, been successfully computed that test the software options and algorithms that would be used?
   - Are there any outstanding code bugs that have been reported for the code options that would be used?
   - Are adequate methods available in the code to estimate or control iterative solution error?
   - Are adequate methods available in the code to estimate spatial discretization error in the SRQs of interest?
   - Are methods available in the code for generating global mesh refinement with adequate user control e.g., refinement of the mesh uniformly?
   - Does the code have the capability for adaptive re-meshing of the domain of interest and, if needed, re-meshing as a function of time?

3 Model validation: is the model validation domain adequate for the phenomena in question, and have adequate quantitative accuracy assessments of the model been made for the SRQ of interest? Examples of detailed questions in this activity are
   - At what tiers in the validation hierarchy has the model been compared to experimental measurements for the phenomena in question?
   - Has the model been compared to experimental measurements over the range of relevant parameters for the phenomena of interest?
   - Was adequate spatial and temporal discretization attained for the comparisons that were made with experimental data to be certain that the physics of the model was evaluated, as opposed to a mixture of physics and numerical error?
   - For the comparisons that have been made, is there adequate quantitative agreement between simulation and experiment for the SRQs for the phenomena of interest?
   - Have the comparisons that were made adequately related to the phenomena of interest in terms of system geometry and material properties?

4 Uncertainty quantification: is the accuracy of the predicted SRQ of interest, including its estimated uncertainty, adequate for the phenomena of interest? Examples of detailed questions in this activity are
   - Are small or large extrapolations of the model required from the validation domain to the conditions and phenomena of interest?

- Are extrapolations required in terms of the relevant parameters for the phenomena of interest, or are the extrapolations required in terms of higher tiers in the validation hierarchy?
- Are the extrapolations based on calibration of the model over the validation domain, or are they based on extrapolation of validation metrics computed over the validation domain?
- Will adequate spatial and temporal discretization be possible for the conditions and phenomena of interest, given the computer resources available?
- Are adequate UQ computational tools and expertise available for the task at hand?
- Are computer resources available to compute an adequate number of simulations to estimate the uncertainty in the SRQ of interest for the phenomena of interest?

Answering these questions requires a great deal of knowledge about the models and the code, forthright assessment of past performance of the models and codes, and a reasonable idea of what is required for the physical phenomena and system of interest. The discussion of these questions among the team should be frank and constructive. The questions should be answered given their status at the time of the gap analysis effort, not some anticipated or hoped for capability or completion date. Many of the issues related to physics modeling, code and solution verification, and model validation are more factual than speculative. For example, they are more of the variety *Does the code have this option?* or *Have we made this comparison with data?* Most of the issues related to uncertainty quantification are more speculative. For example, they are commonly of the type *Do we believe the models or code can do this?* or *How large do we think the uncertainty will be?* On the more speculative questions, expect a wide range of views and substantial debate.

After some level of consensus among the team has been achieved, the gap analysis table can be completed. Table 14.4 shows an example of a completed gap analysis table for a given environment, scenario, and SRQ of interest. This table only shows three levels of assessment: adequate, inadequate, and unknown. When the gap analysis approach was being developed only two levels were available, adequate and inadequate. It was found, however, that there were a number of situations where there was unavailable or conflicting information concerning the adequacy or inadequacy of an activity. As a result, the unknown level was created to keep the gap analysis process moving instead of prolonged debate on the adequacy. As will be discussed in Section 14.3.3, updates and new information can be added to a completed gap analysis table.

Two observations should be pointed out in Table 14.4. First, the most important activities that need to be pursued to close perceived gaps in needed capabilities are those that occur in the most important phenomena. These are denoted by the dark stripe on the left. Which ones to pursue is more than a technical issue, it is also a resource issue and an organizational responsibility issue. Although it is beyond the scope of this discussion to delve into these topics, one example is mentioned. One can have the situation where an important physical phenomenon is assessed *inadequate*, but the organization responsible for the activity either disagrees with this assessment or is unresponsive to the needs of the project sponsoring the gap analysis. Second, note that there is a clear directional nature to the table. Once an unknown or inadequate level occurs for a phenomenon, it only gets worse as one

Table 14.4 *Example of a completed gap analysis table for a given environment, scenario, and SRQ of interest.*

Phenomena \ Gap Areas	Physics Modeling	Code and Solution Verification	Model Validation	Uncertainty Quantification
Physical Phenomenon 1	Adequate	?	Inadequate	Inadequate
Physical Phenomenon 2	Adequate	Adequate	Adequate	?
Physical Phenomenon 3	Adequate	Adequate	Inadequate	Inadequate
Physical Phenomenon 1	Adequate	Adequate	Adequate	Adequate
Physical Phenomenon 2	Adequate	Adequate	?	Inadequate
Physical Phenomenon 3	Inadequate	Inadequate	Inadequate	Inadequate
Physical Phenomenon 4	?	Inadequate	Inadequate	Inadequate

Adequate   Inadequate   ? Unknown

proceeds to the right in the table. For example, if code and solution verification or model validation is unknown or inadequate, uncertainty quantification has little chance of being adequate. It should also be noted that in model validation, if an unknown or inadequate is registered, it may *not* be a modeling deficiency with the code. It may simply be that the experimental data is not available for comparison in order to make the accuracy assessment needed.

14.3.2 Documenting the PIRT and gap analysis processes

Although it is seldom a welcomed task, documentation of the work of the team is an extremely important aspect of the project. The PIRT and gap analysis tables produced capture only the summary information from the processes. A report should be completed that also captures the objectives of the processes, specification of the environments, discovery and description of the scenarios and the SRQs of interest, identification of the plausible phenomena, ranking of the phenomena, and justification of the gap analysis table. The documentation should include the reasoning and arguments made for important decisions, as well as why certain aspects or issues were excluded from the processes.

As discussed earlier in this chapter, the purpose of the PIRT and gap analysis processes is to recommend how best to allocate resources between various activities to achieve success of a project-oriented effort. The PIRT and gap analysis processes should be viewed as a generation of information, which management should use for optimizing the allocation of resources. If the processes are not documented at some appropriate level of formality and detail, then experience has shown that the value and impact of the effort will be minimal. Not only will the resources expended during the processes be wasted, but, more importantly, there could be a major waste of resources in the M&S effort. This waste of information could possibly lead to the failure of the effort, as well as failure of the engineering system to which it is contributing.

14.3.3 Updating the PIRT and gap analysis

The results of the combined PIRT and gap analysis processes should not be viewed as cast in stone once the documentation is complete. It must be fully recognized that the information generated falls into the category of expert opinion. The quality of the expert opinion depends on the quality of the team members and team leadership, how the processes were conducted, and the level of effort expended. With reasonable effort, the information generated is always valuable, but there can be errors and misjudgments in certain aspects of the expert opinion. When new information becomes available, the gap analysis table can be updated, for example: (a) activities marked as unknown can be changed to adequate or inadequate, (b) capabilities are added to a code, or verification testing is completed, (c) experimental data is obtained and model validation activities are conducted, and (d) UQ activities are conducted on a similar system. If appropriate for a large-scale effort, a Gantt chart can be constructed and updated as new information becomes available, particularly when identified gaps in capability are eliminated.

It is to be expected that some surprises will be found in the follow-on efforts related to each activity. Examples of some surprises, most of them bad, which have been observed in practice, are the following.

- A combination of code options that was needed for the physical phenomena of interest was known to be available, but it was discovered that the code would not run the needed combination.
- It was discovered that when a manufactured solution was constructed to test the combination of options needed, the code was found to be 0th order accurate, i.e., it converged to the wrong answer.
- It was discovered that the numerical solution error estimator would not run on the physical phenomenon, i.e., it showed wild oscillations for the problems of interest.
- After an experiment or simulation was conducted, it was found that a physical phenomenon that had been ranked as high importance was changed to moderate importance.
- A model comparison with an experiment that had shown good agreement was recomputed and it was found that when the mesh resolution was improved, the agreement became unacceptable.
- When a new validation experiment was conducted, it was found that a major recalibration of the model parameters had to be conducted to obtain good agreement with the data.
- When an alternative plausible model for a physical phenomenon was used to compute the SRQ of interest, it was found that there was a large disagreement compared to the traditional model used.

14.4 Planning and prioritization with commercial codes

Our discussion on P&P has concentrated on large-scale M&S projects that are conducted primarily within an organization. Here we make several comments concerning how P&P would be conducted differently by an organization that is using, or is considering using, commercial software. The context of P&P when using commercial software must be understood as a business relationship between the organization doing the planning and the software company, *not* a technical relationship. Technical issues are certainly important, especially to staff members, but the dominant issue is: *can the organization build a strategic business relationship with a software company as a stakeholder?* If the organization only has relatively few software licenses from the software company, it is doubtful that a substantive relationship can be built. The software company may proclaim how important a stakeholder relationship is to them, but if few licenses are involved, there will be little genuineness in the claim. Only when the software company becomes a true stakeholder can the kind of P&P discussed here be accomplished.

An important aspect that management of the organization needs to consider before a stakeholder relationship is considered is the question of confidence that the software company can and will protect their proprietary information. The proprietary information would not only deal with existing products the organization sells, but also research information, proprietary experimental data, proposal information, and designs for new products. The mind set toward information within a software company is typically very different from a company that designs and sells hardware products. If the organization is dealing with a large software company, then another issue comes into play. The software company may also be working closely with a competitor of the organization. For example, the same software company may have information on, or be working on, two competing proposals from their clients.

To conduct the gap analysis discussed here, the software company would need to be an intimate part of the discussions and probably represented on the gap analysis team. The software company individuals on the team would need to be very candid and constructively critical of their own company's product. When explicitly discussing shortcomings in their software, they would be concerned about potentially losing the organization's business, as well as detrimental information being passed on to other potential new customers for their software. It is clear this is a delicate business as well as technical issue. On the one hand, the organization needs to be certain it will obtain forthright information from the software company. On the other hand, the software company needs to be certain it will not lose its client, nor have its skeletons in the closet exposed for public consumption.

As a final topic, consider the issue of ownership of software capabilities. Suppose that after a gap analysis has been completed, it is found that a new capability is needed in the commercial software. The software company may decide it would be in its best interest to fund the development and testing of the new capability. It may decide, however, that it has other priorities for new capability development. Suppose the new capability is of high importance to the company conducting the planning so that it will fund the development and testing of the new capability as an add-on, standalone feature, to the code. If so, then the planning issues will also include the level of support and information needed from the software company. It may even involve proprietary information from the software company, requiring negotiated contracts protecting the intellectual property of the software company.

Suppose, however, the organization did not have the expertise to develop and test the new capability, or the capability must be added directly in the source code. The organization may fund the software company to develop, test, and document the new capability. Then the question must be addressed: who will own the new capability? The organization would primarily view the issue as: what is the return on investment for funding the new capability? It may also want increase its return on investment by not allowing any competitor to use the capability. The software company would view the issue as: how can we increase our software license sales by advertising this new capability to existing or potential new customers? Various solutions to these differing perspectives would need to be carefully negotiated between the organization and the software company. One idea would be to allow the organization to have exclusive rights to use the new capability for a set period of time in the future. In addition, the software company could not advertise or discuss the capability until after that time period.

14.5 Example problem: aircraft fire spread during crash landing

The example discussed here is concerned with fire spreading on board a commercial transport aircraft during a crash landing. The abnormal environment is defined to be a survivable crash and the fuel fire initiates immediately after initial contact of the aircraft with the ground. The goal of the PIRT analysis is to identify and rank the importance of the physical phenomena with regard to the survivability of the passengers to the fire environment. The goal of the gap analysis is to identify the important gaps in an existing capability for predicting certain SRQs of interest. In our discussion, we will identify various scenarios and SRQs, but only one branch of the scenario-SRQ tree will be pursued.

The environment is specified as:

- the aircraft has a single aisle and carries 150 passengers and crew;
- the crash does minor damage to the aircraft cabin structure;
- the fuel carried on board is JET A-1, and the quantity can range from 1000 L to 25 000 L;
- the fire initiates in the undercarriage of the aircraft as it first impacts the ground.

The scenarios are specified as:

- Scenario 1: 1000 L of fuel are on board, the landing gear is extended and has minor damage during landing rollout on a runway, and fully equipped aircraft rescue and firefighting (ARFF) personnel arrive at the aircraft in 2 min.
- Scenario 2: 25 000 L of fuel are on board, the landing gear is extended, but it is destroyed during rollout on a runway, and fully equipped ARFF personnel arrive at the crash in 5 min.

Table 14.5 *Completed PIRT for the aircraft fire environment and scenario 3.*

Physical Phenomena \ SRQ	SRQ 1	SRQ 2	SRQ 3	SRQ 4	SRQ 5
Convective: Buoyant turbulent mixing	Moderate	Moderate	Moderate	Moderate	High
Convective: Combusting turbulent flow	High	High	High	High	High
Convective: Wind effects	Moderate	Moderate	Moderate	High	High
Mass transport: Fuel evaporation	Low	Low	Low	Low	Low
Chemistry: Fuel combustion	Moderate	Moderate	Moderate	High	Low
Chemistry: Soot production	Low	High	Low	Low	Low
Chemistry: Cabin material combustion	High	High	High	Low	Low
Chemistry: Carbon monoxide production	Low	Moderate	High	Low	Low
Radiation: Emissive flux soot formation	Moderate	High	Moderate	Low	Low
Radiation: Emissive flux combustion chemistry	Low	Moderate	Low	Moderate	Moderate
Radiation: Mesoscale turbulent mixing	Low	High	Low	Moderate	Moderate
Radiation: Material properties in cabin	High	High	High	Low	Low
Radiation: Material properties of emergency slide	Low	Low	Low	High	Moderate
Conduction: Through aluminum structure	Low	Low	Low	Low	Low
Structure: Aluminum melting	Low	Low	Low	Low	Low
Structure: Emergency slide melting	Low	Low	Low	High	Moderate

High Importance / Moderate Importance / Low Importance

- Scenario 3: 1000 L of fuel are on board, the landing gear is not deployed before ground impact, there is considerable damage to the underbelly of the aircraft during impact and deceleration on an open field, and no ARFF personnel are available.

A team of experts with the needed expertise was formed to conduct the PIRT and gap analysis. The key modeling expertise needed for the PIRT are combustion modeling of a wide range of materials, toxicology, fluid dynamics, heat transfer, and solid dynamics. The team was not given the SRQs of interest as part of the goals of the analysis, so these had to be defined. A number of possible SRQs were considered by the team to try to determine what were appropriate quantities for gauging the survivability and escape of the passengers and crew from the burning aircraft. The following quantities were chosen for all three scenarios:

- SRQ 1: gas temperature at mid-height of the cabin, along the center of the aisle, as a function of time;
- SRQ 2: soot temperature at mid-height of the cabin, along the center of the aisle, as a function of time;
- SRQ 3: carbon monoxide concentration at mid-height of the cabin, along the center of the aisle, as a function of time;

Table 14.6 *Completed gap analysis table for the aircraft fire environment, scenario 3, and SRQ 1.*

Phenomena \ Gap Areas	Physics Modeling	Code and Solution Verification	Model Validation	Uncertainty Quantification
Convective: Combusting turbulent flow	Adequate	Adequate	?	Inadequate
Chemistry: Cabin material combustion	Inadequate	Inadequate	Inadequate	Inadequate
Radiation: Material properties in cabin	Inadequate	Inadequate	Inadequate	Inadequate
Convective: Buoyant turbulent mixing	Adequate	?	?	Inadequate
Convective: Wind effects	Adequate	Adequate	Adequate	Adequate
Chemistry: Fuel combustion	Adequate	Adequate	?	Inadequate
Radiation: Emissive Flux soot formation	Adequate	Inadequate	Inadequate	Inadequate

Adequate Inadequate ? Unknown

- SRQ 4: time of collapse of the emergency exit slide due to melting;
- SRQ 5: gas temperature along a vertical line from the top of each cabin exit to the ground, as a function of time.

Plausible physical phenomena were considered that could affect each of the scenarios and the prediction of the SRQs of interest. The key difference between scenarios 1 and 2 and scenario 3 is the effect of the ARFF personnel in containing and controlling the fire. In addition to the fire-spread issues, scenarios 1 and 2 would also need to deal directly with physical phenomena related to various foams, fire fighting, and life-saving tactics. In scenario 3, these phenomena would not come into play. Table 14.5 lists all the physical phenomena considered by the team for scenario 3. The table also shows the results of the team's analysis yielding three levels of importance ranking of phenomena for each SRQ.

The team was then asked to conduct a gap analysis for the physical phenomena and the SRQs identified in Table 14.5. The code being assessed was an existing in-house code. Since some of the PIRT analysis team members were unfamiliar with the code, new team members were added that were part of the team that developed the code, as well as those that provide maintenance and support for the code.

The four areas of assessment of the existing code were physics modeling, code and solution verification, model validation, and uncertainty quantification. A gap analysis table was then constructed for each of the SRQs listed in Table 14.5, considering only the high and moderate importance levels. Table 14.6 shows the completed gap analysis table for SRQ 1.

As can be seen from the gap analysis, the existing code did not fare very well. In two of the three high importance phenomena it is missing modeling options for combustion chemistry of cabin material and properties for materials commonly occurring in cabin interiors. In addition, its existing validation domain had very little overlap with this application domain and/or its level of agreement with existing validation data was inadequate for the present application. This type of gap analysis result commonly occurs when existing codes are assessed for new applications. If one were to rely on the broad claims and advertising of code capabilities, a project or a proposal effort could be severely misled. Even though this experience is widely recognized, it is still very difficult to convince project managers, especially for proposal efforts, to set aside time and resources to make informed decisions concerning needed capabilities and existing deficiencies.

14.6 References

Boughton, B., V. J. Romero, S. R. Tieszen, and K. B. Sobolik (2003). *Integrated Modeling and Simulation Validation Plan for W80–3 Abnormal Thermal Environment Qualification – Version 1.0 (OUO)*. SAND2003–4152 (OUO), Albuquerque, NM, Sandia National Laboratories.

Boyack, B. E., I. Catton, R. B. Duffey, P. Griffith, K. R. Katsma, G. S. Lellouche, S. Levy, U. S. Rohatgi, G. E. Wilson, W. Wulff, and N. Zuber (1990). Quantifying reactor safety margins, Part 1: An overview of the code scaling, applicability, and uncertainty evaluation methodology. *Nuclear Engineering and Design*. **119**, 1–15.

Charette, R. N. (2005). Why software fails. *IEEE Spectrum*. September.

Dorner, D. (1989). *The Logic of Failure, Recognizing and Avoiding Error in Complex Situations*, Cambridge, MA, Perseus Books.

Flowers, S. (1996). *Software Failures: Management Failure: Amazing Stories and Cautionary Tales*, New York, John Wiley.

Galin, D. (2003). *Software Quality Assurance: From Theory to Implementation*, Upper Saddle River, NJ, Addison Wesley.

Hanson, R. G., G. E. Wilson, M. G. Ortiz, and D. P. Grigges (1992). Development of a phenomena identification and ranking table (PIRT) for a postulated double-ended guillotine break in a production reactor. *Nuclear Engineering and Design*. **136**, 335–346.

Karolak, D. W. (1996). *Software Engineering Risk Management*, Los Alamitos, CA, IEEE Computer Society Press.

Kroeger, P. G., U. S. Rohatgi, J. H. Jo, and G. C. Slovik (1998). *Preliminary Phenomena Identification and Ranking Tables for Simplified Boiling Water Reactor Loss-of-Coolant Accident Scenarios*. Upton, NY, Brookhaven National Laboratory.

NRC (1988). *Acceptance Criteria for Emergency Core Cooling Systems for Light Water Reactors*. U. S. N. R. Commission, U. S. Code of Federal Regulations. 10 CFR 50.

Pilch, M., T. G. Trucano, J. L. Moya, G. K. Froehlich, A. L. Hodges, and D. E. Peercy (2001). *Guidelines for Sandia ASCI Verification and Validation Plans – Content and*

Format: Version 2. SAND2000–3101, Albuquerque, NM, Sandia National Laboratories.
Rohatgi, U. S., H. S. Cheng, H. J. Khan, and W. Wulff (1997). *Preliminary Phenomena Identification and Ranking Tables (PIRT) for SBWR Start-Up Stability*. NUREG/CR-6474, Upton, NY, Brookhaven National Laboratory.
Saaty, T. L. (2001). *Decision Making for Leaders – the Analytic Hierarchy Process for Decisions in a Complex World*. 3rd edn., Pittsburgh, PA, RWS Publications.
Shaw, R. A., T. K. Larson, and R. K. Dimenna (1988). *Development of a Phenomena Identification and Ranking Table (PIRT) for Thermal-Hydraulic Phenomena during a PWR LBLOCA*. Idaho Falls, ID, EG&G.
Stepanek, G. (2005). *Software Project Secrets: Why Software Projects Fail*, Berkeley, CA, Apress.
Tieszen, S. R., T. Y. Chu, D. Dobranich, V. J. Romero, T. G. Trucano, J. T. Nakos, W. C. Moffatt, T. F. Hendrickson, K. B. Sobolik, S. N. Kempka, and M. Pilch (2002). *Integrated Modeling and Simulation Validation Plan for W76–1 Abnormal Thermal Environment Qualification – Version 1.0 (OUO)*. SAND2002–1740 (OUO), Albuquerque, Sandia National Laboratories.
Trucano, T. G., M. Pilch, and W. L. Oberkampf (2002). *General Concepts for Experimental Validation of ASCI Code Applications*. SAND2002–0341, Albuquerque, NM, Sandia National Laboratories.
Vose, D. (2008). *Risk Analysis: a Quantitative Guide*. 3rd edn., New York, Wiley.
Wiegers, K. E. (2003). *Software Requirements*. 2nd edn., Redmond, Microsoft Press.
Wilson, G. E. and B. E. Boyack (1998). The role of the PIRT in experiments, code development and code applications associated with reactor safety assessment. *Nuclear Engineering and Design*. **186**, 23–37.
Wilson, G. E., B. E. Boyack, I. Catton, R. B. Duffey, P. Griffith, K. R. Katsma, G. S. Lellouche, S. Levy, U. S. Rohatgi, W. Wulff, and N. Zuber (1990). Quantifying reactor safety margins, Part 2: Characterization of important contributors to uncertainty. *Nuclear Engineering and Design*. **119**, 17–31.
Wulff, W., B. E. Boyack, I. Catton, R. B. Duffey, P. Griffith, K. R. Katsma, G. S. Lellouche, S. Levy, U. S. Rohatgi, G. E. Wilson, and N. Zuber (1990). Quantifying reactor safety margins, Part 3: Assessment and ranging of parameters. *Nuclear Engineering and Design*. **119**, 33–65.