

Verification, Validation, and Uncertainty Quantification in Thermal Hydraulics, Freeman Scholar Lecture (2019)

Upendra S. Rohatgi
 Mem. ASME
 Brookhaven National Laboratory,
 Building 490C,
 Upton, NY 11973

Engineering problems are generally solved by analytical models or computer codes. These models, in addition to conservation equations, also include many empirical relationships and approximate numerical methods. Each of these components contributes to the uncertainty in the prediction. A systematic approach to judge the applicability of the code to the intended application is needed. It starts from verification of implementation of formulation in the code, identification of important phenomena, finding relevant tests with quantified uncertainty for these phenomena, and validation of the code by comparing predictions with the relevant test data. The relevant tests must address phenomena as expected in the intended application. In case of small size or limited condition tests, the scaling analyses are needed to assess the relevancy of the tests. Finally, a statement of uncertainty in the prediction is needed. Systematic approaches are described to aggregate uncertainties from different components of the code for intended application. In this paper, verification, validation, and uncertainty quantifications (VVUQs) are briefly described. [DOI: 10.1115/1.4053718]

1 Introduction

Many problems in engineering are solved by analytical models or computer codes. These models are based on conservation equations, for mass, momentum, and energy, constitutive or empirical relationships, and numerical methods. These codes are applied in designing large facilities or for investigating their response to abnormal conditions. This is more acute for systems where multi-component fluid undergoes phase change either due to heating or depressurization. Additionally, full system level data may not be available. In these situations, code predictions provide system response. So, the question is how much to trust the code or models? How to determine fidelity of code predictions? More complicated is the physics, harder will be to answer these questions.

To have confidence in predictions, the code must undergo rigorous review. The first step is verification. The formulation is transformed into a computer code. Verification is a process of ensuring that formulation has been correctly programmed. The next step is validation where accuracy of the models is evaluated by comparing the predictions with data obtained from *relevant* tests. Usually there is lack of relevant data corresponding to applications. Finally, all analyses require an estimate of uncertainty in the prediction for an application.

The uncertainty in the prediction is sum of contributions from numerics, models, and parameters such as geometry, initial, and boundary conditions. In general, the statement of uncertainty is presented as a statistical statement due to uncertainty distribution of components contributing to the uncertainty.

In this paper, verification, measure of relevancy (scaling) of the tests, validation, and methods of uncertainty evaluations will be briefly discussed.

“A model without observation is a mathematical exercise of little use. Observations without models contribute mostly to confusion” J.-L. Lions 1928–2001.

All verification, validation, and uncertainty quantifications (VVUQs) are *application* dependent. Figures 1 and 2 summarize the concepts. The first diagram is traditional approach where performance of reality or application is converted into mathematical formulation. The formulation is coded into computerized model. The verification ensures that the formulation is correctly coded. The computerized model is applied to application relevant tests, and predictions are compared with the data. This way the code is validated for the intended application. However, in case of complicated physics such as two-phase flow, or impossibility to conduct tests due to power, pressure, or safety with the intended facility, an intermediate step is needed to design a facility that is

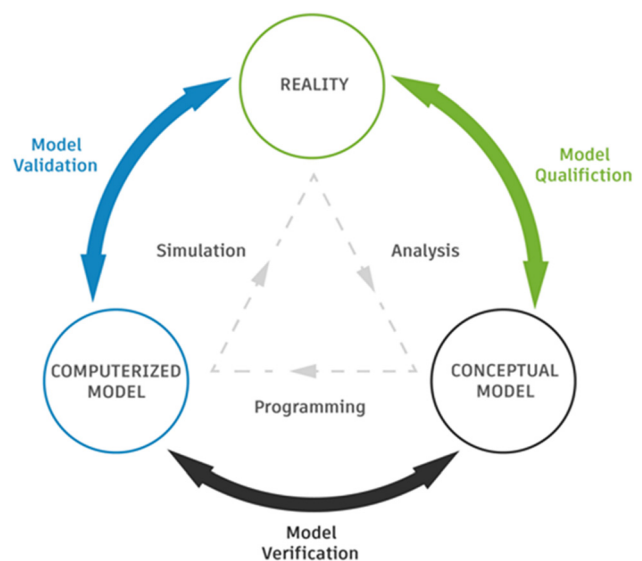


Fig. 1 Tradition approach of validation

Contributed by the Fluids Engineering Division of ASME for publication in the JOURNAL OF FLUIDS ENGINEERING. Manuscript received September 24, 2021; final manuscript received January 5, 2022; published online February 21, 2022. Editor: Francine Battaglia.

The United States Government retains, and by accepting the article for publication, the publisher acknowledges that the United States Government retains, a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for United States Government purposes.

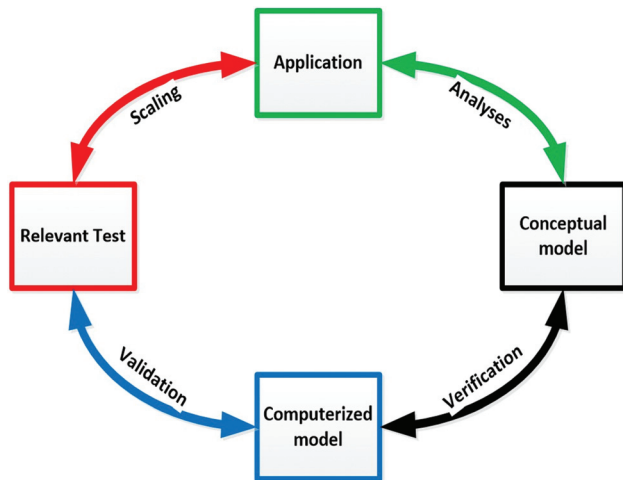


Fig. 2 General approach of validation

scaled down version of the intended application. The data from this scaled down facility are a surrogate of the actual application for validation of the computerized model. Final step is the estimation of the uncertainty of predictions from validated computerized model or code.

Verification, validation, and uncertainty quantification is important for many applications. The concepts of VVUQ have been extensively used in nuclear industry. A general perspective of VVUQ is provided by Roache [1].

U.S. Nuclear Regulatory Commission has licensed nuclear reactors based on modeling and simulation since 1960s [2]. There are many subscale tests but no prototype testing. Biggest contributors to uncertainty in the predictions are model uncertainties. Weapons area modeling and simulation has been a driver for VVUQ in scientific computing. Again, no prototype tests exist. Aircrafts have been designed with wind tunnel tests, and prototype performance is predicted based on validated codes.

ASME started *Journal of Verification, Validation, and Uncertainty Quantification* in 2016 to promote validation and verification advancements.

In an effort to provide uniformity with reporting errors, in 1985, the *ASME Journal of Fluids Engineering* started requiring a statement of numerical error, and full guideline was created in 1993 [3]. ASME Verification and Validation-20 committee has also published guidelines for computational fluid dynamics [4]. American Institute of Aeronautics and Astronautics has also published a guide on verification and validation in 2002 [5].

Some examples of comprehensive publications related to scientific computing are given in few references [6,7].

2 Verification

Verification is an important step in assuring that the code represents the formulation, and the code results are accurate, or accuracy can be quantified. This is accomplished by two steps: code verification and solution verification. These methods have been described with examples [6,8–10].

The code verification ensures that formulation, balance equations, constitutive relationships, and numerical scheme have been correctly programmed in the code. In one approach, during development, the coding is checked by another person, knowledgeable about equations and coding. If the application is for single-phase flow, results can be compared to actual analytical solution for simple geometry. The other methods to give confidence in code are changing boundary conditions and reviewing the trend, providing symmetrical boundary conditions, and checking for expected symmetry in solution, and finally code to code comparisons. All these

methods may not be sufficient but collectively they provide verification.

The additional method for complicated physics is the method of manufactured solutions that generates an analytical solution for code verification [8,9,11,12]. The basic idea of the procedure is to simply *manufacture* an exact solution for the equations with simplified physics for simple geometry. An approximate analytical solution is created and plugged in the equations. A source term is calculated that will make solution satisfy the formulation. This source term is incorporated in the code, and the code prediction should be same as the solution that was the basis of computing the source term.

The solution verification deals with the numerical accuracy of the computational model and provides estimate of numerical uncertainty. The uncertainty occurs due to inadequate spatial and temporal discretization and loose convergence criteria. *ASME Journal of Fluids Engineering* requires grid convergence studies before results are accepted. Grid convergence index provides a measure of numerical uncertainty [3,8,10,13,14].

In case of complex physics such as multicomponent, multiphase flows, nodalization (spatial discretization) studies for system codes are difficult. In these situations, the model form uncertainties and numerical uncertainty are combined for the purpose of estimating aggregate uncertainty in the prediction.

3 Phenomena Identification and Ranking Table

For computer codes to be credible, they should have physics-based models that have undergone rigorous validation—where code results are compared with the test data. Codes are applied to a variety of problems, and the submodels of physical phenomena have different levels of impact on the quantities of interest (QoIs). Therefore, requirements for models of phenomena and components will be specific to an application. This leads to a need to prioritize the model development and tests for each application. In addition, the uncertainty quantification methods must also appropriately choose characterization of important physical phenomena and their interactions. This is important as the code includes hundreds of models and is expensive to quantify uncertainty for each of them.

A method of identifying important phenomena is essential and was first developed under U.S. Nuclear Regulatory Commission program of code scaling, applicability, and uncertainty (CSAU) quantification [15,16]. This is called phenomena identification and ranking table (PIRT). The PIRT has two parts: first, identification of phenomena and components, and second, ranking them based on their impact on QoI or figure of merit (FOM). Another recent addition to traditional PIRT is inclusion of status of knowledge. Knowledge will include appropriate database and basic models. Combination of PIRT ranking and status of knowledge will set the priority of future experiment and model development. This has now been included in most of the recent PIRTs. This information will guide in model development and performing needed tests.

Past PIRTs were developed based on expert opinion. However, this becomes subjective. The other approaches are based on scaling or sensitivity analyses that quantify the impact of phenomena on figure of merit or QoI. These are called quantitative PIRT or QPIRT [17,18].

As PIRT is application dependent, the first step is to identify the application. In case of nuclear energy, the application will be related to nuclear reactor. The problem could be steady-state or transient. In case of transient, the duration of the transient is divided into periods based on important events such as valve opening or operator action, or some significant change in important phenomenon.

Once problem and period of transient have been established, the phenomena are decomposed from top-down approach. Reactor system first decomposed in larger component or sections with group of phenomena. Each of these sections is further decomposed. Finally, the decomposition leads to single basic

phenomena. This step assures that all the phenomena and their hierarchy have been accounted. This step depends on the expertise of the PIRT participants.

The next step in PIRT is to rank the phenomena based on their impact on figure of merit. The first approach is based on expert opinion. This was earlier developed in CSAU [15] and applied to pressurized water reactor large break loss of coolant event. This approach requires broad expertise in the participants of the PIRT. Subsequently, this approach has been adopted in different fields. This is the most common method of developing PIRT. The weakness of this approach is subjectiveness, dependency on expert opinion, and resulting consensus.

The second approach, consists of two methods, is based on relative quantitative impact of phenomena/component on figure of merit and is called QPIRT [17,18]. The first method requires an order of magnitude analyses based on conservation equations and reference values of the variables for each phase of the transient or for steady-state. The examples of this method are hierarchical two tier scaling (H2TS) [19] and fractional scaling analyses [20]. In this method, the initial or average values of reference quantities are used to nondimensionalize the system level balance equations for each phase of the transient. This method provides a quantitative approach of phenomena/component ranking.

The other method for QPIRT is based on sensitivity analyses using the computer codes. In this approach, the sensitivity of output parameter or figure of merit to parameters of model or initial and boundary conditions are estimated. Models with higher sensitivity have higher intensity of change or rate of change. However, the impact will be measured by net change of the parameter of interest or FOM over the variation of the model or input parameters during period of interest. The weakness of this method is that sensitivity and ranking will be based on the models in the code that will be the target of code validation.

4 Calibration

The term calibration has different meaning for instrumentation and for code models. In this paper, calibration will only refer to improvement of models. Process of improving the predictive capability with of part of data, by adjusting parameters of the computational model, is called calibration. The process is illustrated in Fig. 3 [6]. This approach is used in scientific computing. The data are divided in two groups. The first group is for calibrating the models, and the second group is for validating the models. However, for general purpose code with application to regulatory issues, this may not be acceptable approach as it implies tuning the code for every application. The other studies provided mathematical foundation based on Bayesian approach for calibration [1,6,21]. The calibration described here is for improving the fidelity of the models in the code, in response to comparison with the data, and therefore is not a validation. Code should be validated with other independent tests that are separate from data for calibration. The term calibration has been used to improve accuracy of the measurements but is not the subject of this paper.

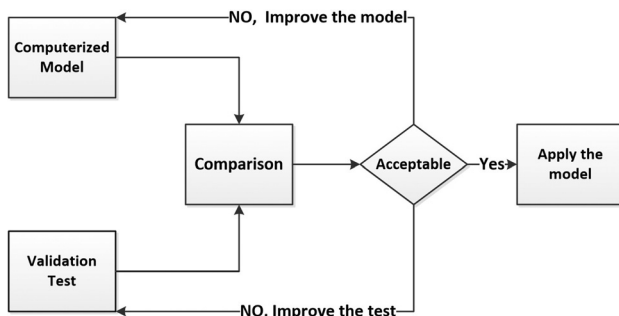


Fig. 3 Calibration process

5 Validation

Validation is a step where code's ability to predict the actual performance of a system is assessed. As indicated in Fig. 4, the validation is performed by comparing code predictions for tests with the data. The validation is always related to an application, guided by PIRT and with appropriate tests.

In Fig. 4, the conclusion from the comparison of prediction with data is acceptable, when the tests represent the facility of interest (Fig. 2). That is established by scaling study that is addressed in Sec. 6. Figure 4 also refers to frozen code. This implies that the validation is performed with a specified (frozen) version of the code, and results will only apply to that version of the code.

Phenomena identification and ranking table process is prerequisite for performing validation. As described earlier, PIRT is for a specific application for a given prototype. It identifies the important phenomena for this application.

After PIRT, relevant tests are identified that will be modeled with the code. The three types of tests are separate effect tests (SETs), component tests, and integral effect tests (IETs). The SETs are simpler tests and represent single or a group of few phenomena. They are generally well instrumented and can be at large scale. The component tests were sometimes placed in category of SETs. Many facilities include components that are not easily represented by the basic formulation in the code. These components are represented with empirical models obtained from standalone tests for these components. These components are also part of scaled integral facilities (IET). These component models are validated with the data not used for model development. The IETs represent the prototype facility at smaller scale. They test the ability of the code to integrate different sections and components undergoing different thermal-hydraulic conditions. These test facilities are generally designed for one application to minimize scale distortions but when applied to another application, they will have larger distortions. Still, they provide valuable data for code validation.

Validation also determines the accuracy of prediction of the models in the code. The contributions to uncertainty in the prediction from the sources such as numerics or initial and boundary conditions and geometric representations are minimized. However, in some applications, the numerical uncertainty is difficult to minimize, and the result of validation will be a combination of model uncertainty and numerical uncertainty. The validation provides two important information: Is code applicable to intended application and how much is the uncertainty in the prediction of important phenomena.

The uncertainty in prediction of individual phenomenon from SETs is essential in estimating the aggregate uncertainty in QoI. The uncertainty quantification is described in Sec. 7.

As illustrated in Fig. 5, domains or range of tests should cover the domain of application but is not always possible. The scale related distortions always exist and will contribute to the

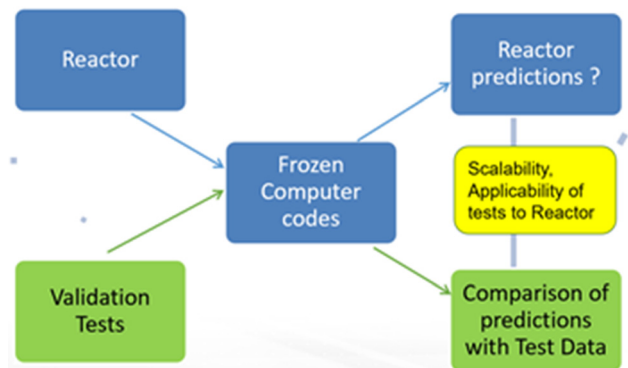


Fig. 4 Validation process

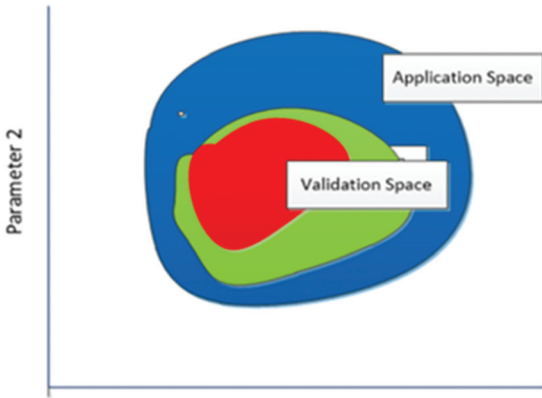


Fig. 5 Domains of application and validation

uncertainty in expected prediction of QoI. The contribution of scale distortion is described in Sec. 6. One approach that is followed is to have tests at different sizes to ensure code can predict phenomenon at different scales. These tests are called counterpart tests, and they increase the domain of validation [22–24].

In addition, Dinh et al. [25] pointed out that the tests produce large amount of data, but only small part is utilized for validation. The data have spatial and temporal details among many instruments, but the code only relies on some aggregate or average test values.

6 Scaling

Most systems or control volumes under consideration are characterized by quantities of interest. For fluid system, the region of interest is a control volume that is subjected to many forces that will affect the characteristics of the fluid in the control volume such as pressure and density. In solid mechanics, the region of interest is a solid body that can deform in presence of surface and internal forces. In case of motion of solid body, friction and external forces govern the quantity of interest such as velocity or acceleration. These external and internal forces are called agents of change.

In these examples, the quantity of interest characterizes the region under consideration, and a set of agents of change contribute to the change in this quantity of interest. These agents have different impact on the quantity of interest. A simulation model should correctly predict the relative contributions of individual agents. Scaling is applied to design experiments. Scaling identifies agents of change that affect the QoI and preserves their ranking between the actual application and the surrogate test. This will ensure that data will be relevant for the validation of simulation codes for *specific* application.

Validation is relevant only if the tests represent the intended application. In single-phase flow systems, the area to volume ratio plays a significant role and is major contributor to scale distortion. Smaller facilities have larger area to volume ratio, so the surface effects such as heat transfer or friction get magnified. The facilities must be designed to account for it. Unlike single-phase flow, in case of two-phase flow, the scale distortions have significant effect on the prediction. Beside surface to volume effect, the two phases exchange mass, momentum, and heat at the interface. The interface shape and interfacial area density depend on flow regimes that are scale dependent. So, if interfacial related phenomena are important and have distortion, they will have impact on prediction of the quantity of interest.

Scaling is an approach for designing appropriate test facilities for given application for code validation and for estimating scale distortions for alternate applications. The scaling methods also provide a quantitative estimate of PIRT ranking. The nondimensional groups in the global balance equations for application

represent the specific phenomenon, and their relative magnitude will determine the importance of the corresponding phenomenon [26].

In general, the two approaches of scaling are reductionist and global. In the reductionist approach [27], local balance equations are nondimensionalized, and nondimensionalized groups are identified. This approach leads to many groups and must be matched throughout the flow field, and their impact on QoI is not easily evident. This approach is useful in simplifying formulation like developing boundary layer equations.

In the global approach [19,28–30], the whole system is considered. The balance equations are integrated over the control volume of interest with exchange taking place at the boundaries and provide nondimensional groups that can be related to QoI. The two approaches for global scaling are H2TS [19] and fractional scaling analyses [20,31].

In the H2TS approach, two broad categories are top-down and bottom-up. The top-down looks at the system with different level of decomposition as indicated in Fig. 6. The H2TS starts from system of interest and moves deeper with subsystems, modules in subsystems such as pumps or separators. The modules may have different constituents such as fluid and structures that may exchange heat and momentum. For single constituent case with liquid and vapor phases, the interfacial exchange becomes important. Once the global level is identified, the volume averaged equations are cast in form of quantity of interest and agents of change that affect this quantity. Here, Ψ is quantity of interest or state variable, and φ_i are agents of change

$$\frac{d\Psi}{dt} = \sum_{i=1}^n \varphi_i \quad (1)$$

Each agent of change and QoI are nondimensionalized by their initial values. The coefficient of each agent of change represents the magnitude of each agent and indicates the rank of the phenomena represented by agent of change, and to go deeper in hierarchy for high ranked phenomena

$$\frac{d\Psi^*}{dt} = \frac{1}{\Psi_0} \sum_{i=1}^n \varphi_{i,0} \varphi_i^+ \quad (2)$$

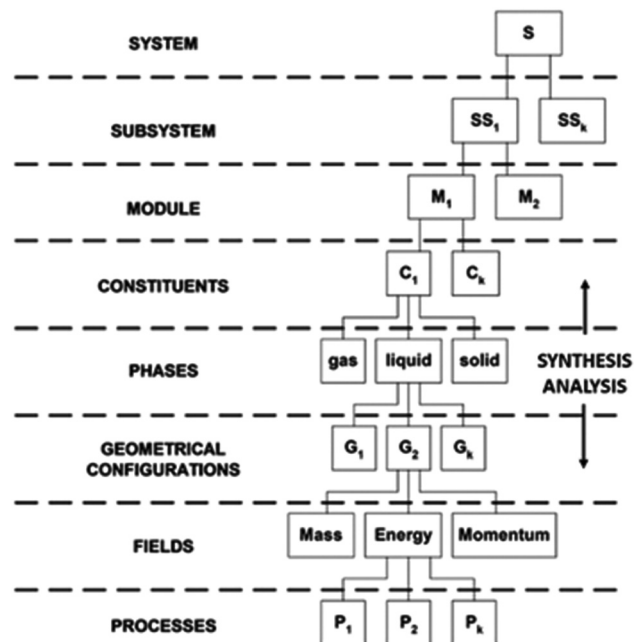


Fig. 6 Decomposition in H2TS method [19]

The scaling of three local phenomena is the bottom-up approach. In case of small modular reactor, the containment pressure is affected by processes such as condensation on the walls. Once that have been identified, the models for processes are considered for applicability. This is bottom-up approach. A good example of application of H2TS is scaling of AP600 facilities [32].

Fractional scaling analyses are like H2TS. However, the approach of nondimensionalizing is different. In this approach, fraction effects are estimated. An aggregate value ($\varphi_{\text{aggr},0}$) of reference values of each agent of change is estimated and applied for nondimensionalization

$$\varphi_{\text{aggr},0} = \sum_{i=1}^n \varphi_{i,0} \quad (3)$$

$$\varphi_i^* = \frac{\varphi_i}{\varphi_{\text{aggr},0}} \quad (4)$$

$$\frac{d\Psi^*}{dt^*} = \frac{t_{\text{ref}} \varphi_{\text{aggr},0}}{\Psi_0} \sum_{i=1}^n \varphi_i^* \quad (5)$$

$$t_{\text{ref}} = \frac{\Psi_0}{\varphi_{\text{aggr},0}} = \frac{1}{\omega_{\text{aggr}}} \quad (6)$$

Each agent of change can be represented by fractional rate of change (FRC). Larger is FRC, ω_i , more effective is the agent of change. In designing a facility, the FRCs should be matched

$$\frac{d\Psi^*}{dt} = \frac{1}{\Psi_0} \sum_{i=1}^n \varphi_{i,0} \varphi_i^+ = \omega_{\text{aggr}} = \sum_{i=1}^n \omega_i = \sum_{i=1}^n \frac{\varphi_{i,0}}{\Psi_0} \quad (7)$$

For meaningful validation, the tests should represent the phenomena appearing in the actual application. As was illustrated in Fig. 5, the test data may not cover the range of parameters expected in the application. If the region of application is bigger than region of validation, extrapolation of the discrepancy in the prediction may be required. There are counterpart tests that simulate same phenomenon at different scales and based on these counterpart tests, if the discrepancy decreases with size, the discrepancy for full size facility is expected to be less. As a conservative approach, the discrepancy for the largest size may be assumed to be same for the full-scale facility.

As per Fig. 2, for performing validation study to judge the code capability to simulate the phenomena or operation, a scaling analysis is needed to establish relevancy of tests. Some examples of scaling analyses for nuclear plants, advanced passive pressurized water reactor (AP600) [32] and small modular reactor [33] and economical simplified boiling water reactor [34], are in the literature as referred here.

7 Uncertainty Quantification

After establishing the applicability of the computerized model to a given problem, based on verification and validation, the next question is to estimate the total uncertainty in the prediction of quantity of interest. The five basic contributors to uncertainty are

initial and boundary conditions, geometrical representation, formulation of problem, model form based on empirical correlations in the code, and finally numerics. The uncertainties are propagated through different steps to reach final QoI prediction. Roache [35] has some general views on validation and uncertainty quantification that are for computational fluid dynamics (CFD) but are general enough to apply to other analyses.

Helton [36] has described quantification of margins and uncertainties as applied to reliability and safety of nuclear weapon stockpile. His paper describes statistical aspects of two types of uncertainties such as aleatory and epistemic. In aleatory uncertainty, the data have inherent randomness such as bubbly or droplet flows, where the sizes and interfacial density are always changing. The epistemic uncertainty is due to the lack of knowledge or inability to measure some parameters such as distribution of heat transfer from the structures to different phases in two-phase flow. The study [36] also describes the methods of addressing these uncertainties and their contribution to overall uncertainty as illustrated in Fig. 7.

Westinghouse has developed an uncertainties propagation method, ASTRUM (automated statistical treatment of uncertainty method) [37], that is like CSAU evaluation [15] and considers the effect of the uncertainties of the input parameters, computer-code models, initial and boundary conditions, geometry, scale effect and other application-specific input data, and solution algorithms on the predicted results for FOM or QoI. The ASTRUM differs from CSAU in the way that uncertainties are combined and the numbers of contributors to uncertainties that are considered. ASTRUM is based on most widely applied nonparametric methodology, first proposed by Gesellschaft für Anlagen- und Reaktorsicherheit, Global Research for Safety [38], that was based on well-established concepts and tools from probability and statistics given by Wilks [39]. The main advantage of this approach is that the number of calculations needed is independent of the number of uncertainty parameters to be considered. The necessary number of code calculations is given by the Wilks' formula [39]. The number of calculations depends only on the chosen tolerance limits (confidence) or intervals of the uncertainty statements of the results. Wilks' formula does not require the results to have a specific distribution (e.g., normal). The number of code calculations, N , for given confidence (tolerance) of γ and percentile of β , for one parameter uncertainty is given by Wilks' formula

$$\gamma = 1 - \beta^N \quad (8)$$

In case of multiple parameters of interest, p , that are independent, the number of calculations is higher and is given by the following relationship [38,39]:

$$\beta = \sum_{j=0}^{N-p} \frac{N!}{(N-j)!j!} \gamma^j (1-\gamma)^{N-j} \quad (9)$$

For example, for single parameter, the number of calculations needed for 95% confidence (β) of predicted QoI in 95% percentile (γ) is 59. In case of three independent parameters, the number of calculations required, for same level of confidence and percentile, increases to 124 based on Wilks' formulae described in Eq. (9).

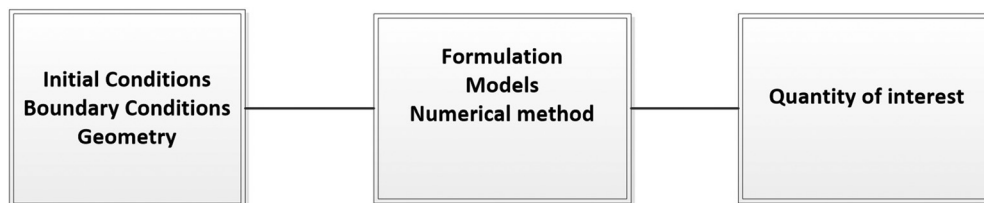


Fig. 7 General approach for uncertainty determination

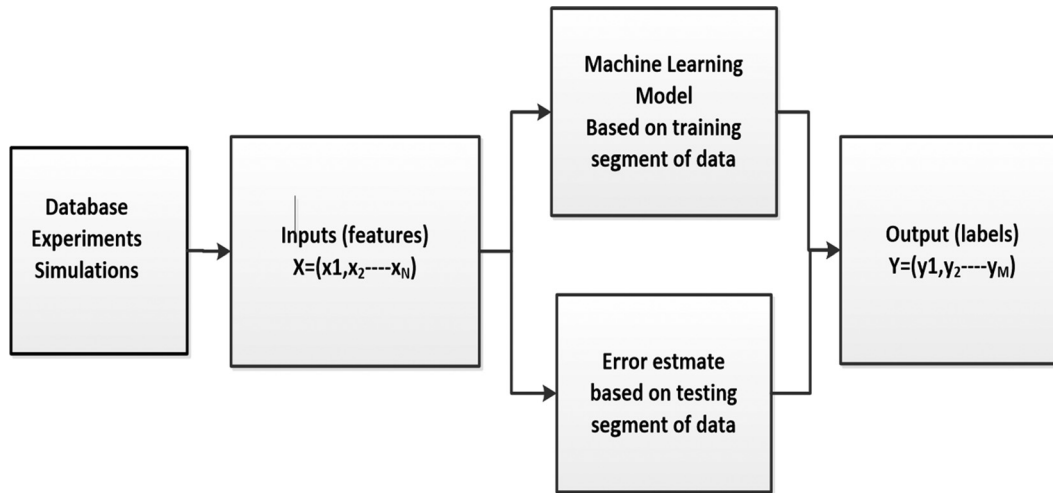


Fig. 8 Machine learning schematic

For practical application, the modeling uncertainties are represented by specific parameters that are either added or multiplied to the existing correlations. The state of knowledge is quantified by probability distributions of this parameter that includes the range and shape of distribution. Such a distribution expresses how well the appropriate value of an uncertain parameter of the code is known. A state of knowledge based on minimum information at the parameter level is expressed by *uniform distributions*. This is often considered the state of maximum ignorance. The uniform distribution is often chosen because to greatly increase occurrence of the probability of extreme values (i.e., values in the tails). As many distributions are normal, or like normal in practice, and because extreme values often result in increased uncertainties, the common practice is to assume a uniform distribution, when no other distribution can be justified.

The selection and quantification of these uncertainty parameters are based on experience gained from validating the computer codes by comparing the models' predictions with the data from integral tests and separate effect tests. The tests themselves are based on phenomena identified in the PIRT. Additionally, the uncertainty is increased to account for any scaling effects in the tests. Any statement of uncertainty in prediction is for a specific application.

In the ASTRUM approach [37], values of model parameters are randomly sampled from the distribution of the value of the parameter representing the corresponding uncertainty. The code is then run with set of values representing all the uncertainties considered. For every run, a new set of parameters are randomly selected from their distribution. The minimum number of code calculations depends on the requested probability content, and confidence level of the statistical tolerance limits in the uncertainty statements of the results and is computed from Eqs. (8) and (9) [38,39]. The set of predicted values of the FOM are arranged from the smallest to the largest value. In case of peak clad temperature in nuclear reactor accident simulations, the highest value will represent 95% percentile (that is 95% values will lower than this peak clad temperature) at a 95% confidence. In case of three independent FOMs, the set of values of three FOM from 124 runs are arranged in order. The top three values for each parameter represent the 95/95 conditions.

Generally, the SET/IET matrix will be different for different scenarios. The range and distribution of each parameter representing the model uncertainty will be specific to the transient of interest. While the ASTRUM approach is not limited by the number of parameters representing uncertainty, the largest effort in uncertainty estimation is in determining the range and distribution of these parameters. Therefore, PIRT helps in making problem

manageable. Among many papers that compare different approaches of uncertainty analyses, one is by Bucalossi et al. [40].

8 Machine Learning

Traditional validation involves simulating relevant tests with the code and comparing the predictions with the data. The data could be sparse. Another approach is to develop a machine learning (ML) models based on synthetic data from tests and high-fidelity simulations that can either be a predictive tool or can create relevant benchmark for validation.

Dinh and his team have done extensive work in the area of data driven modeling and application of ML in validation [41–45]. The details of applications can be found in these papers.

The ML approach consists of first organizing the available data in features or the inputs and labels that are output as represented in Fig. 8. This database can be augmented by using high-fidelity simulation. The dataset is divided into two segments: one for training and other for testing. Once feature set and label set have been established, algorithm from many available techniques such as neural network, support vector machine, decision tree, and linear regression random forest [45] can be applied to develop machine learning model from the training segment of the database. This ML model becomes a surrogate to the simulation methods. The approach also provides the accuracy of prediction from simulating testing segment of the database.

The machine learning model can then be applied for creating benchmarks. The low-dimensional or low-fidelity analytical tools can be validated with synthetic database. Radaideh and Kozlowski [46] have described an example for application of deep learning (neural network) for uncertainty quantification.

9 Summary

This paper summarizes the various concepts and steps that are needed to validate a computer code and to estimate the uncertainty in predictions.

Computer codes have been developed to simulate performance of systems or control volumes of concern, and they consist of conservation equations for mass, momentum, and energy along many constitutive relationships that have been in general empirically obtained. The predictions will always have uncertainty due to approximation in formulation and uncertainty in constitutive relationships.

Through verification process, errors in implementation in the codes are eliminated. However, uncertainty due to numerical scheme will still be present and is minimized.

The validation process is an evaluation of formulation and empirical constitutive relationships. This is accomplished by comparing code predictions with the data for SETs and IETs. As the validation is for an application, the test matrix should also be relevant to the intended application. The tests should simulate important phenomena as determined by PIRT process. This is ensured by scaling analyses for either existing tests or for designing new tests. The tests should have same phenomena as in the application and with the similar relative impact on the quantity of interest (or figure of merit). Validation with appropriate tests indicates that the code is applicable to the intended application.

In some cases, where the code is exclusive for an application, the models in the codes can be modified based on relevant data to minimize uncertainty in code predictions. This step is calibration and should be carefully executed to avoid compensating errors. The calibration, described here, is based on comparison of prediction with data but is a step for code improvement and generally not considered validation.

The last step, in the determination of the fidelity of the code application, is an estimate of uncertainty in the prediction of the quantity of interest. Many factors contribute to this aggregate uncertainty. These are boundary and initial conditions, empirical models or constitutive relationships, geometric simplifications, and numerics. The common approach is to estimate uncertainty distribution in these contributors except for numerics. The values of these contributors are sampled from their distributions, and the code is executed with different set of the contributors. This approach leads to a distribution of prediction from which mean and standard deviation can be calculated. In general, uncertainty is propagated in two ways. In a response surface approach, the code is replaced by a surrogate response surface that represents quantity of interest as a function of parameters representing important phenomena or agent of change. The important parameters are sampled from their uncertainty distribution, and distribution of QoI is obtained from the response surface. The second approach is nonparametric method [38]. The Wilks' formulae determine the minimum number of system calculations needed to determine the prediction with specified confidence. This approach saves computational effort.

Finally, machine learning is being investigated and applied for estimating scale distortion and code validation. This is alternate approach to traditional scaling and validation.

Funding Data

- Office of Nuclear Energy of U.S. Department of Energy (FT-22BN11010210; Funder ID: 10.13039/100006147).

Nomenclature

N = number of calculations
 t = time
 β = percentile
 γ = tolerance
 ϕ = agents of change
 Ψ = quantity of interest
 ω = fraction rate of change

Subscripts

Aggr = sum of fraction rate of changes or sum of agents of change
 Ref = reference values for scaling
 0 = initial values

Abbreviations

AP600 = advanced passive pressurized water reactor
 ASTRUM = automated statistical treatment of uncertainty method

CSAU = code scaling, applicability, and uncertainty
 FOM = figure of merit
 FRC = fractional rate of change
 IET = integral effect test
 ML = machine learning
 PIRT = phenomena identification and ranking table
 QoI = quantity of interest same as FOM
 QPIRT = quantitative PIRT
 SET = separate effect test
 VVUQ = verification, validation, and uncertainty quantification

References

- [1] Roache, P. J., 2009, "Perspective: Validation—What Does It Mean?," *ASME J. Fluids Eng.*, **131**(3), p. 034503.
- [2] Rohatgi, U. S., and Kaizer, J. S., 2020, "Historical Perspectives of BEPU Research in US," *Nucl. Eng. Des.*, **358**, p. 110430.
- [3] Freitas, C. J., 1993, "Comments on Policy Statement on the Control of Numerical Accuracy," *ASME J. Fluids Eng.*, **115**(3), pp. 339–340.
- [4] ASME V&V-20 Guideline, 2009, "Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer," ASME, New York, Standard No. ASME V&V-2009(R2021).
- [5] Computational Fluid Dynamics Committee, 2002, *Guide for the Verification and Validation of Computational Fluid Dynamics Simulations* (AIAA G-077-1998), AIAA, Reston, VA.
- [6] Oberkampf, W., and Roy, C. J., 2010, *Verification and Validation in Scientific Computing*, Cambridge University Press, Cambridge, UK.
- [7] Smith, R. C., 2014, *Uncertainty Quantification, Theory, Implementation and Application* (Computational Science and Engineering), SIAM, Philadelphia, PA.
- [8] Roache, P. J., 1998, *Verification and Validation in Computational Science and Engineering*, Hermosa Publishing, Albuquerque, NM.
- [9] Salari, K., and Knupp, P., 2000, "Code Verification by Methods of Manufactured Solutions," Sandia National Laboratories, Albuquerque, NM, Report No. SAND 2000-1444.
- [10] Roy, C. J., 2005, "Review of Code and Solution Verification Procedures for Computational Simulation," *J. Comput. Phys.*, **205**(1), pp. 131–156.
- [11] Roache, P. J., 2002, "Code Verification by the Method of Manufactured Solutions," *ASME J. Fluids Eng.*, **124**(1), pp. 4–10.
- [12] Etienne, S., Garon, A., and Pelletier, D., 2012, "Some Manufactured Solutions for Verification of Fluid-Structure Interaction Codes," *Comput. Struct.*, **106–107**, pp. 56–67.
- [13] Roache, P. J., 1994, "Perspective: A Method for Uniform Reporting of Grid Refinement Studies," *ASME J. Fluids Eng.*, **116**(3), pp. 405–413.
- [14] Cadafalch, J., Perez-Segarra, C. D., Consul, R., and Oliva, A., 2002, "Verification of Finite Volume Computations on Steady-State Fluid Flow and Heat Transfer," *ASME J. Fluids Eng.*, **124**(1), pp. 11–21.
- [15] Boyack, B., Catton, I., Duffey, R. B., Griffith, P., Katsma, K. R., Lellouche, G. S., Levy, S., 1989, "Quantifying Reactor Safety Margins—Application of Code Scaling, Applicability and Uncertainty Evaluation Methodology to a Large Break, Loss-of-Coolant Accident," USNRC, Washington, DC, Report No. NUREG/CR-5249.
- [16] Wilson, G., and Boyack, B., 1998, "The Role of PIRT Process in Experiments, Code Development and Code Applications Associated With Reactor Safety Analyses," *Nucl. Eng. Des.*, **186**(1–2), pp. 23–37.
- [17] Luo, H., 2012, "Quantified PIRT and Uncertainty Quantification for Computer Code Validation," Ph.D. thesis, Oregon State University, Corvallis, OR.
- [18] Yurko, J., and Buongiorno, J., 2012, "Quantitative Phenomena Identification and Ranking Table (QPIRT) for Bayesian Uncertainty Quantification," 2012 International Congress on Advances in National Power Plants (ICAPP-12), Chicago, IL, June 24–28, ANS.
- [19] Zuber, N., 1991, "An Integrated Structure and Scaling Methodology for Severe Accident Technical Issue Resolution, Appendix D," Report No. NUREG/CR-5809, R4.
- [20] Wulff, W., Zuber, N., Rohatgi, U. S., and Catton, I., 2009, "Application of Fractional Scaling Analysis (FSA) to Loss of Coolant Accidents (LOCA); System Level Scaling for System Depressurization," *ASME J. Fluids Eng.*, **131**(8), p. 121401.
- [21] Kennedy, M. C., and O'Hagan, A., 2001, "Bayesian Calibration of Computer Models," *J. R. Stat. Soc. B*, **63**(3), pp. 425–464.
- [22] Takeda, T., and Ohtsu, I., 2018, "Uncertainty Analysis of ROSA/LSTF Test by RELAP5 Code and PKL Counterpart Test Concerning PWR Hot Leg Break LOCAs," *Nucl. Eng. Technol.*, **50**(6), pp. 829–841.
- [23] Lorduy-Alós, M., Gallardo, S., and Verdú, G., 2020, "Scaling Analysis of an IBL/LOCA Counterpart Test Between the ATLAS and LSTF Facilities," *Prog. Nucl. Energy*, **127**(10), p. 103460.
- [24] Mascari, F., D'Auria, F., Bestion, D., Lien, P., Nakamura, H., Austregesilo, H., and Moon, S. K., 2020, "OECD/NEA/CSNI State-of-the-Art Report on Scaling in System Thermal-Hydraulics Applications to Nuclear Reactor Safety and Design (the S-SOAR)," OECD/NEA/CSNI Specialist Meeting Transient, Spain.
- [25] Dinh, N., Nourgaliev, R., and Lee, H., 2013, "Perspective on Nuclear Reactor Thermal Hydraulics," NURETH-15, Rich Data versus Poor Knowledge, Pisa, Italy, May 12–17.

- [26] Frepoli, C., 2020, "Scaling Analyses of Thermal-Hydraulics Integral Systems: Insight From Practical Applications and Recent Advancement," *Nucl. Sci. Eng.*, **194**(8–9), pp. 825–832.
- [27] Nahavandi, A., Castellana, F. S., and Moradkhanian, E. N., 1979, "Scaling Laws for Modeling Nuclear Reactor Systems," *Nucl. Sci. Eng.*, **72**(1), pp. 75–83.
- [28] Wulff, W., 1996, "Scaling of Thermal-Hydraulic System," *Nucl. Eng. Des.*, **163**(3), pp. 359–395.
- [29] D'Auria, F., and Galassi, G. M., 2010, "Scaling in Nuclear Reactor Thermal Hydraulics," *Nucl. Eng. Des.*, **240**(10), pp. 3267–3293.
- [30] Reyes, J., Frepoli, C., and Yurko, J. P., 2015, "The Dynamical System Scaling Methodology: Comparing Dimensionless Governing Equations With H2TS and FSA Methodologies," NURETH 16, Chicago, IL, Aug. 30–Sept. 4.
- [31] Zuber, N., Rohatgi, U., Wulff, W., and Catton, I., 2007, "Application of Fractional Scaling Analyses (FSA) to Loss of Coolant Accidents (LOCA) Methodology Development," *Nucl. Eng. Des.*, **237**(15–17), pp. 1593–1607.
- [32] Wulff, W., and Rohatgi, U. S., 1998, "System Scaling for the Westinghouse AP600 Pressurized Water Reactor and Related Test Facilities; Analyses and Results," Report No. NUREG/CR-5541.
- [33] Reyes, J., and King, J., 2003, "Scaling Analysis for the OSU System Integral System Test Facility," Report No. OSU-NE-2003-01.
- [34] Ishii, M., Revankar, S. T., Dowlati, R., Bertodano, M. L., Babelli, I., Wang, W., Pokharna, H., 1996, "Scientific Design of Purdue University Multi-Dimensional Integral Test Assembly (PUMA) for GE SBWR," Report No. NUREG/CR 6309.
- [35] Roache, P. J., 2016, "Verification and Validation in Fluids Engineering: Some Current Issues," *ASME J. Fluids Eng.*, **138**(10), p. 101205.
- [36] Helton, J. C., 2011, "Quantification of Margins and Uncertainties: Conceptual and Computational Basis," *Reliab. Syst. Saf.*, **96**(9), pp. 976–1013.
- [37] Frepoli, C., 2008, "An Overview of Westinghouse Realistic Large Break LOCA Evaluation Model," *Sci. Technol. Nucl. Install.*, **2008**, p. 498737.
- [38] Glaeser, H., 2008, "GRS Method for Uncertainty and Sensitivity Evaluation of Code Results and Applications," *Sci. Technol. Nucl. Install.*, **2008**, p. 798901.
- [39] Wilks, S., 1941, "Determination of Sample Sizes for Setting Tolerance Limits," *Ann. Math. Stat.*, **12**(1), pp. 91–96.
- [40] Bucalossi, A., Petrucci, A., Kristof, M., and D'Auria, F., 2010, "Comparison Between Best Estimate Plus Uncertainty Methods and Conservative Tools for Nuclear Power Plants Licensing," *Nucl. Technol.*, **172**(1), pp. 29–47.
- [41] Bao, H., Lin, L., and Dinh, N., 2020, "Machine Learning in Nuclear Thermal Hydraulics, Part 2: Opportunities and Perspectives," OECD/NEA/CSNI Specialist Meeting on Transient Thermal-Hydraulics in Water Cooled Reactors, CIEMAT, Madrid, Spain, Dec. 14–17.
- [42] Chang, C.-W., 2018, "Data Driven Modeling of Nuclear System Thermal Hydraulics," Ph.D. thesis, North Carolina State University, Raleigh, NC.
- [43] Hanna, B. N., 2018, "Coarse-Grid Computational Fluid Dynamics (CG-CFD) Error Prediction Using Machine Learning," Ph.D. thesis, North Carolina State University, Raleigh, NC.
- [44] Lin, L., Bao, H., and Dinh, N., 2020, "Machine Learning in Nuclear Thermal Hydraulics, Part 1: Fundamentals and Application," OECD/NEA/CSNI Specialist Meeting on Transient Thermal-Hydraulics in Water Cooled Reactors, CIEMAT, Madrid, Spain, Dec. 14–17.
- [45] Liu, Y., 2018, "Development of a Data Driven Framework for Boiling Problems With Multiphase CFD Solver," Ph.D. thesis, North Carolina State University, Raleigh, NC.
- [46] Radaideh, M. I., and Kozlowski, T., 2019, "Combining Simulations and Data With Deep Learning and Uncertainty Quantification for Advanced Energy Modelling," *Int. J. Energy Res.*, **43**(14), pp. 7866–7890.