Nuclear Engineering and Technology 51 (2019) 968–976

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Verification of Reduced Order Modeling based Uncertainty/Sensitivity Estimator (ROMUSE)

Bassam Khuwaileh [a, *], Brian Williams [b], Paul Turinsky [c], Donny Hartanto [a]

[a] Department of Mechanical and Nuclear Engineering, University of Sharjah, P.O.BOX, 27272, Sharjah, USA
[b] Los Alamos National Laboratory, P.O. Box 1663, Los Alamos, NM, 8754, USA
[c] Department of Nuclear Engineering, North Caroline State University, Raleigh, NC, 27695-7926, USA

ARTICLE INFO

Article history:
Received 6 November 2018
Received in revised form
5 January 2019
Accepted 30 January 2019
Available online 31 January 2019

Keywords:
Sensitivity/uncertainty estimation
CASL
VERA
DAKOTA
SCALE
ROMUSE 2.0

ABSTRACT

This paper presents a number of verification case studies for a recently developed sensitivity/uncertainty code package. The code package, ROMUSE (Reduced Order Modeling based Uncertainty/Sensitivity Estimator) is an effort to provide an analysis tool to be used in conjunction with reactor core simulators, in particular the Virtual Environment for Reactor Applications (VERA) core simulator. ROMUSE has been written in C++ and is currently capable of performing various types of parameter perturbations and associated sensitivity analysis, uncertainty quantification, surrogate model construction and subspace analysis. The current version 2.0 has the capability to interface with the Design Analysis Kit for Optimization and Terascale Applications (DAKOTA) code, which gives ROMUSE access to the various algorithms implemented within DAKOTA, most importantly model calibration. The verification study is performed via two basic problems and two reactor physics models. The first problem is used to verify the ROMUSE single physics gradient-based range finding algorithm capability using an abstract quadratic model. The second problem is the Brusselator problem, which is a coupled problem representative of multi-physics problems. This problem is used to test the capability of constructing surrogates via ROMUSE-DAKOTA. Finally, light water reactor pin cell and sodium-cooled fast reactor fuel assembly problems are simulated via SCALE 6.1 to test ROMUSE capability for uncertainty quantification and sensitivity analysis purposes.

1. Introduction

1.1. ROMUSE2.0

Providing best estimate predictions with error bounds in nuclear reactor simulation calculations is a vital need by regulatory and industrial bodies. Therefore, the ability to quantify the uncertainties and sensitivities in model predictions is a common pursuit within the area of reactor modeling and simulation.

ROMUSE [1] is an effort within the Consortium for Advanced Simulation of Light water reactors (CASL) to provide an analysis tool to be used in conjunction with reactor core simulators, in particular the Virtual Environment for Reactor Applications (VERA) core simulator [2]. ROMUSE is written in C++ and is currently capable of performing various types of parameter perturbations and associated sensitivity analysis, uncertainty quantification, surrogate model construction and subspace analysis. The current version 2.0 has the capability to interface with the Design Analysis Kit for Optimization and Terascale Applications (DAKOTA) code [3], which gives ROMUSE access to the various algorithms implemented within DAKOTA. The code is mainly designed to interface with VERA and the Comprehensive Modeling and Simulation Suite for Nuclear Safety Analysis and Design (SCALE) code [4]. Although SCALE has also modules to perform the sensitivity and uncertainty calculations, ROMUSE offer benefits associated with its capability to construct reduced order, surrogate models used in place of SCALE. The benefit for Monte Carlo based uncertainty quantification application is derived from the reduced computer execution time of the surrogate model versus SCALE. The benefit for Markov Chain-Monte Carlo based model calibration is derived from both the reduced computer execution time and reduction in the number of parameters. For sensitivity analysis, computer execution time savings are realized by ROMUSE from the reduction in the number of parameters. Also note that assuming a linear relationship between

* Corresponding author.
E-mail address: bkhuwaileh@sharjah.ac.ae (B. Khuwaileh).

https://doi.org/10.1016/j.net.2019.01.020

the response of interest and parameters, sometimes used in uncertainty quantification and model calibration to avoid the need for Monte Carlo sampling thereby reduce computer execution time, need not be made when employing a surrogate model.

ROMUSE also can interface with any general model (e.g. python and MATLAB) with Input/Output (I/O) format that utilizes the Hierarchical Data Format 5 (HDF5) [5]. The algorithms within ROMUSE are discussed in detail in Ref. 1, 6 and 7. It is also designed to be used in conjunction with reactor analysis codes (e.g. reactor core simulators). ROMUSE interfaces with the I/O of the simulator of interest such that the I/O data are wrapped, modified and then used in ROMUSE modules to complete analysis for the problem of interest. ROMUSE can be utilized stand-alone or interfaced with DAKOTA to complete the analysis. DAKOTA can provide ROMUSE with access to a number of novel and efficient algorithms for uncertainty quantification, surrogate construction, model calibration and much more. More information on DAKOTA can be found in Ref. 3.

There are various sequences in ROMUSE, each having different parameters and performing a different algorithm or execution flow [1]. Comprehensive Uncertainty Quantification (UQ) studies can be performed via ROMUSE, where ROMUSE supports three different UQ methods:

1. Brute force Monte Carlo UQ.
2. Multi-Physics Karhunen-Loeve (KL) expansion based UQ (utilizes the Multi-Physics Efficient Range Finding Algorithm – MP-EUQ in the case of coupled multi-physics codes).
3. Surrogate Based Monte Carlo UQ (SBUQ).

For more information about these algorithms consult Ref. 6. Algorithms 2 and 3 require performing dimensionality reduction when the core simulator requires substantial computer resources, and in addition.

- for Algorithm 2 when there are a large number of input parameters and adjoint solutions are not available or there are a large number of Responses of Interest even when adjoint solutions are available; and
- for Algorithm 3 when there are a large number of input parameters.

This can be done via ROMUSE, which implements efficient algorithms for single physics and multi-physics dimensionality reduction that are explained in detail in Ref. 6. These algorithms are used to perform dimensionality reduction on the uncertainty source space i.e. revealing the active or important degrees of freedom (DoFs). Once the active DoFs are determined they can be used to perform linear KL-based uncertainty quantification, surrogate construction, or can be communicated to DAKOTA.

Perturbing input parameters is a vital feature for any analysis code as many mathematical and statistical analyses such as UQ and Model Calibration (MCal) require manipulating the parameter set. The ROMUSE perturbation module is designed such that it allows the user to manipulate the input parameters of any given model as long as the parameters are formatted in the Hierarchical Data Format 5 (HDF5) [5] format or SCALE cross-sections library binary format [8] or VERA cross-section library format [2, 8, 9]. The ability to perturb any parameters in HDF5 formatting allows ROMUSE to work with any simulator with such I/O format.

1.2. Verification of ROMUSE

To establish the credibility of any computational tool, rigorous verification and validation (V&V) must be performed. Model V&V is necessary in order to provide engineering predictions with quantified confidence [10]. Therefore, model V&V procedures and applications are needed.

Verification is the process of determining that a model implementation accurately represents the developer's conceptual description of the model and its solution. On the other hand, validation is the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model. Generally, verification requires comparing the performance of the model of interest with other V&V'ed models or problems with known analytical solutions. This paper presents a number of verification case studies for ROMUSE by comparing its solutions and analysis algorithms with those provided for four problems representing various levels of complexity.

The overarching goal is to verify ROMUSE capability to estimate the sensitivities and uncertainties using Reduced Order Modeling (ROM). At the core of ROM is the ability to calculate the basis of the so-called active subspace. Once determined, the basis can be used to construct surrogates and therefore, estimate uncertainties and sensitivities efficiently. In this work, four case studies will be used to complete the verification of the uncertainty/sensitivity capabilities in ROMUSE:

1. A quadratic problem: this problem is used to test the Gradient Based-Range Finding Algorithm (GB-RFA). The problem is characterized by a known analytical solution which shows the fundamental capability of computing the basis of the active subspace as implemented in ROMUSE.
2. The Brusselator problem: this non-linear coupled problem is used to test the newly developed surrogate construction capability, which is available through the ROMUSE-DAKOTA interface. This problem implements a well-studied coupled problem and resembles ROMUSE capabilities in constructing surrogate models for coupled problems.
3. A pressurized water reactor (PWR) pin cell uncertainty/sensitivity criticality problem via surrogate modeling. In this case study, the bases are calculated and then used to estimate the sensitivities and uncertainties for the infinite multiplication factor k_{inf}.
4. A sodium-cooled fast reactor (SFR) assembly model uncertainty/sensitivity problem via SCALE6.1. In this case study a surrogate is constructed and then used to estimate the uncertainty contribution of the various nuclide-reactions in the multiplication factor.

2. Verification case studies

2.1. Quadratic problem

In this case study, a quadratic model with a parameter space $\in \mathbb{R}^{10}$ is used:

$$f(x) = \frac{1}{2}x^T \boldsymbol{A} x \qquad x \in [-1, 1]^{10} \tag{1}$$

whose gradient is given by

$$\nabla_x f(x) = \boldsymbol{A} x \tag{2}$$

Since $\boldsymbol{A}$ is known, its column space and row space are known. Therefore, the Gradient Based - Range Finding Algorithm (GB-RFA) [7] can be tested against matrix $\boldsymbol{A}$ features. The GB-RFA should capture approximately the column space of matrix $\boldsymbol{A}$. ROMUSE 2.0 can perform the GB-RFA on any model by collecting snapshots of the gradient vectors. In order to verify the ROMUSE version of the GB-RFA algorithm, its results can be compared to known results as

now described.

Engineering models usually solve the governing equations via iterative numerical techniques; therefore, we do not have access to the analytical form of the model relating the free parameters, x to the solution, $f(x)$. However in this case study the matrix $\boldsymbol{A}$ is known, so we can compute the exact target of the GB-RFA and compare it to ROMUSE2.0 calculations. Via singular value decomposition (SVD) we can compute the orthonormal basis for the column space ($\mathbf{U}$):

$$\boldsymbol{A} = \boldsymbol{USV}^T \tag{3}$$

where the columns of matrix $\boldsymbol{U}$ form a basis for the column space of matrix $\boldsymbol{A}$, while the columns of matrix $\boldsymbol{V}$ form a basis for the row space of matrix $\boldsymbol{A}$, and finally matrix $\boldsymbol{S}$ is a diagonal matrix containing the singular values on its diagonal. Each singular value represents the importance of the corresponding column of $\boldsymbol{U}$ in representing the column space of matrix $\boldsymbol{A}$. The GB-RFA can compute an approximation ($\boldsymbol{U}$') to the matrix $\mathbf{U}$. In order to test the similarity of the active spaces represented by these matrices, three different tests will be performed:

1. Measuring the angle between the two spaces: the angle between the two spaces represents the degree of agreement between the basis vectors by measuring the angle between these vectors.
2. Computing and comparing the singular values.
3. Measuring the error in representing vectors drawn from the column space of the matrix [6],

$$\boldsymbol{error} \leq \boldsymbol{C}\max_{\mathrm{i=1..p}}\left\| \left(\boldsymbol{I} - \boldsymbol{U}^{'\boldsymbol{nxr}}\boldsymbol{U}^{'\boldsymbol{nxr}T}\right) * \bar{s}_i \right\|_2 = \epsilon_{upper} \tag{4}$$

where p snapshots $\bar{s}_i$ are taken from the column space of matrix $\boldsymbol{A}$ and $\boldsymbol{C}$ is a constant scalar.

A 10×10 random matrix was generated with a priori known rank ($r = 6$), implying if the GB-RFA employs 6 snapshots the approximated active space should be exact within numerical precision. First the angle between the two spaces has been calculated and found to be effectively 0 rads (8.44×10^{-16}), which means that the basis of the computed basis vectors is in complete agreement (refer to Table 1). Both singular values (Fig. 1) and the computed error analysis (Fig. 2) indicate the same rank and illustrate the agreement between the GB-RFA (performed by ROMUSE2.0) and the known analytical values.

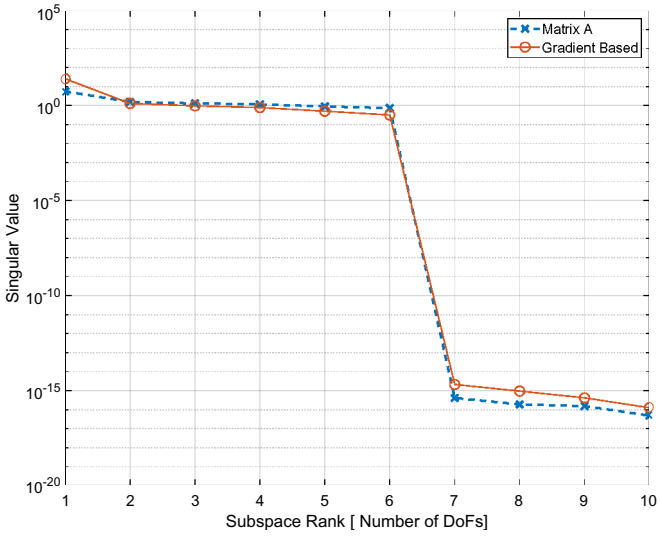

Fig. 1. Singular values.

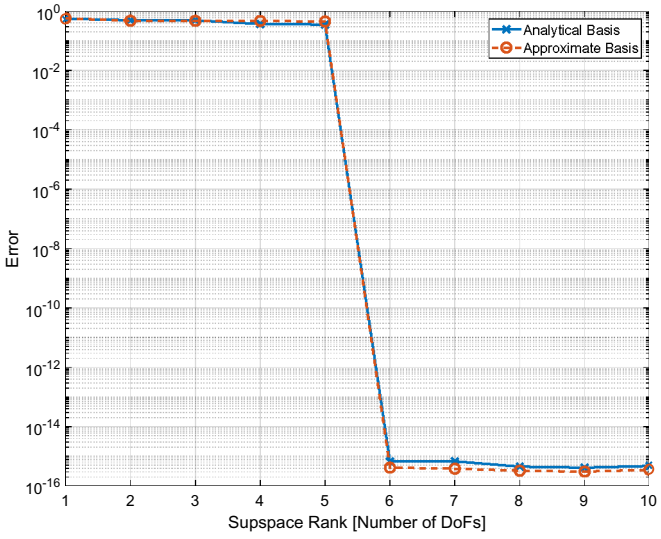

Fig. 2. Error upper bound.

2.2. The Brusselator problem

The Brusselator is a theoretical model for a type of autocatalytic reaction. Reaction–diffusion models frequently arise in the study of various scientific and engineering systems. The Brusselator system resembles reaction diffusion in chemical reactions. British mathematician Alan Turing proved that a particular mathematical system could produce spatial patterns from an arbitrary initial state. In its most generic form a Turing model describing chemical diffusion and reactions with respect to the chemical concentrations U and V can be written in the form:

$$\frac{dU}{dt} = D_U\nabla^2 U + f(U,V) \tag{5}$$

$$\frac{dV}{dt} = D_V\nabla^2 V + g(U,V) \tag{6}$$

where D_U and D_V are the diffusion coefficients setting the magnitude of diffusion for chemical species concentrations U and V, respectively. These two equations represent the Brusselator problem, which is a coupled problem determining the concentration of chemical species U and V. The coupling via the reaction kinetics is typically nonlinear.

A reaction-diffusion model often corresponds to real chemical reactions and reaction formulas. The Brusselator model used in this case study is one of the simplest chemical models exhibiting a pattern forming instability called Turing instability. In the case of the Brusselator model the chemical reactions are given as:

Table 1
Results for quadratic problem.

Algorithm	SVD − Rank (r)	Error-Rank (Eq. (4))	Angle (Theta)
Matrix A	6	6	8.44e-16 rads
GB-RFA	6	6	

$$A \rightarrow U \tag{7}$$

$$B + U \rightarrow V \tag{8}$$

$$2U + V \rightarrow 3U \tag{9}$$

$$U \rightarrow E \tag{10}$$

where the concentrations of the chemicals A, B and E are kept constant. One can derive the reaction kinetics equations corresponding to the reaction scheme just defined. The reaction kinetics of the Brusselator model is given by Ref. [11]:

$$f(U,V) = A - (B+1)U + U^2V \tag{11}$$

$$g(U,V) = BU - U^2V \tag{12}$$

therefore, the transient equations would be:

$$\frac{dU}{dt} = U_t = D_U\nabla^2 U + A - (B+1)U + U^2V \tag{13}$$

$$\frac{dV}{dt} = V_t = D_V\nabla^2 V + BU - U^2V \tag{14}$$

where A and B are scalar parameters, which govern the pattern selection in the model by defining the reaction kinetics. In order to obtain spatial patterns, it is always required that $D_U \neq D_V$. In this case study, the spatial variations will be ignored (implying diffusion can be ignored), implying that the Brusselator system of equations will be used and solved at a certain point in space and for a certain time interval. Setting $A = 1$, $B = 3$, and ignoring diffusion, the Brusselator equations are:

$$\frac{dU}{dt} = 1 - (4*U) + U^2V \tag{15}$$

$$\frac{dV}{dt} = 3*U - U^2V \tag{16}$$

The solution of these two equations will be used to test ROMUSE capability to build surrogate models in conjunction with DAKOTA. First the reference solution is calculated, shown in Figs. 3 and 4 for the concentrations of species U and V for a time interval of 20 s. Based on the reference solution, ROMUSE-DAKOTA is used to build a Gaussian Process (GP) surrogate model. The GP surrogate form can be represented via the following:

$$f(t) = \sum_{i=1}^{7} a_i * e^{-\left(\frac{t-b_i}{c}\right)^2} \tag{17}$$

where $\{b_i\}$ collects the 7 known times at which the process was observed, the $\{a_i\}$ are known functions of the observed process output, and c > 0 is an unknown correlation length parameter estimated by DAKOTA. In general, practitioners often select approximately 10 locations per input dimension at which to observe the process for surrogate construction.

Once constructed, the surrogate model will be compared to the reference solution. In this case study, the role of ROMUSE is to generate the samples, and pass information to and from DAKOTA.

Fig. 5 presents the residuals for $\boldsymbol{U}(\boldsymbol{t})$ which are the differences between the reference solutions and surrogate model predictions, and Table 2 presents the goodness of fit results and compares them to a corresponding Gaussian Mixture Model (GMM) MATLAB fit for the same data. Similarly, Fig. 6 represents the residual for $V(\boldsymbol{t})$. Note

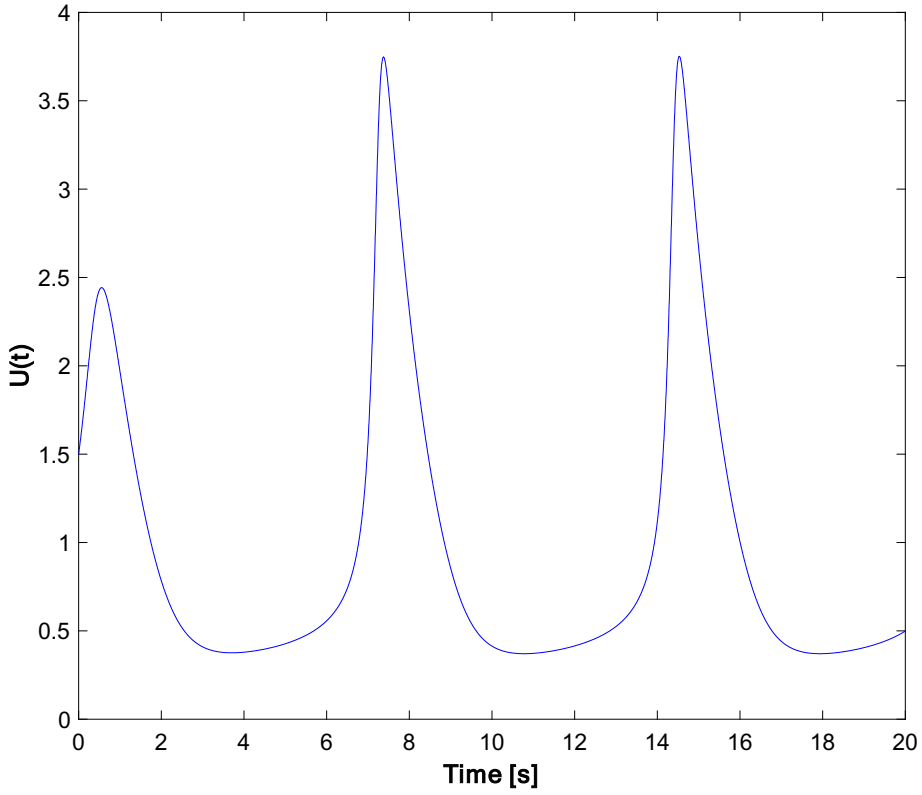

Fig. 3. $\boldsymbol{U}(\boldsymbol{t})$ Reference solution.

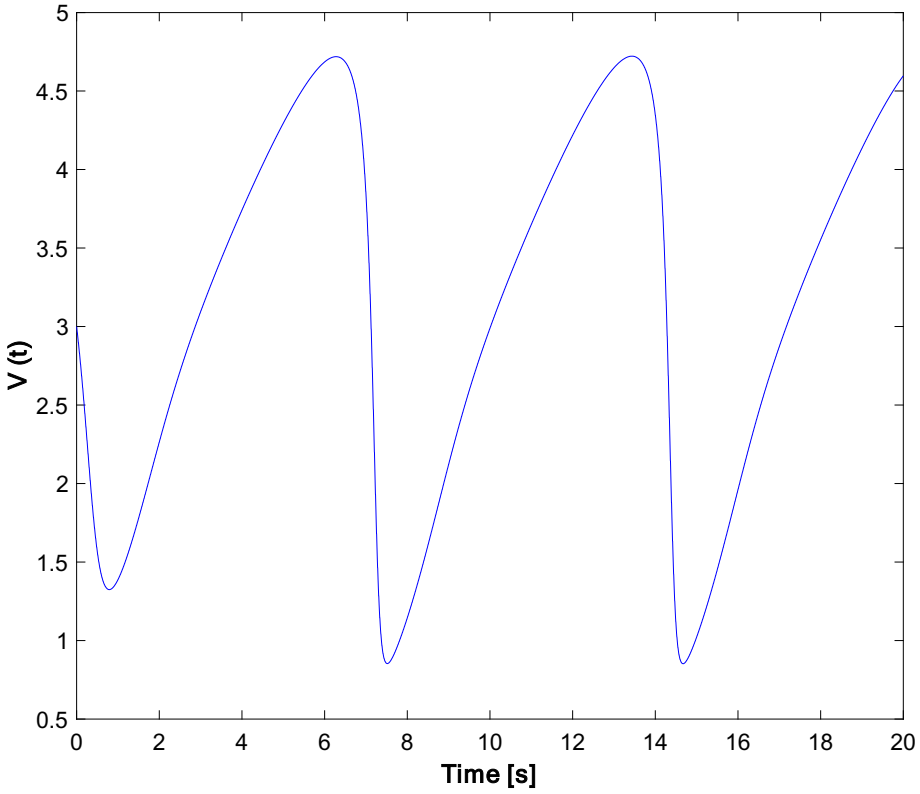

Fig. 4. $\boldsymbol{V}(\boldsymbol{t})$ Reference solution.

Table 2
Goodness of fit $-\boldsymbol{U}(\boldsymbol{t})$

ROMUSE-DAKOTA		MATLAB	
R-square:	0.996	R-square:	0.998
RMSE:	0.0541	RMSE:	0.041
Evaluation points	1000	Evaluation Points	1000

Table 3
Goodness of fit $-\boldsymbol{V}(\boldsymbol{t})$

ROMUSE-DAKOTA		MATLAB	
R-square:	0.9527	R-square:	0.922
RMSE:	0.2638	RMSE:	0.307
Evaluation Points	1000	Evaluation Points	1000

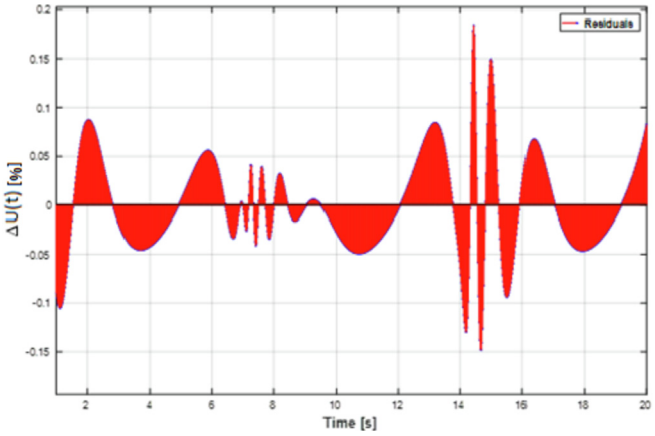

Fig. 5. $\boldsymbol{U}(\boldsymbol{t})$ Relative surrogate residuals- ROMUSE-DAKOTA.

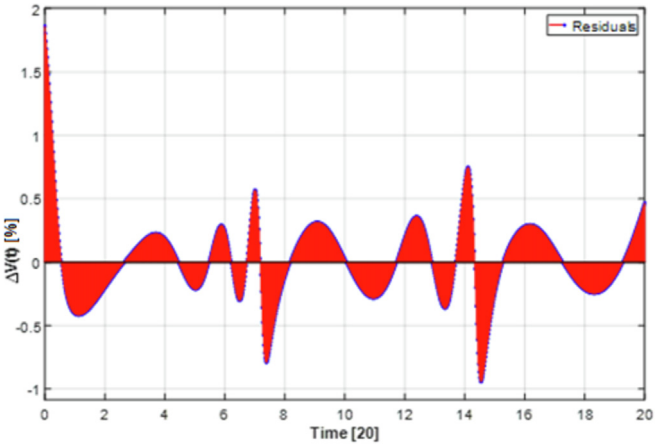

Fig. 6. $\boldsymbol{V}(\boldsymbol{t})$ Relative surrogate residuals- ROMUSE-DAKOTA.

that these residuals are calculated based on comparing the validation set with the mean GP surrogate predictions. Table 3 summarizes the goodness of fit as compared to a GMM MATLAB fit. The slight difference between ROMUSE-DAKOTA and MATLAB results may be due to the difference in the algorithm employed (EM algorithm for MATLAB [12] and Surfpack algorithm for DAKOTA [3]). Note that the goodness of fit in the case of $\boldsymbol{U}(\boldsymbol{t})$ is better than that of $\boldsymbol{V}(\boldsymbol{t})$ due to the fact that the hyper-parameters (kernel function type, kernel function scale) were optimized to match the $\boldsymbol{U}(\boldsymbol{t})$ observations and then used to fit both $\boldsymbol{V}(\boldsymbol{t})$ and $\boldsymbol{U}(\boldsymbol{t})$. This situation resembles the case where two coupled multiphysics are available and we have access to observations from one model (e.g. $\boldsymbol{U}(\boldsymbol{t})$) and access to observations from the coupled calculations ($\boldsymbol{U}(\boldsymbol{t})$, $\boldsymbol{V}(\boldsymbol{t})$).

2.3. Verification against SCALE uncertainty and sensitivity calculations

This case study introduces a Monte Carlo based test to verify the ability to estimate the individual uncertainty contribution of each model parameter using ROMUSE2.0. In this test ROMUSE will be used to determine the active subspace for the parameters, and then used in a sensitivity study to determine the contribution of each uncertainty contributor in the subspace. Once determined, the basis vector can be used to define and solve a linear system of equations, based on linearizing the model of interest, to determine the uncertainty contribution of each DoF in conjunction with the Monte Carlo based samples.

Ref. 14 introduces the so-called Efficient Subspace Method (ESM). The ESM reduces the number of forward model runs used to calculate the sensitivity coefficients. On the other hand, this case study uses the Monte Carlo based samples and the influential DoFs in order to estimate the contribution of the parameters in the Monte Carlo UQ (MCUQ) analysis. Therefore, while Ref. 14 reduces the number of model runs, this case study shows ROMUSE2.0 capability to equip the MCUQ with a technique to rank the parameters according to their uncertainty contribution.

2.3.1. Algorithm

This section introduces the analysis followed by ROMUSE2.0 results to estimate the sensitivity coefficients, S_{yx} in Eq. (19) and therefore the corresponding uncertainty contribution. To start, consider the following model:

$$\overline{y} = f(\overline{x}) \tag{18}$$

If we assume that the model is linear, the uncertainty in the input parameter vector ($\overline{x}$) can be propagated towards the Response of Interest (RoI) vector ($\overline{y}$) using the sandwich equation as follows:

$$\boldsymbol{C}_y = \boldsymbol{S}_{yx}\boldsymbol{C}_x\boldsymbol{S}_{yx}^T \tag{19}$$

where the square root of the diagonal elements of $\boldsymbol{C}_y$ are the standard deviations (uncertainties) in the responses of the $\overline{y}$ vector. $\boldsymbol{C}_x$ is the covariance matrix of the input parameter ($\overline{x}$) and $\boldsymbol{S}_{yx}$ represents the sensitivity profile of the RoI ($\overline{y}$) with respect to the input parameter ($\overline{x}$). However, taking into account that the covariance matrix $\boldsymbol{C}_x$ is symmetric, then its singular value decomposition can be written as follows:

$$\boldsymbol{C}_x = \boldsymbol{U}_{\boldsymbol{C}_x}\sum\nolimits^2_{\boldsymbol{C}_x}\boldsymbol{U}^T_{\boldsymbol{C}_x} \tag{20}$$

where $\boldsymbol{U}_{\boldsymbol{C}_x}$ is the matrix of the orthonormal eigenvectors of the space spanned by the columns of matrix $\boldsymbol{C}_x$ and $\sum^2_{\boldsymbol{C}_x}$ is a diagonal matrix of the corresponding eigenvalues denoting the variances. Hence, Eq. (19) can be rewritten as follows:

$$\boldsymbol{C}_y = \boldsymbol{S}_{yx}\boldsymbol{U}_{\boldsymbol{C}_x}\boldsymbol{\Sigma}_{\boldsymbol{C}x}^2\boldsymbol{U}^T_{\boldsymbol{C}_x}\boldsymbol{S}^T_{yx} = \boldsymbol{S}_{yx}\underbrace{\boldsymbol{U}_{C_x}\boldsymbol{\Sigma}_{\boldsymbol{C}x}}_{\boldsymbol{C}_x^{1/2}}\underbrace{\boldsymbol{\Sigma}_{\boldsymbol{C}x}\boldsymbol{U}^T_{C_x}}_{\boldsymbol{C}_x^{1/2,T}}\boldsymbol{S}^T_{yx} \tag{21}$$

Once a representative orthonormal matrix is determined ($\boldsymbol{U}$) which approximates $\boldsymbol{U}_{\boldsymbol{C}_x}$ by ignoring its column vectors associated with small singular values. Computing this lower dimensional approximation matrix can be performed via various techniques that are reviewed in Ref. [13]. Once matrix $\boldsymbol{U}$ is computed, the covariance matrix ($\boldsymbol{C}_y$) can be approximated as follows:

$$\boldsymbol{C}_y \approx \left(\boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\boldsymbol{U}\right)\left(\boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\boldsymbol{U}\right)^T \tag{22}$$

This conclusion leads to the realization that the uncertainty can be evaluated via r (where r is the number of columns in matrix $\boldsymbol{U}$) model executions instead of n executions (where n is the original number of parameters). Each model execution would quantify the uncertainty in the RoI due to a certain basis vector (degree of freedom in the uncertainty sources space). Notice that the term ($\boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\boldsymbol{U}$), the term in brackets in Eq. (22), can be written as:

$$\boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\boldsymbol{U} = \left[\boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\overline{u}_1 \,\middle|\cdots\middle|\, \boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\overline{u}_r\right] \tag{23}$$

Defining $\overline{\omega}_i = \boldsymbol{C}_x^{1/2}\overline{u}_i$ one obtains:

$$\boldsymbol{S}_{yx}\boldsymbol{C}_x^{1/2}\boldsymbol{U} = \boldsymbol{S}_{yx}\left[\boldsymbol{C}_x^{1/2}\overline{u}_1\middle|\cdots\middle|\boldsymbol{C}_x^{1/2}\overline{u}_r\right] = \boldsymbol{S}_{yx}[\overline{\omega}_1|\cdots|\overline{\omega}_r] = \boldsymbol{S}_{yx}\boldsymbol{W} \tag{24}$$

where $\boldsymbol{W} = [\overline{\omega}_1|\cdots|\overline{\omega}_r]$. One can think of the vector $\overline{\omega}_i$ as a vector of weights that gives every element a certain contribution in the overall uncertainty. Following this logic, one can formulate the following linear system of equations that can be solved to obtain a reduced order estimate of the unknown sensitivity coefficients $\boldsymbol{S}_{yx}$:

$$\boldsymbol{R}^T = \boldsymbol{W}^T\boldsymbol{S}^T_{yx} \tag{25}$$

where the columns of $\boldsymbol{R}$ denote the change in the RoI (*i.e.* $\Delta\overline{y}$). If we are looking at the uncertainty in a single RoI, then $\boldsymbol{R}^T$ and $\boldsymbol{S}^T_{yx}$ will be vectors instead of matrices.

The main issue now is that Eq. (25) denotes an underdetermined system, which means that it has infinitely many solutions (if any!). Fortunately, we can construct a full rank system of equations that has a unique solution. In order to do so we have to realize that once we have the active subspace basis (i.e. $\boldsymbol{U}$), one can write any input snapshots as a linear combination of the basis vectors:

$$\overline{x}^{nx1} = \sum_{i=1}^{n}\beta_i\overline{u}_{C_{x,i}} \approx \sum_{i=1}^{r}\alpha_i\overline{u}_i = \boldsymbol{U}^{nxr}\,\overline{\alpha}^{rx1} \tag{26}$$

Now the response variation ($\Delta\overline{y}^j$) to a perturbation $\Delta\overline{x}^j$ can be written in terms of the derivatives of the RoI with respect to the reduced input variable (α^{rx1}):

$$\Delta\overline{y}^j = \frac{\partial\overline{y}}{\partial\alpha_1}\cdot\Delta\alpha_1^j + \ldots + \frac{\partial\overline{y}}{\partial\alpha_r}\cdot\Delta\alpha_r^j = \boldsymbol{S}_{y\alpha}\cdot\Delta\overline{\alpha}^j \tag{27}$$

where $\frac{\partial\overline{y}}{\partial\alpha_i}$ denotes the directional derivative of $\overline{y}$ in direction $\overline{u}_i$. The directional derivatives denote the sensitivity coefficients $\boldsymbol{S}_{y\alpha}$, where $\boldsymbol{S}_{y\alpha}$ is a vector if the problem has only one RoI and can be obtained by evaluating $\overline{y}$ a total of $r+1$ times as now explained. Choosing the perturbations $\Delta\overline{x}^j$ as

$$\Delta\overline{x}^j = \overline{\omega}_j = \boldsymbol{C}_x^{1/2}u_j \tag{28}$$

the vector $\Delta\overline{\alpha}^j$ can be calculated using Eq. (26):

$$\Delta\overline{\alpha}^j = \boldsymbol{U}^T\Delta\overline{x}^j = \boldsymbol{U}^T\boldsymbol{C}_x^{1/2}u_j \tag{29}$$

Then the response of $\overline{y}$ can be calculated for each $\Delta\overline{\alpha}^j$ allowing Eq. (25) to be written as

$$\boldsymbol{R}^T = \boldsymbol{A}^T\boldsymbol{S}^T_{y\alpha} \tag{30}$$

where

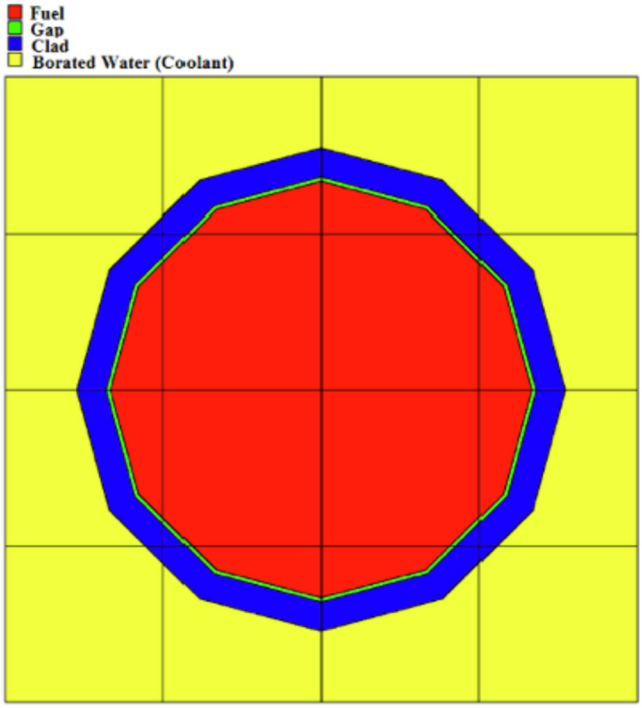

Fig. 7. LWR fuel pin model.

$$\boldsymbol{A} = \left[\Delta\bar{\alpha}^1 \ldots \Delta\bar{\alpha}^r\right] \tag{31}$$

Once Eq. (30) is solved for $\boldsymbol{S}_{y\alpha}$, the reduced order sensitivities $\boldsymbol{S}_{yx}$ can be estimated using the chain rule and Eq. (26):

$$\boldsymbol{S}_{yx} = \boldsymbol{S}_{y\alpha}\boldsymbol{S}_{\alpha x} = \boldsymbol{S}_{y\alpha}\boldsymbol{U}^T \tag{32}$$

Knowing the reduced order estimate of the sensitivity profile, the sandwich equation can be used to calculate an estimate of the uncertainty contribution for each source in the input space. In the case of MCUQ, matrix $\boldsymbol{A}$ represents the random set of perturbed parameters and $\boldsymbol{R}$ is the matrix of corresponding response samples.

The following steps summarize the proposed technique, given that the DoFs (columns of matrix **U**) have been calculated in advance:

1 Generate the samples of the parameters matrix **W**.
2 Collect the corresponding RoI samples and generate the matrix $\boldsymbol{R}$.
3 Using Eq. (29), project the parameter samples onto the active subspace ($\boldsymbol{U}$) yielding matrix **A**.
4 Use Eq. (30) to estimate the sensitivity matrix $\boldsymbol{S}_{y\alpha}$.
5 Use Eq. (32) to map the sensitivity matrix from the reduced space ($S_{y\alpha}$) to the original space (S_{yx}).
6 Knowing the sensitivity matrix ($\boldsymbol{S}_{yx}$), rate the uncertainty contribution of each parameter.

2.3.2. PWR pin cell model

In this numerical test, uncertainty quantification will be performed via the proposed approach, and then the estimated contributions will be compared to those estimated via the sensitivity factors calculated by SCALE 6.1 [15]. This example uses a nuclear fuel pin cell model (cf. Fig. 7) simulated via SCALE with the goal to determine the uncertainty in the infinite multiplication factor (k_{inf}). Fig. 7 shows a representation of the 2-dimensional geometry of the fuel pin cell. At the center (red region) is the fuel pellet which contains the uncertainty source associated with microscopic cross section values, the fuel pellet is surrounded by a helium gap (green region). The Zr-4 alloy cladding (blue region) and borated water (yellow region), that serves as a coolant and moderator, are also shown. In this example, a single, fresh fuel pin cell is extracted from the reactor core model, therefore the reference k_{inf} is 1.172437.

Table 4 presents a comparison of uncertainty contribution results between the proposed technique and the SCALE sensitivity-based analysis. The top eight contributors in the overall uncertainty are reported and compared. The overall uncertainty in k_{inf} is 0.57% (sensitivity based $\%\frac{\Delta k}{k}$) while the overall MC based uncertainty is 0.55% $\left(\%\frac{\Delta k}{k}\right)$. The proposed approach depends on the predetermined DoFs and their precision in representing the parameter space. In this case study, an error tolerance of 1% was used in calculating the subspace basis for the DoFs (i.e. the error in Eq. (4)). Given more precision in the calculated basis (DoFs), the results in Fig. 8 converge towards the MC estimated contributions. The MC based uncertainties in Table 4 are calculated using samples drawn assuming the corresponding covariance matrix (column 2 in Table 4). Therefore, for each nuclide-reaction covariance matrix, MC samples of the multiplication factor where collected and the corresponding relative uncertainty contributions were computed ($\%\frac{\Delta k}{k}$). Sufficient Monte Carlo samples were drawn such that the statistical uncertainty associated with Monte Carlo can be considered negligible with reference to the subspace error.

2.4. Sodium-cooled fast reactor assembly model

In this case study a fuel assembly model of a typical sodium-cooled fast reactor (SFR) [14,16] as shown in Fig. 9 is modeled. The assembly has 217 fuel pins and each fuel pin consists of U−10Zr metallic fuel (red region) contained inside HT9 cladding (dark blue region). The sodium coolant (light blue region) and the fuel rods are located inside the HT9 duck (green region). The ratio of the pitch to the pin diameter is about 1.14. The SCALE package TSUNAMI-2D

Table 4
Comparison of Monte Carlo based versus sensitivity-based uncertainty contributions.

Index	Covariance matrix Nuclide (reaction)	MC based Uncertainty $\left[\%\frac{\Delta k}{k}\right]$	Sensitivity based $\left[\%\frac{\Delta k}{k}\right]$	Relative Error
1	$U^{238}(n,\gamma) - U^{238}(n,\gamma)$	3.6321E-01%	3.6153E-01%	0.5%
2	$U^{235}(\bar{\nu}) - U^{235}(\bar{\nu})$	2.5012E-01%	2.5321E-01%	1.2%
3	$U^{235}(n,\gamma) - U^{235}(n,\gamma)$	1.5543E-01%	1.6112E-01%	3.5%
4	$U^{238}(n,n') - U^{238}(n,n')$	1.4520E-01%	1.6422E-01%	11.6%
5	$U^{235}(\chi) - U^{235}(\chi)$	1.3411E-01%	1.2343E-01%	8.7%
6	$U^{235}(n,f) - U^{235}(n,\gamma)$	1.2432E-01%	1.1652E-01%	6.7%
7	$U^{235}(n,f) - U^{235}(n,f)$	1.1265E-01%	1.1301E-01%	0.3%
8	$U^{238}(\bar{\nu}) - U^{238}(\bar{\nu})$	1.0212E-01%	9.8597E-02%	3.6%

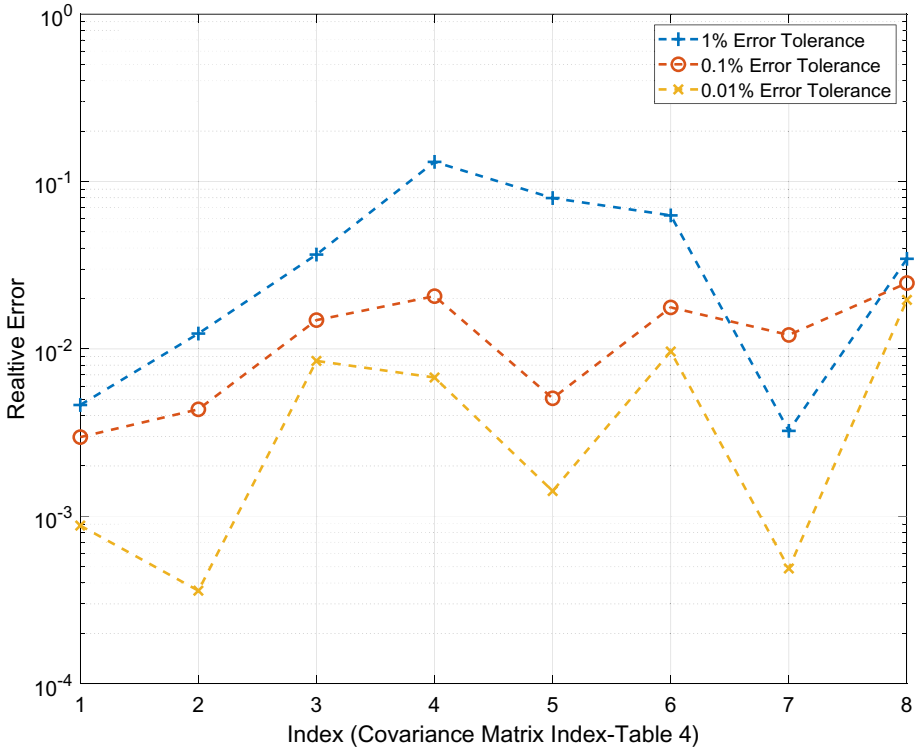

Fig. 8. Converging towards the MC estimation - Different Error Tolerance.

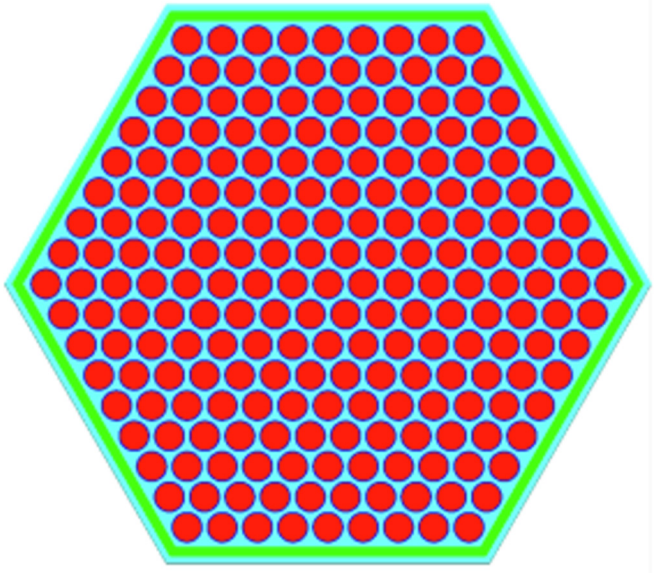

Fig. 9. SFR fuel pin model.

[13] is employed to complete the analysis. TSUNAMI-2D uses NEWT as the neutronic solver and SAMS to apply the adjoint method to estimate the sensitivity; hence, the uncertainty is estimated linearly by combining the uncertainty library and the sensitivity coefficients using Eq. (19).

The reference value of k_{inf} is 1.20195 with an uncertainty $\frac{\Delta k}{k} = 2.5121\%$ estimated via TSUNAMI-2D. In this case study, a polynomial surrogate is constructed for k_{inf} using ROMUSE. The formulated surrogate takes the following form:

$$\Delta k_{inf}^{surr} = \sum_{i=1}^{r} s_i \Delta\alpha_i + \sum_{i=1}^{r}\sum_{j\geq i}^{r} s_{ij}\Delta\alpha_i\Delta\alpha_j \tag{33}$$

where $\Delta\alpha_i$ is the perturbation of the ith element of the reduced cross-sections vector (the cross-sections vector projected on the active subspace – Eq. (29)) and s_i can be seen as the first order sensitivity coefficient while s_{ij} are the elements of the Hessian matrix. The values for s_i and s_{ij} are estimated via least squares.

Table 5 compares the uncertainty contributions as obtained using TSUNAMI-2D and ROMUSE 1st order surrogate (the first term in Eq. (33)) and those obtained via a 2nd order surrogate (Eq. (33)). It is clear that the second order approximation predicts the contributors with a better accuracy than the predictions of the 1st order surrogate. Moreover the first order surrogate approximation results in a higher relative error with a maximum of 8.1% while the 2nd order surrogate relative error is much lower in general.

3. Summary and conclusions

In this work, four verification case studies have been used to verify the range finding algorithm, surrogate construction and uncertainty quantification capabilities in ROMUSE2.0. The first case study compared the range finding algorithm performance for a quadratic problem with known solution. The second case study used ROMUSE to construct a surrogate model for the Brusselator problem. The predictions of the ROMUSE surrogate and the Brusselator model were compared. The third and fourth problems are criticality problems. The third problem is an uncertainty quantification for a PWR pin cell model. The uncertainty was quantified via a Monte Carlo sampling scheme and compared to that estimated via the algorithm presented in section 2.3.1 and used by ROMUSE to estimate the overall uncertainty and the contribution of each

Table 5
Comparison of TSUNAMI-2D based versus 2nd order surrogate based uncertainty contributions.

No	Covariance matrix (nuclide-reaction)	TSUNAMI-2D based Uncertainty $\left[\%\frac{\Delta k}{k}\right]$	2nd Order Surrogate based Uncertainty $\left[\%\frac{\Delta k}{k}\right]$	2nd Order Relative Error	1st order Surrogate based Uncertainty $\left[\%\frac{\Delta k}{k}\right]$	1st Order Relative Error
1	$U^{235}(n,\gamma) - U^{235}(n,\gamma)$	2.0334%	1.9433%	4.43%	1.9073%	6.20%
2	$U^{238}(n,n') - U^{238}(n,n')$	1.3549%	1.3264%	2.10%	1.3067%	3.56%
3	$U^{238}(n,\gamma) - U^{238}(n,\gamma)$	0.3780%	0.3697%	2.21%	0.3512%	7.10%
4	$U^{235}(\chi) - U^{235}(\chi)$	0.2902%	0.2793%	3.75%	0.3104%	7.98%
5	$U^{235}(fission) - U^{235}(fission)$	0.2151%	0.2139%	0.56%	0.2126%	1.20%
6	$U^{238}(\overline{\nu}) - U^{238}(\overline{\nu})$	0.1962%	0.1921%	2.04%	0.2042%	−4.09%
7	$U^{238}(elastic) - U^{238}(n,n')$	−0.1396%	−0.1353%	3.10%	−0.1283%	8.10%
8	$U^{235}(\overline{\nu}) - U^{235}(\overline{\nu})$	0.1305%	0.1297%	0.57%	0.1292%	1.01%
9	$U^{235}(n,n') - U^{235}(n,n')$	0.1016%	0.1014%	0.23%	0.1009%	0.69%
10	$Na^{23}(elastic) - Na^{23}(elastic)$	0.0864%	0.0873%	−1.07%	0.0835%	3.41%
11	$Na^{23}(n,n') - Na^{23}(n,n')$	0.0696%	0.0702%	−0.91%	0.0523%	6.75%
12	$U^{235}(elastic) - U^{235}(n,\gamma)$	−0.0642%	−0.0621%	3.27%	−0.0660%	−2.83%

parameter. The top individual nuclide-reaction type contributors to the overall uncertainty were reported to agree with a maximum relative error of 11.6%, indicating a non-linear dependency of the k_{inf} on the $U^{238}(n,n') - U^{238}(n,n')$ reaction covariance. In the fourth case study ROMUSE was used to replace the neutronics solver NEWT with a polynomial surrogate model. The uncertainties were estimated via TSUNAMI-2d and a surrogate model for a SFR assembly model. Results showed that the major individual nuclide-reaction type uncertainty contributors can be revealed via the 2nd order surrogate model with a maximum relative error of 4.43% in the corresponding uncertainty.

The overarching goal of ROMUSE is to equip the analyst with a tool to perform uncertainty quantification and sensitivity analysis for nuclear reactor models and simulators. The work reported upon here serves as a basis to verify ROMUSE, so that it can be used with confidence to complete sensitivity analysis, uncertainty quantification, surrogate model construction and subspace analysis.

Acknowledgments

This research was supported by the Consortium for Advanced Simulation of Light Water Reactors (http://www.casl.gov), an energy innovation hub (http://www.energy.gov/hubs) for modeling and simulation of nuclear reactors under the US Department of Energy contract number DE-AC05- 00OR22725. In addition, parts of this research have been funded by the University of Sharjah, Project Number 1702040775-P.

References

[1] B. Khuwaileh, et al., "ROMUSE 2.0 User Manual", LA-UR-16-27623, Los Alamos National Laboratory, 2016.
[2] J.A. Turner, "Virtual Environment for Reactor Applications (VERA): Snapshot 3.1," CASL-U-2013-0164-000, Oak Ridge National Laboratory, 2013.
[3] B.M. Adams, et al., "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 5.0 User's Manual," SAND-2010-2183, Sandia National Laboratories, 2010.
[4] Steven M. Bowman, SCALE 6: comprehensive nuclear safety analysis code system, Nucl. Technol. 174 (2) (2011) 126–148.
[5] S. Koranne, "Hierarchical Data Format 5: HDF5," in Handbook of Open Source Tools, Page 191-200, Springer, 2011.
[6] B.A.A. Khuwaileh, "Scalable Methods for Uncertainty Quantification, Data Assimilation and Target Accuracy Assessment for Multi-Physics Advanced Simulation of Light Water Reactors," PhD Dissertation, North Carolina State University, 2015.
[7] R.C. Smith, Uncertainty Quantification: Theory, Implementation, and Applications, SIAM, 2013.
[8] M.L. Williams, et al., in: SCALE Nuclear Data Covariance Library," ORNL/TM-2005/39 Version 6 Sect M, vol. 19, 2009.
[9] B. Kochunas, et al., Overview of development and Design of MPACT: Michigan parallel characteristics transport code, in: Proceedings of 2013 International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, Sun Valley, ID, USA, May 5-9, 2013.
[10] Robert G. Sargent, Verification and validation of simulation models, J. Simulat. 7 (1) (2013) 12–24.
[11] E.H. Twizell, et al., "A second-order scheme for the "Brusselator" reaction–diffusion system, J. Math. Chem. 26 (4) (1999) 297–316.
[12] Dellaert F. reportThe Expectation Maximization Algorithm. College of Computing, Georgia Institute of Technology. Technical Report GIT-GVU-02–20. 2002 Feb.
[13] Paul G. Constantine, Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies, vol. 2, SIAM, 2015.
[14] H.S. Abdel-Khalik, et al., Overview of hybrid subspace methods for uncertainty quantification, sensitivity analysis, Ann. Nucl. Energy 52 (2013) 28–46.
[15] Oak Ridge National Laboratory, SCALE: A Comprehensive Modeling and Simulation Suite for Nuclear Safety Analysis and Design, ORNL, 2011. ORNL/TM-2005/39, Version 6.1.
[16] K. Chihyung, et al., Uranium enrichment reduction in the prototype gen-IV sodium-cooled fast reactor (PGSFR) with PbO reflector, Nucl. Eng. Technol. 48 (2016) 351–359.