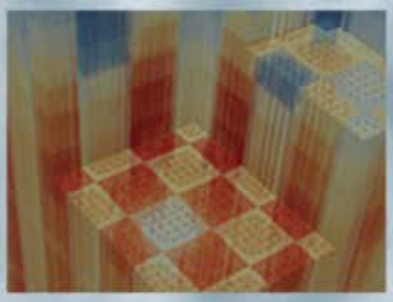
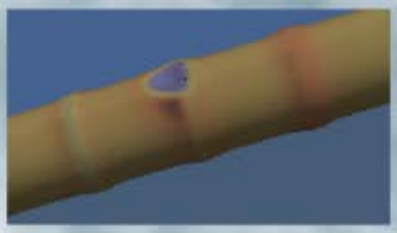
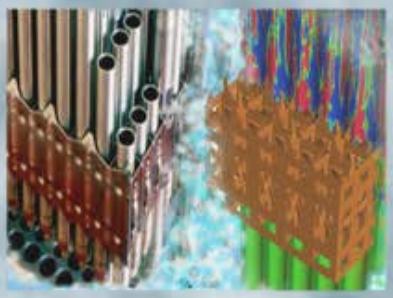
CASL-U-2015-0255-000

A Framework for Predictive Capability Maturity Quantification







Paridhi Athe, Nam Dinh
North Carolina State University

July 7, 2015

U.S. DEPARTMENT OF ENERGY | Nuclear Energy

CASL-U-2015-0255-000

A framework for Predictive Capability Maturity Quantification

Paridhi Athe, Department of Nuclear Engineering, North Carolina State University
Dr. Nam Dinh, Department of Nuclear Engineering, North Carolina State University

Introduction

Modeling and simulation permeate all fields of science and engineering. In nuclear engineering, numerical results generated by computer simulation codes have long been used in making decisions about nuclear reactor design, operation, and safety. In particular, comprehensive methodologies and systematic processes have been developed, adopted, and applied to guide the development of analytical tools and evaluate their adequacy for applications in nuclear reactor safety analysis. Notable are the "Code Scaling, Applicability, and Uncertainty" (CSAU) methodology, "Evaluation Model Development and Assessment Process" (EMDAP), both developed by the U.S. Nuclear Regulatory Commission (NRC) and "Predictive Capability Maturity Model" (PCMM) developed by Sandia National Laboratory. However, due to lack of data/evidence and the inherent risk associated with various nuclear engineering processes, the assessment process becomes very rigorous and time consuming, and when it comes to decision making its hard to answer in "Yes" or "No" if a certain code is "good enough" to describe a particular application of interest.

This work describes an initial effort to develop and demonstrate a systematic, formalized and computerized framework for quantification of reliability of a simulation tool for (a given) nuclear reactor engineering or safety application. It is noted that the present study is limited to "validation", which is an important component in PCMM along with other attributes considered in PCMM.

Boyack B.C., Catton I. et.al," Quantifying Reactor Safety Margins Part 1: An Overview of the Code Scaling, Applicability, and Uncertainty Evaluation Methodology," Nuclear Engineering and Design 119 (1990) 1-15
Richard G. Hills, Walter R. et. Al., **Development of a Fourth Generation Predictive Capability Maturity Model, SANDIA REPORT** SAND2013-8051, September 2013
Transient and Accident Analysis Methods, U.S. Nuclear Regulatory Commission, Regulatory Guide 1.203, December 2005

Fuzzy logic and code maturity quantification

A fuzzy set can be defined as a set with fuzzy boundaries. Fuzzy logic is theory of sets that calibrate obscurity or uncertainty. Unlike Boolean algebra that works on binary logic (Truth or false, 0 or 1), fuzzy logic characterizes the data using degree of membership. It provides a precise way of representing approximate reasoning and imprecise information. A fuzzy set "A" of universe of discourse X, with elements represented by x, is defined by its membership function (MF), $\mu_A(x)$, as $\mu_A(x): X \rightarrow [0,1]$

where, $\mu_A(x) = 0$, if x does not belongs to A,
$\mu_A(x) = 1$, if x completely belongs to A,
and $0 < \mu_A(x) < 1$, if x partially belongs A

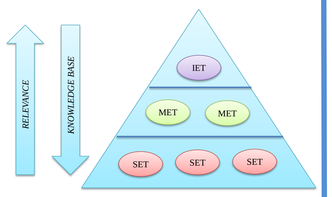
Since its conception, fuzzy logic has find wide application in different areas of engineering and science . There are several feature of fuzzy logic that makes it suitable choice for developing a quantifiable framework for assessing the reliability of simulation tools used in nuclear engineering :

- Fuzzy logic provides an efficient methodology for representing incomplete knowledge and imprecise information using degree of belief or degree of confidence in the truth value of the information.
- It can handle heterogeneity very well. Extraction of important information from large amount of data is easier.
- Availability of large number of mathematical operation (min, max, product, average, sum, etc.) makes it a very efficient tool for developing an expert inference system.
- Simplicity and flexibility of fuzzy logic makes it possible to combine fuzzy logic with other methodologies. e.g. Genetic fuzzy systems are fuzzy system developed by combining fuzzy logic and genetic algorithms. Similarly, neuro-fuzzy system are developed by combining fuzzy logic with neural networks.
- CSAU, EMDAP and PCMM are all expert elicitation methodologies. Expert judgement plays a critical rule at each level/stage of these methodologies. Fuzzy logic provide a unique technique to codify the expert knowledge which makes it very attractive tool for developing a quantifiable framework for code's maturity evaluation.

Zadeh, Lotfi A. "Fuzzy sets." *Information and control* 8.3 (1965): 338-353.
Zadeh, Lotfi A. "Fuzzy logic, neural networks, and soft computing."*Communications of the ACM* 37.3 (1994): 77-84.

Governing factors and Membership function

Validation Pyramid : In lieu of full-scale plant testing, a hierarchical approach is adopted for code validation. A set of relevant separate-effect tests (SET), mixed-effect tests (MET) and integral-effect tests (IET) is used to construct a validation pyramid. We assume that each layer consist of a finite number of independent experiments, that together describe the knowledge base of that layer in the validation hierarchy.

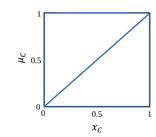


Validation Pyramid

Coverage : The range of applicability or coverage of a particular application is decided on the basis of operating range of its primary control parameters (like pressure, mass flow rate and temperature). We assume that our application of interest is governed by a single control parameter P.

Coverage ratio is defined by, $x_c = \frac{RE_l - RE_u}{RA_l - RA_u}$

here, RE_l and RE_u represents the lower and upper limit of P for the experiment; RA_l and RA_u represents the lower and upper limit of P for the application. Therefore, x_c lies between 0 and 1, 0 when experiment does not lies in the range of experiment and 1 when experiment covers the entire range of application.

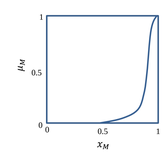


$\mu_C = x_C$

MF for Coverage

Measurement Uncertainty: Code validation is based on the comparison of code prediction and measurements. However, due to instrumentation errors and limited resolution of measurement technique, there is a finite measurement uncertainty associated with each experiment. Experiments with higher measurement uncertainty cannot be used for code validation analysis. Confidence in the measurements is expressed by a variable x_M defined by, $x_M = 1 - MU$

here, MU is the measurement uncertainty which is assumed to lie between 0-100%. Hence, the variable x_M lies between 0 and 1.

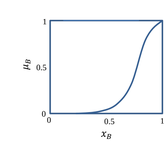


$\mu_M = exp\left(-\frac{(x_M - 1)^2}{0.021}\right)$

MF for Measurement uncertainty

Bias Uncertainty: Bias uncertainty can be defined as the statistical measure of the difference between code prediction and measurement. If measurement uncertainty is small, bias uncertainty is an adequate measure of the code deficiency. Confidence in the code prediction is expressed by a variable x_B defined by $x_B = 1 - BU$

here, BU is the bias uncertainty that is assumed to lie between 0-100%. Hence, the variable x_B lies between 0 and 1.



$\mu_B = exp\left(-\frac{(x_B - 1)^2}{0.065}\right)$

MF for Bias Uncertainty

Scaling : Experiments used for validating the code are scaled version of the real application. The relevance of these experiment to the real application is determined on the basis of different scaling factors like material scaling, geometric similarity and physics scaling. We assume that all experiments used in the validation pyramid are of reactor prototypic level. Therefore, we do not consider effect of scaling in this analysis.

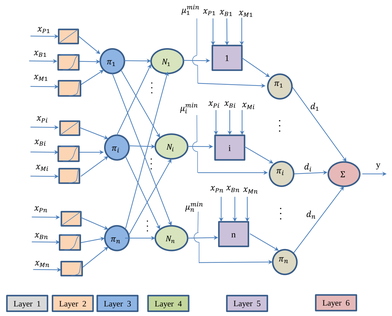
Summary Remarks

This work presents a quantitative framework for assessing the maturity of a simulation code. In essence, it provides an efficient methodology that assimilates objective data (evidence) with subjective data (based on expert knowledge) for a codes maturity quantification. This method integrates expert's knowledge in the framework by using fuzzy membership function. These membership functions help in weighting all the evidence based on the of degree of confidence in their truth value. These evidences are combined using different fuzzy operations to obtain the total maturity function of the code for a specific application of interest.

Evaluation of assessment base

This step assess the completeness of a particular layer in the validation pyramid on the basis of code bias uncertainty, coverage and measurement uncertainty of all the experiments in that layer. The assessment result gives a number y that represents the evaluation of the respective layer on a scale of 0 to 1 using all the evidences that constitute the knowledge base of that layer. Steps of evaluation:

Step 1(Specification of input) : Crisp inputs, x_C, x_B and x_M for each validation test are specified.



Evaluation of each layer of Validation Pyramid

Step 2 (Fuzzification) : Fuzzification helps in weighting the evidence on the basis of expert knowledge.

The fuzzified input variable corresponding to x_{Ci},x_{Bi} and x_{Mi} are represented by μ_{Ci},μ_{Bi} and μ_{Mi}, respectively (where i is the experiment index)

Step 3 (Rule) : $\pi_i = \mu_{Ci}.\mu_{Bi}.\mu_{Mi}$

Step 4 (Normalization): $N_i = \frac{\pi_i}{\pi_1.\pi_2....\pi_n}$

Step 5 (Defuzzification) : Defuzzification is performed according to the following rule –

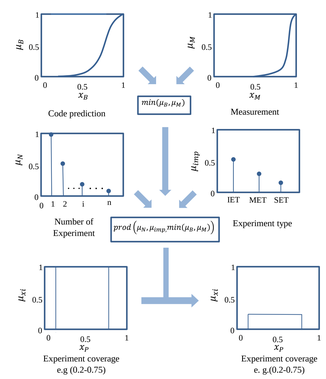
$$d_i = N_i.\mu_i^{min}\left(\frac{x_{Ci} + x_{Bi} + x_{Mi}}{3}\right)$$

Step 6 (Summation) : Summation of the defuzzified output variable gives y, $y = d_1 + d_2 + \cdots + d_i + .. + d_n$

We can calculate value of y for each layer of the validation pyramid and obtain a quantitative measure of completeness of that layer.

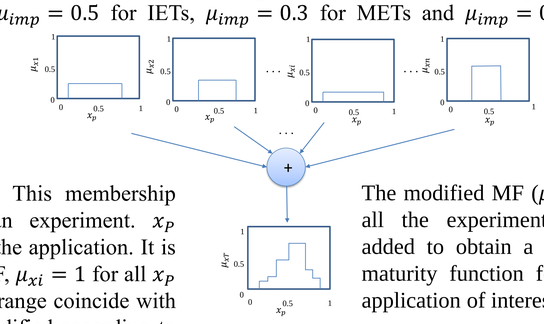
Assessment of entire knowledge base

In this section, we construct a maturity function that provides a graphical representation of the code's predictive capability over the entire range of operation of the application.



Here, we consider three additional MF function in addition to the MFs described in the previous sections: (1) MF for number of experiments, μ_N: This MF is defined according to the number of experiments in its constitutive layer, e.g. if the bottom layer consists of M experiments, than μ_N for any experiment belonging to this layer would be (1/M).

(2) Based on the subjective knowledge about a particular type of experiment (viz. SET or MET or IET), we assign weight to each type of experiment using expert's knowledge about the importance of that experiment. Assumed membership values for each type of experiment are: $\mu_{imp} = 0.5$ for IETs, $\mu_{imp} = 0.3$ for METs and $\mu_{imp} = 0.2$ for SETs



(3) MF for Experiment Coverage, μ_{xi}: This membership function represents the coverage of an experiment. x_P represent the entire range of operation of the application. It is normalized to lie between 0 and 1. The MF, $\mu_{xi} = 1$ for all x_P for which the application and experiment range coincide with each other. This membership value is modified according to the membership value of all other evidences (code bias, measurement uncertainty, type and importance of experiment) related to this experiment.

The modified MF (μ_{xi}) of all the experiments are added to obtain a global maturity function for the application of interest.

This Maturity function (μ_{xT}) represents the degree of confidence in the code's predictive capability over the entire operating range of the application.

Acknowledgements

This work is performed for the U.S. Department of Energy's Consortium for Advanced Simulation of Light Water Reactors (CASL) to support application of the Predictive Capability Maturity Model to the CASL's Virtual Environment for Reactor Applications (VERA) codes

Department of NUCLEAR ENGINEERING

CASL-U-2015-0255-000