Polynomial Regression with Derivative Information for Uncertainty Analysis of Complex Simulation Models

Oleg Roderick
Argonne National Laboratory
Email: roderick@mcs.anl.gov

Mihai Anitescu
Argonne National Laboratory
Email: anitescu@mcs.anl.gov

Yiou Li
Illinois Institute of Technology
Email: yli88@stuart.iit.edu

Zhu Wang
Virginia Polytechnic Institute
Email: wangzhu@vt.edu

***Abstract*—We discuss uncertainty analysis of advanced simulation models. At high model complexity, studying the propagation of uncertainty is a serious challenge that cannot be resolved by large-scale sampling alone. For many models, even with improvement in hardware and computing techniques, only small-scale sampling is feasible, and has to be combined with advanced techniques to extract more information from each model run. We have previously introduced a hybrid polynomial regression method that constructs a surrogate uncertainty model using the original model outputs, and also first derivatives of the outputs as fitting conditions. In our ongoing work, we resolve mathematical questions related to the method, including best choice of approximating polynomial basis, and use of dimensionality reduction techniques to create a better representation of uncertainty. Our main application area is nuclear engineering simulation models. Our work also has implications for the tasks of uncertainty quantification, verification and validation, and parametric dependence analysis at extreme computational complexity, in a wide range of applied fields.**

I. Introduction

Many fields of modern science and technology require the use of advanced simulation models. Improvements in hardware and computing techniques allow models of high fidelity, high geometric resolution, with a large number of effects taken into account by the simulation. The task of uncertainty analysis consists of relating the available information on uncertainties (including measurement errors, discretizations and simplifications in the model setup) to the resulting variation in the outputs of the model. The usual difficulties in performing the analysis include the lack of convenient, low-dimensional representation of uncertainty, nonlinearity in the dependency of the model on parameters, high complexity of the model code (so it cannot be rewritten for analysis purposes, or even read completely by a specialist in feasible time), and high computational cost of each model evaluation.

In our work, we mainly address the challenge that results from high dimension of the uncertainty space, and high computational cost of each model evaluation. When the dimension is high, and it is not known which uncertainty states are representative, comprehensive sampling is not possible. For such cases, we have suggested that the effect of uncertainty can be approximated by Polynomial Regression with Derivative Information (PRD). In our main publication [1] we introduce the basic form of the approach, and show that it offers substantial advantage (in precision, computational efficiency, or both) over classical methods of uncertainty analysis: pure random sampling, linear approximation, and polynomial regression that does not use derivative information. In subsequent work, we raised additional questions, such as: selection of a better polynomial basis for approximation, use of dimensionality reduction to create an efficient representation of uncertainty in very large models, and the technical task of applying automatic differentiation to complex models.

In the following sections, we overview the approach, describe out typical applied cases, present recent results on selection of the polynomial basis, and on reducing the dimension of the uncertainty space, and comment on the long-term implications of our research effort.

II. Polynomial Regression with Derivative Information

Consider a generic model as a system of differential-algebraic equations:

$$\begin{gathered} F(T,R)=0 \\ R=R(T)\cdot(1+\Delta R(T,x)) \\ J=J(T), \end{gathered} \tag{1}$$

where the variables $T = (T_1, T_2, ..., T_n)$ characterize the model state; the intermediate (physical) parameters $R = (R_1, R_2, ..., R_N)$ include uncertainty-induced error terms $\Delta R = (\Delta R_1, \Delta R_2, ..., \Delta R_N)$; an output of interest is expressed by the merit function $J(T)$; uncertainty in the model is described by a set of stochastic variables $x = (x_1, x_2, ..., x_d)$. The structure under which uncertainty is introduced can be as simple as $\Delta R(T, x) = x$, or more complex, depending on the modelling principles.

We *redefine* the output as a function of uncertainty quantifiers: $J(T) := J(x)$ and choose a set $\{\Psi_l\}$ of multivariate polynomials to approximate the merit function:

$$J \approx \widetilde{J} = \sum_i a_i \Psi_i(x). \tag{2}$$

For high validity of the surrogate model $\widetilde{J}$, the basis $\{\Psi_l\}$ should include polynomials of high order, but that does not necessarily mean that the basis size grows uncontrollably with increase in the uncertainty space dimension. If the variables

x can be ranked by importance (for example, using first-order sensitivity information), it is possible to construct an *incomplete basis*, with high-order polynomials in important variables only. This realization that a *truncated* (or *incomplete*) basis can be used is an important step in making the approach effective [1].

The coefficients a_l are obtained by requiring that the function and the derivative values of the surrogate model $\widetilde{J}(x)$ match the ones of the real model $J(x)$, in a least-squares sense. Approximation of the uncertain effects by a flexible basis of functions on stochastic variables is essentially Stochastic Finite Element method (SFEM; such a basis is sometimes called *polynomial chaos* [2]); we extend the idea by using derivatives as additional fitting conditions. The polynomial regression equations are:

$$\begin{pmatrix} \Psi_1(S_1) & \Psi_2(S_1) & \cdots & \Psi_k(S_1) \\ \vdots & \vdots & \vdots & \vdots \\ \Psi_1(S_m) & \Psi_2(S_m) & \cdots & \Psi_k(S_m) \\ \frac{\partial \Psi_1(S_1)}{\partial \mathbf{x}_1} & \frac{\partial \Psi_2(S_1)}{\partial \mathbf{x}_1} & \cdots & \frac{\partial \Psi_k(S_1)}{\partial \mathbf{x}_1} \\ \frac{\partial \Psi_1(S_1)}{\partial \mathbf{x}_2} & \frac{\partial \Psi_2(S_1)}{\partial \mathbf{x}_2} & \cdots & \frac{\partial \Psi_k(S_1)}{\partial \mathbf{x}_2} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \Psi_1(S_1)}{\partial \mathbf{x}_d} & \frac{\partial \Psi_2(S_1)}{\partial \mathbf{x}_d} & \cdots & \frac{\partial \Psi_k(S_1)}{\partial \mathbf{x}_d} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \Psi_1(S_2)}{\partial \mathbf{x}_1} & \frac{\partial \Psi_2(S_2)}{\partial \mathbf{x}_1} & \cdots & \frac{\partial \Psi_k(S_2)}{\partial \mathbf{x}_1} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \Psi_1(S_m)}{\partial \mathbf{x}_d} & \frac{\partial \Psi_2(S_m)}{\partial \mathbf{x}_d} & \cdots & \frac{\partial \Psi_k(S_m)}{\partial \mathbf{x}_d} \end{pmatrix} \cdot a = \begin{pmatrix} J(S_1) \\ \vdots \\ J(S_m) \\ \frac{\partial J(S_1)}{\partial \mathbf{x}_1} \\ \frac{\partial J(S_1)}{\partial \mathbf{x}_2} \\ \vdots \\ \frac{\partial J(S_1)}{\partial \mathbf{x}_d} \\ \frac{\partial J(S_2)}{\partial \mathbf{x}_1} \\ \vdots \\ \frac{\partial J(S_m)}{\partial \mathbf{x}_d} \end{pmatrix}, \tag{3}$$

where $S_1, S_2, ...S_d$ are sample training points in the uncertainty space: $S_i = (x_1^{(i)}, x_2^{(i)}, ..., x_m^{(i)})$. A single right-side entry $J(S_i)$ generates a subcolumn of entries $(\frac{\partial J(S_i)}{\partial x_j})$, providing right-side information for several rows at once.

We justify our use of derivative information (as opposed to adding more sample points) by the fact that it is possible to obtain complete gradient information of the model with a limited computational overhead, independent of the model complexity. A computability theory result [3] puts this overhead at 500%, making it advantageous to use PRD for models with uncertainty dimension higher than 5.

Differentiation of the model, as a practical task, depends on the available development budget, i.e. on how much in-depth code analysis one can afford to perform. In the cases where every part of the model is well-documented, and only the large dimension of the input space is a challenge, we used a simple chain-rule approach based on augmentation of the code with partial derivatives of every mathematical procedure. In effect, together with evaluation of the mathematical model $F(T, R(T, x)) = 0$, we also solved the equations

$$(\frac{\partial F}{\partial T} + \frac{\partial F}{\partial R} \cdot \frac{\partial R}{\partial T}) \cdot \frac{dT}{dx} + \frac{\partial F}{\partial R} \cdot \frac{\partial R}{\partial x} = 0 \tag{4}$$

to find $\frac{dT}{dx}$ used in evaluation of $\frac{dJ(T)}{dx} = \frac{\partial J}{\partial T} \cdot \frac{dT}{dx}$. The augmented model can then output both the state and the full gradient in the same run, at acceptable computational overhead of 100 to 200%.

For a wider class of models, with sparsely documented, complex code, and high dimension of model state, a preferred approach is the Automatic (or algorithmic) Differentiation (AD), based on the same chain-rule (4), but constructed automatically, with minimal human involvement. In an ideal situation, the only required processing is to identify inputs and outputs of interest. In practice, some compilation errors, usually related to memory management, have to be resolved for the code processed by automatic differentiation tools.

In our work, the mathematical structure of the model did not have to be modified. In cases of inherent non-differentiability of a mathematical operation, or incomplete convergence of an iterative procedure, we suggest constructing (and differentiating) a smoothing interpolation of such problematic code portions. This implies automatic detection of non-differentiable code, a capacity that some AD tools already possess.

The technical task of differentiating models of industrial complexity is a subject of on-going collaboration with AD development groups. We are currently working on uncertainty analysis of a much more complex SAS4A/SASSYS-1 (over 100,000 lines of code; uncertainty structure can thousands of variables). To our knowledge, successful automatic differentiation and construction of high-validity representation of uncertainty has not been performed at this complexity level, although the code by itself requires modest computing resources. Computation at extreme scale can potentially present additional difficulties for automatic differentiation. For additional details on applying AD to complex (nuclear engineering) simulation models, we refer to [4,5].

We note that almost every aspect of PRD can be improved. The two primary issues are differentiation of the model (to obtain derivatives with respect to at least some variables), and choice of the polynomial basis (to make sure the regression procedure is not badly conditioned). Other questions include optimal selection of the training set, reduction in complexity and dimension of the polynomial structure, and a posteriori estimation of quality of polynomial approximation.

III. Uncertainty Analysis: Applied Cases

Our main application area is the currently expanding field of nuclear engineering, where physical experimentation has always been very expensive, and sophisticated simulation codes are used to design mechanisms of control, improve efficiency and safety of engineering projects. While many industrial simulation codes do not require high performance computing resources, comprehensive description of uncertainty almost always implies dimension on the order of thousands, or more. In addition, the current change of computational and engineering paradigm ("nuclear renaissance", [6]) will lead to the use of high-validity multiphysics codes with high associated computational cost, and even higher cost of error analysis, verification and validation, and uncertainty quantification.

Model I. For basic testing purposes, we use a simplified 3-dimensional steady state model of a nuclear reactor core. The

requirements for this prototype code were to include sufficient complexity to exhibit realistic structure geometry and temperature distribution, and avoid design-specific complexities of nuclear reactor analysis. Currently, the modelled effects include simple heat transport (diffusion and convection) in the core with an arrangement of a small number of fuel elements and freely flowing coolant. The model can also take into account neutron interactions (resulting in non-uniform heat generation in the fuel elements), and changes in the chemical composition of the fuel (fuel depletion). The output of interest is the maximal temperature in the core.

The uncertainty in the model results from the experimental errors in measurement of dependency of intermediate parameters (such as heat conductivity, heat capacity, neutronic macroscopic cross-sections) on temperature; the uncertainty quantifiers are dimensionless coefficients in the expressions $\Delta R(T, x)$. Depending on what effects are taken into account, the effective dimension of the uncertainty space is 12 - 66, larger if more fuel elements are added.

In our numerical experiments on Model I, PRD (with incomplete basis, constructed on a training set of just 10 points) shows an improvement in quality over linear approximation by an order of magnitude (maximal relative error of 0.5% instead of 10%. Other approaches required 100 or more samples for the same precision.

Model II. To provide perspective, we also applied PRD to a professional complexity code MATWS, a combination of the neutron kinetics code from a larger SAS4A/SASSYS code, and a representation of heat removal system. MATWS was used, in combination with a simulation tool Goldsim, to model a number of nuclear reactor accident scenarios [7].

The uncertain quantities in the model are the reactivity feedback coefficients, the dimension is 4 - 10. Since most accident scenarios involve overheating, an output of interest is the maximal temperature (in the fuel, coolant, cladding, etc).

For model II, we constructed a surrogate model using 50 evaluations with derivative information. We then sampled the (explicit) surrogate $\widetilde{J}$, and reproduced statistical results of an accident scenario analysis [7] that required 1000 full model runs. In Figure 1, we show the cumulative distribution of the maximal temperature in the loss-of-coolant flow scenario; note that the predicted distribution almost perfectly falls into the 95% confidence interval of the more expensive analysis.

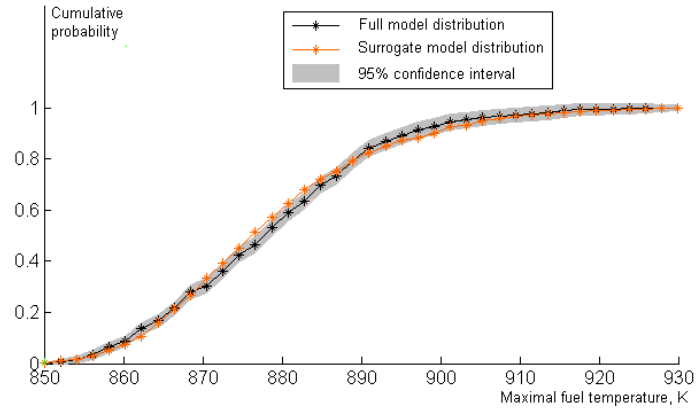

Fig. 1. MATWS: cumulative distribution of maximal terminal heat, ULOF scenario

IV. Polynomial basis selection

The choice of polynomial basis for approximation is an important feature of PRD setup. We inherited the use of multivariate Hermite polynomial basis from SFEM, where it was convenient for statistical reasons (orthogonality in Gaussian probability measure). With the use of derivative information in (3), such standard choices as Hermite or Legendre polynomials are no longer effective. In particular, they lead to poor numerical conditioning in the regression. We have now shown that for best conditioning, the elements of the basis have to satisfy the orthogonality condition

$$\int_{\mathbf{\Omega}} \left(\Psi_j(\mathbf{x})\Psi_h(\mathbf{x}) + \sum_{i=1}^{d} \frac{\partial \Psi_j}{\partial x_i}(\mathbf{x}) \frac{\partial \Psi_h}{\partial x_i}(\mathbf{x}) \right) d\mathbf{x} = \delta_{jh} \quad (5)$$

sufficient to define a basis using Gram-Schmidt orthogonalization.

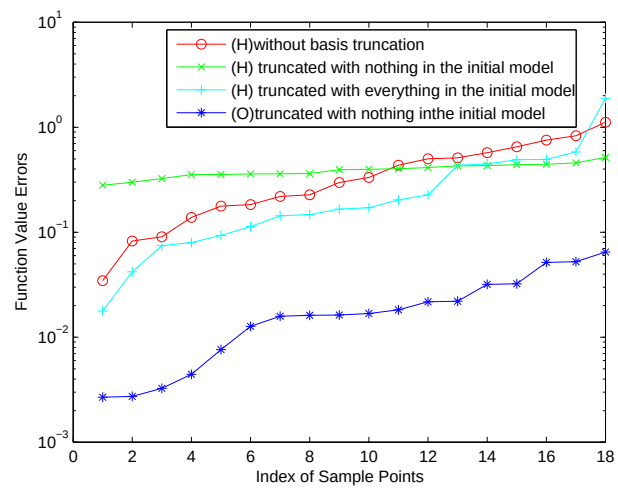

Fig. 2. Sorted error of PRD model: performance of the Hermite multivariate basis (H) compared with adaptive basis selection using Orthogonal basis (O); log_{10} plot

We tested the performance of the orthogonal basis (5) on Model I. We observed a moderate improvement in numerical conditioning of the regression matrix. The real advantage of the orthogonal basis, however, becomes apparent in the performance in combination with adaptive selection of the incomplete (truncated) basis.

In our tests, we used a stepwise fitting procedure in which polynomials are rejected from the expansion if their coefficients are consistently close to zero, or added to the expansion, if they receive a non-negligible coefficient coefficient. The process can go in two directions: either starting with a full

basis of fixed order and reducing the number of polynomials, or starting with nothing and adding polynomials (of at most a fixed order).

A comparison of complete versus truncated polynomial basis, and of Hermite versus orthogonal polynomials on 18 sample points is shown in Figure 2. A combination of adaptive basis construction and correct orthogonality condition results in a more regular representation, with a significant improvement in quality (maximal relative error of 0.2% instead of 1.1%). We hope to provide more detail in our upcoming publication.

V. Dimensionality reduction

Our previous work shows that for the dimension of uncertainty space on the order of $10 - 100$ the use of derivative information allows to construct effective approximations of the uncertainty effect. At the high end of the range, we observe poor numerical conditioning of the regression matrix due to its size, and overall inefficiency due to the large required training set. This prompts our interest in dimensionality reduction of the uncertainty space. Note that the reduction is not applied to the model itself: the model state remains hidden (only participating in differentiation procedures).

For dimensionality reduction, we use two variations of an approach based on Principal Component Analysis (PCA).

For a large dimension of the input variable x and an approximation $\widetilde{J}$ in the basis $\{\Psi\}$, we seek a reducing projection $y = \Phi x$ leading to an approximation

$$J \approx \widetilde{J}(\Phi x) = \sum_i b_i \Psi_i(y). \tag{6}$$

The reduction, defined by Φ, is applied to the training set of inputs ($X = (S_i)$), which does not contain information about the model, and may not have any exploitable structure to allow reduction. Interaction with the model can be established via assessing the influence of input set components on the output.

Approach I is based on dual-weighted variation of PCA, discussed in [8]. In regular PCA, the projection Φ is chosen as dominant eigenspace of correlation matrix $C = X \cdot X^T$. To assign an importance measure w_i to each input vector x_i, and importance measure λ_j to each input variable $(x_i)_j$), we define Φ as dominant eigenspace of dual-weighted matrix

$$C = diag(\lambda_j) \cdot X \cdot diag(w_j) \cdot X^T. \tag{7}$$

We define the measures empirically: the most effective definition is based on sensitivity properties of the output J

Approach II is based on a form of latent factor extraction. Suppose there exists a function f on a few variables, such that

$$J(x) = f(\beta_1 x, \beta_2 x, ..., \beta_k x). \tag{8}$$

Differentiation of (8) shows

$$span(\nabla J) \approx span(\beta_i) \tag{9}$$

so Φ can be chosen as dominant eigenvector space on the matrix that contains *gradients* of the output function at the points in the training set. We note that Approach II is empirically justified, it depends on whether $span(\nabla J)$ can be sampled efficiently: in very high dimension, this may be non-trivial.

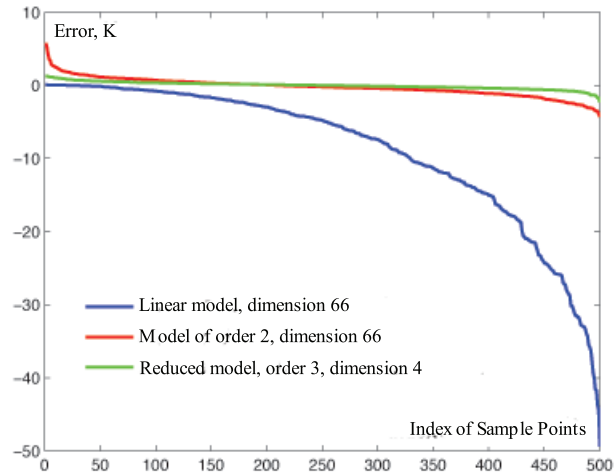

Fig. 3. Sorted error of PRD model: linear model and model of order 2 on full uncertainty space, dimension 66; model of order 3 on a reduced representation of uncertainty space, dimension 4.

We tested the model reduction approaches on Model I, with uncertainty space dimension set to 66. In Figure 3, we compare the performance of a high-order approximation on fewer uncertainty quantifiers with a lower-order surrogate model on the full uncertainty space; reduction to dimension 4 is shown, the reduced model uses full basis of order 3.

In other experiments, Approach II showed a moderate improvement over Approach I.

VI. Conclusion

We view the ongoing work as a successful part of a wider research effort. We believe that it is possible to automatically obtain sensitivity information, and then efficiently build high-quality approximations of parametric dependence of a wide class of models (the requirements are that access to source code is provided, the model is differentiable, in case of large dimension, at least some variables are not essential; work-around techniques exist if any of the requirements are not satisfied). This fact has has implications for the tasks of verification and validation, as well. For example, in an engineering design process, each model can be augmented with its uncertainty representation, outputs of which can be used in validation of other models, thus enabling multidisciplinary validation as a natural part of the design process rather than an expensive additional project. In the context of verification, we find that computationally cheap approximations not only helps to understand the behavior of a model, but also allows the developer to locate important regions both in the parameter space of the model, and in the execution flow of the code. The advantages of using derivative information in uncertainty analysis grow with the increase in the model complexity and dimension; we believe that the use of such advanced techniques is almost unavoidable at extreme scale.

ACKNOWLEDGMENT

The authors would like to thank Thomas Fanning, Mihai Alexe, Jean Utke, and Paul Hovland for their contributions to collaborative research effort. This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, Contract DE-AC02-06CH11357.

REFERENCES

[1] O. Roderick, M. Anitescu, and P. Fischer. *Polynomial Regression Approaches using Derivative Information for Uncertainty Quantification*, Nuclear Science and Engineering, **164** (2), pp 122-139 (2010).

[2] R. Ghanem and P. Spanos, *The Stochastic Finite Element Method: a Spectral Approach*, Springer, New York (1991).

[3] A. Griewank, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, 19, Frontiers in Applied Mathematics. SIAM, Philadelphia,(2000).

[4] J. Utke, U. Naumann, M. Fagan., N. Tallent, M. Strout, P. Heimbach, C. Hill and C. WUNSCH, *OpenAD/F: a Modular Open-source Tools for Automatic Differentiation of Fortran Codes*, ACM Trans. Math. Softw., **34**, pp 1-36 (2008).

[5] M. Alexe, O. Roderick, J. Utke, M. Anitescu, P. Hovland and T. Fanning, *Automatic Differentiation of Codes in Nuclear Engineering Applications*, Tech. Report ANL/MCS-310 (2010).

[6] W. Nuttal, *Nuclear Renaissance: Technologies and Policies for the Future of Nuclear Power*, CRC Press (2004).

[7] E. Morris and W. Nutt, *Uncertainty Analysis for Unprotected Loss-of-heat-sink, Loss-of-flow, and Trainsient-overpower Events in Sodium-cooled Fast Reactors*, in Proc. Int. Conference on Fast Reactors and Related Fuel Cycles (2009).

[8] D. Daescu and I. Navon, *A Dual-weighted Approach to Order Reduction in 4d-var Data Assimilation*, Monthly Weather Review, **136**, pp 1026-1041, (2008).