SAND2016-9310C

Gradient-Enhanced Polynomial Chaos Methods for Circuit Simulation

Eric R. Keiter[1], Laura P. Swiler[2], and Ian Z. Wilcox[3]

1 Introduction

Sensitivity analysis and uncertainty quantification (UQ) are important capabilities for circuit simulation. Sensitivity analysis allows one to determine the most important parameters governing the response(s) of interest, and uncertainty quantification allows one to understand the probability distribution of the response, given probability distributions on the inputs.

Sampling methods are commonly used to perform UQ. While sampling is an attractive approach for several reasons (e.g the accuracy and computational burden is independent of the number of uncertain parameters, it is repeatable given a particular seed, it is fault tolerant in the sense one can drop failed sample evaluations, and it is easy to understand), sampling suffers from the curse of dimensionality. A large number of samples are required to estimate the output statistics, especially to resolve small tail probabilities. The accuracy of the mean estimate obtained form a set of random samples exhibits $1/\sqrt{(N)}$ convergence, meaning that on average one needs to quadruple the number of sample points N to halve the error. Although many improvements on sampling schemes have been developed to overcome these limitations, such as Latin Hypercube Sampling (LHS) and space-filling designs, the essential limitations of sampling still remain.

One recent interest in the computational simulation community is the use of more "embedded" UQ methods. Specifically, there has been interest in implementing adjoints within expensive computational models, so that the adjoint equations can be solved as part of the equation system governing the physics of the problem. [1–3]. Adjoint calculations generate local sensitivities, that is, the derivatives of a quantity of interest with respect to an input parameter (usually at a particular time or

Electrical Models and Simulation, Sandia National Laboratories, MS 1177, Albuquerque, NM, 87185 USA erkeite@sandia.gov, · Optimization and Uncertainty Quantification, Sandia National Laboratories, MS 1318, Albuquerque, NM, 87185 USA lpswile@sandia.gov, · Component and Systems Analysis, Sandia National Laboratories, MS 1177, Albuquerque, NM, 87185 USA iwilcox@sandia.gov

condition such as a particular voltage level). In this paper, we document the implementation of direct sensitivities for steady-state and transient behavior in the Xyce circuit simulation. We then explore the use of these sensitivities in a UQ method called polynomial chaos expansions. We outline the formulations for this UQ method, and demonstrate the computational savings that can be gained when using local sensitivities from Xyce in the UQ process. Note that in this paper, we focus on formulations for the sensitivities and uncertainty quantification methods that employ gradients. These approaches are broadly applicable. The particular implementations of the approaches and algorithms that we describe are in two software frameworks: Xyce, which is a parallel circuit simulator developed at Sandia National Laboratories [4], and Dakota [5], which is an optimization and UQ toolkit also developed at Sandia. These are both open-source software packages available at https://info.sandia.gov/xyce and https://dakota.sandia.gov, respectively. We emphasize that the algorithms and approaches presented here are general, but the specific implementations we use to demonstrate these approaches are in Xyce and Dakota.

2 Sensitivities

Many UQ techniques can be enhanced if the application code is able to produce parameter sensitivities with respect to objective functions of interest. of interest. The Xyce circuit simulator now has steady-state and transient sensitivity capabilities. In this paper, we provide a high-level overview of how the sensitivities are calculated in Xyce. For a more detailed description and derivation of the direct equations, the reader is encouraged to look at [6]. Our primary use case has been using the sensitivity capability in Xyce (either steady-state or transient, depending on the problem), with Xyce passing the sensitivities (the gradients of the objective function with respect to parameters) to Dakota for use in an uncertainty quantification method.

Typically sensitivities are computed with respect to an output of interest. For example:

$$\frac{dO}{dp} = \frac{\partial O}{\partial x}\left(\frac{\partial F}{\partial x}\right)^{-1}\frac{\partial F}{\partial p} + \frac{\partial O}{\partial p} \tag{1}$$

Where O is the scalar objective function, p is a scalar parameter, F is the residual equation and x is the solution vector. dF/dx is the Jacobian matrix. A typical objective function, O, could be something like a circuit output current, or possibly a comparison of that current to measured data for purposes of calibration. The parameter, p, is a compact model parameter such as saturation current.

Sensitivities can be computed using two different methods; the direct method and the adjoint method. These methods are mathematically equivalent, and the best method for any given problem is a matter of computational efficiency. The difference between direct and adjoint is related to the order in which the terms of equation 1 are computed [3, 7].

The general form for direct sensitivities is given by equation 2, in which square brackets have been added to equation 1 to indicate that dx/dp is determined first by way of a linear matrix solve of the Jacobian matrix.

$$\frac{dO}{dp} = \frac{\partial O}{\partial x} \cdot \left[\left(\frac{\partial F}{\partial x} \right)^{-1} \frac{\partial F}{\partial p} \right] + \frac{\partial O}{\partial p} \tag{2}$$

$$\frac{dO}{dp} = \frac{\partial O}{\partial x} \cdot \frac{\partial x}{\partial p} + \frac{\partial O}{\partial p} \tag{3}$$

Direct transient sensitivities are described in section 3.

3 Transient Direct Sensitivities

Transient direct sensitivities can be derived following the approach described by Hocevar. [10]. In transient, the derivation is slightly more complicated than steady-state, as the time derivative term, $\dot{q}$, must be accounted for. To be consistent with the original DAE solve, it is necessary to derive forward direct formulas for all the time integration methods of interest. For the sake or brevity, only a derivation for Backward Euler will be given here, but a similar approach can be taken for other integration methods such as higher-order Gear and Trapezoid. Detailed derivations of the transient direct equations for second-order Gear and Trapezoid are given in reference [6].

For any integration method, a transient direct equation can be solved by starting with the differential algebcaic equation (DAE) form, which is given by:

$$F = \dot{q} + j - b = 0 \tag{4}$$

This equation is minimized at every time step in transient using a Newton solver. To obtain the direct sensitivity equation, equation 4 must be differentiated with respect to a parameter, p, and then re-arranged to give a linear system to be solved at each time step:

$$\frac{dF}{dp} = \frac{d}{dp}(\dot{q} + j - b) = 0 \tag{5}$$

$$\frac{dF}{dp} = \frac{\partial \dot{q}}{\partial p} + \left[\frac{\partial j}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial j}{\partial p} \right] - \frac{\partial b}{\partial p} \tag{6}$$

3.1 Backward Euler Derivation

For Backward Euler integration, the general time derivative form is given by:

$$\frac{\partial \dot{q}}{\partial p} = \frac{\alpha}{h}\left(\frac{\partial q}{\partial p}(x_n) - \frac{\partial q}{\partial p}(x_{n-1})\right) - \beta \frac{\partial \dot{q}}{\partial p}(x_{n-1}) \tag{7}$$

If Backward Euler, then $\alpha = 1$ and $\beta = 0$, giving:

$$\frac{\partial \dot{q}}{\partial p} = \frac{1}{h}\left(\frac{\partial q}{\partial p}(x_n) - \frac{\partial q}{\partial p}(x_{n-1})\right) \tag{8}$$

Using Backward Euler, substitute in the equation for dq/dp:

$$\frac{\partial q}{\partial p} = \frac{\partial q}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial q}{\partial p} \tag{9}$$

$$\begin{aligned}\frac{dF}{dp} =& \frac{1}{h}\left(\left[\frac{\partial q}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial q}{\partial p}\right]_n - \left[\frac{\partial q}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial q}{\partial p}\right]_{n-1}\right) \\ &+ \left[\frac{\partial j}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial j}{\partial p}\right] - \frac{\partial b}{\partial p} = 0\end{aligned} \tag{10}$$

Rearrange to put all the $\frac{\partial x}{\partial p}_n$ terms on the left hand side, and everything else on the right:

$$J\frac{\partial x}{\partial p}_n = -FD + CR \tag{11}$$

Where J is the original Jacobian given by:

$$J = \left[\frac{1}{h}\frac{\partial q}{\partial x} + \frac{\partial j}{\partial x}\right] \tag{12}$$

FD is the "function derivative", or the partial derivatives that come directly from the device models, and is given by:

$$FD = \frac{1}{h}\left[\frac{\partial q}{\partial p}_n - \frac{\partial q}{\partial p}_{n-1}\right] + \frac{\partial j}{\partial p} - \frac{\partial b}{\partial p} \tag{13}$$

The remaining term, CR, in this paper is referred to as the chain-rule term, given by:

$$CR = \frac{1}{h}\left[\frac{\partial q}{\partial x}\right]\frac{\partial x}{\partial p}_{n-1} \tag{14}$$

Note that the chain rule term is using the Q-matrix ($\frac{\partial q}{\partial x}$) from the previous time step, $n-1$. So, for the implementation to be correct, it is necessary to either store store previous matrix-vector muliplication results of $\frac{\partial q}{\partial x}\frac{\partial x}{\partial p}$.

4 Polynomial Chaos Expansion Methods

Stochastic expansion UQ methods approximate the functional dependence of the simulation response on uncertain model parameters by expansion in a polynomial basis. The polynomials used are tailored to the characterization of the uncertain variables. Polynomial chaos expansion is based on a multidimensional orthogonal polynomial approximation.

One advantage of PCE methods is their convergence rate. For smooth functions (i.e., analytic, infinitely-differentiable) in ℓ_2 (i.e., possessing finite variance), exponential convergence rates can be obtained under order refinement for integrated statistical quantities of interest such as mean, variance, and probability. Another advantage of stochastic expansion methods is that the moments of the expansion (e.g. mean or variance of the response) can be written analytically, along with analytic formulations of the derivatives of these moments with respect to the uncertain variables. This property can be exploited in design optimization under uncertainty or epistemic uncertainty problems [16]. A disadvantage of polynomial chaos, as for all global approximation-based methods, is that they may not scale well to high dimensions. Recent research in adaptive refinement and sparse recovery methods strives to address this limitation.

Variance-based decomposition, which explains how output variance relates to the variance of each input variable, may also be calculated analytically from a stochastic expansion. This is a powerful capability for sensitivity analysis, where influential input parameters can be identified and rank ordered. In particular, Dakota can generate Sobol' indices for main, interaction, and total effects. A larger value of the sensitivity index, S_i, means that the uncertainty in the input variable i has a larger effect on the variance of the output. Analytic dependence on expansion coefficients makes computing Sobol' indices essentially free. In contrast, estimating Sobol' indices with LHS can be extremely expensive, since repeat multi-dimensional integrations must be performed.

In PCE, the output response is modeled as a function of the input random variables using a carefully chosen set of polynomials. For example, PCE employs Hermite polynomials to model Gaussian random variables, as originally employed by Wiener [17]. Dakota implements the generalized PCE approach using the Wiener-Askey scheme [18], in which Hermite, Legendre, Laguerre, Jacobi, and generalized Laguerre orthogonal polynomials are used for modeling the effect of continuous random variables described by normal, uniform, exponential, beta, and gamma probability distributions, respectively. These orthogonal polynomial selections are optimal for these distribution types since the inner product weighting function corresponds to the probability density functions for these continuous distributions.

To propagate input uncertainty through a model using PCE, Dakota performs the following steps: (1) input uncertainties are transformed to a set of uncorrelated random variables, (2) a basis such as Hermite polynomials is selected, and (3) the parameters of the functional approximation are determined. The general polynomial chaos expansion for a response g has the form

$$g(\boldsymbol{x}) \approx \sum_{j=0}^{P} \alpha_j \Psi_j(\boldsymbol{x}) \tag{15}$$

where each multivariate basis polynomial $\Psi_j(\boldsymbol{x})$ involves products of univariate polynomials that are tailored to the individual random variables. If a total-order polynomial basis is used (e.g. a total order of 2 would involve terms whose exponents are less than or equal to 2, such as ${x_1}^2$, ${x_2}^2$, and $x_1 x_2$ but not ${x_1}^2 {x_2}^2$), the total number of terms N in a polynomial chaos expansion of arbitrary order p for a response function involving n uncertain input variables is given by: $(n+p)!/n!p!$. If on the other hand, an isotropic tensor product expansion is used with order p in each dimension, the number of terms is $(p+1)^n$. If the order p of the expansion captures the behavior of the true function, polynomial chaos methods will give very accurate results for the output statistics of the response.

In non-intrusive PCE, as in Dakota, simulations are used as black boxes and the calculation of the expansion coefficients α_j for response metrics of interest is based on a set of simulation response evaluations. To calculate these response PCE coefficients, two primary classes of approaches are used: spectral projection and regression. The spectral projection approach projects the response against each basis function $\Psi_j(\boldsymbol{x})$ using inner products and employs the polynomial orthogonality properties to extract each coefficient. Each inner product involves a multidimensional integral over the support range of the weighting function, which can be evaluated numerically using sampling, tensor-product quadrature, Smolyak sparse grid [19], or cubature [20] approaches.

In this work, we use regression-based PCE. Regression-based PCE approaches solve the linear system:

$$\boldsymbol{\Psi \alpha} = \boldsymbol{R} \tag{16}$$

for a set of PCE coefficients $\boldsymbol{\alpha}$ that best reproduce a set of response values $\boldsymbol{R}$. The regression approach finds a set of PCE coefficients α_j which best match a set of response values obtained from either a design of computer experiments ("point collocation" [21]) or from sub-sampling a set of tensor Gauss points ("probabilistic collocation" [22]). The set of response values can be defined on an unstructured grid obtained from sampling within the density function of $\boldsymbol{\xi}$ (point collocation [21,23]) or on a structured grid defined from uniform random sampling on the multi-index[1] of a tensor-product quadrature grid (probabilistic collocation [22]), where the quadrature is of sufficient order to avoid sampling at roots of the basis polynomials[2]. In either case, each row of the matrix $\boldsymbol{\Psi}$ contains the N_t multivariate polynomial terms Ψ_j evaluated at a particular $\boldsymbol{\xi}$ sample.

Additional regression equations can be obtained through the use of derivative information (gradients and Hessians) from each collocation point (refer to `use_derivatives` in the PCE regression specification details in the Dakota Reference Manual [24]), which can aid in scaling with respect to the number of random

[1] Due to the discrete nature of index sampling, we enforce unique index samples by sorting and resampling as required.

[2] Generally speaking, dimension quadrature order m_i greater than dimension expansion order p_i.

variables, particularly for adjoint-based derivative approaches. The derivative equations are added to the set of regression equations as follows:

$$\frac{dg(\boldsymbol{x})}{d\boldsymbol{x}} = \sum_{j=0}^{P} \alpha_j \frac{d\Psi_j(\boldsymbol{x})}{d\boldsymbol{x}} \tag{17}$$

Various methods can be employed to solve (16). The relative accuracy of each method is problem dependent. Traditionally, the most frequently used method has been least squares regression. However when $\boldsymbol{\Psi}$ is under-determined, minimizing the residual with respect to the ℓ_2 norm typically produces poor solutions. Compressed sensing methods have been successfully used to address this limitation [25, 26]. Such methods attempt to only identify the elements of the coefficient vector $\boldsymbol{\alpha}$ with the largest magnitude and enforce as many elements as possible to be zero. Such solutions are often called sparse solutions. Dakota provides algorithms that solve the following formulations: orthogonal matching pursuit, least angle regression (LARS), least absolute shrinkage (LASSO), basis pursuit, basis pursuit denoising, and a standard least squares. Typically, we recommend using least squares for over-determined systems and compressed sensing methods for under-determined systems, which is the case when the basis functions are augmented with additional basis functions representing gradient terms. Details of these methods are documented in the Linear Regression section of the Dakota Theory Manual [27].

5 Results for CMOS Inverter Circuit

In this section, we demonstrate the use of the gradient-enhanced UQ methods on a five-stage CMOS inverter which involves transient sensitivities calculated by Xyce. We model a simple CMOS five-stage inverter circuit, which uses 10 instances of the BSIM6 [28] compact model. This circuit is meant to mimic applications where signal delay is the important metric. Each inverter stage adds to the signal delay. The CMOS circuit is shown in Figure 1. The PMOS and NMOS oxide thicknesses are thus critical uncertain parameters. We model these as normal uncertainties, centered around a nominal value with a standard deviation equal to 10% of nominal.

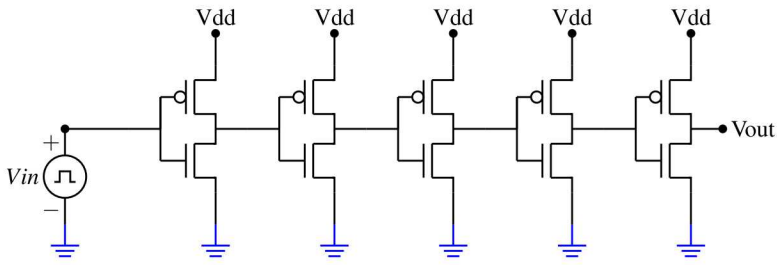

Fig. 1 CMOS circuit with five inverters.

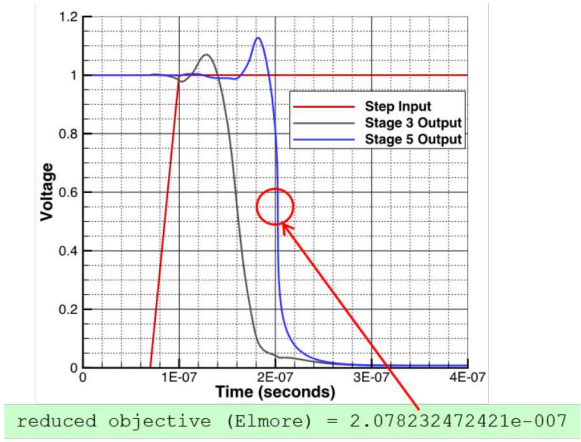

Fig. 2 Overall behavior of CMOS circuit with Elmore delay highlighted

The circuit is driven by a step input, and the output of interest is the output voltage V_{out}. This is shown in Figure 2.

Since the output voltage is a transient signal, we used a generalized Elmore delay, similar to that given by [2], as our objective function of interest. The Elmore delay is given by:

$$O = \texttt{Elmore Delay} = \frac{\int_0^T g'_A(t) \cdot t \cdot dt}{\int_0^T g'_A(t) dt} \tag{18}$$

and represents the approximate time for the signal rise or fall. Note that $g_A(t) = V_{out}$.

Xyce returns the transient sensitivities (e.g. the derivatives of the Elmore delay objective with respect to the thickness of the NMOS and PMOS oxide layers as a function of time). This is shown in Figure 3.

We performed uncertainty quantification on the CMOS circuit using a variety of UQ techniques. As a baseline, we performed Latin Hypercube Sampling (LHS) with 100 and 1000 samples. LHS is a stratified sampling method which has good space-filling properties and generally gives better results than plain Monte-Carlo sampling (e.g. results which have lower variance on statistical estimators such as the mean). Then, we performed polynomial chaos expansion using a full tensor product quadrature of order 5 for each of the two input parameters, resulting in a total of 25 sample points. Finally, we performed two types of regression-based PCE. In the first, we used 30 samples but did not include the gradients. In the second, we used 10 samples. For each sample, we had two gradient values representing the derivative of the Elmore delay with respect to the two input parameters. Thus, the last PCE calculation used 30 pieces of information and was comparable to the 30 sample regression PCE with no gradients, but it only required 10 samples.

Note that for all of these sample runs, the two input parameters were the NMOS and PMOS oxide layer thickness.They were varied according to a normal distributions with means of 1.74E-9m and 2.34E-9m, and standard deviations of 1.75E-10

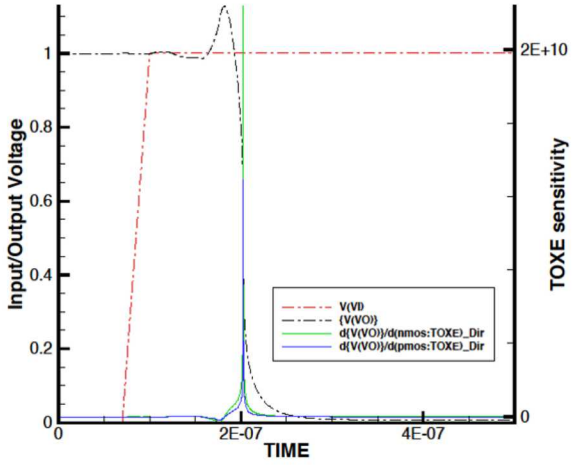

Fig. 3 Transient sensitivities from the CMOS circuit highlighted

and 1.34E-10, respectively. After each Xyce run, the Elmore delay was calculated and used as the quantity of interest in these analyses. All of the runs were performed using Dakota for the UQ methods and Xyce as the circuit simulator.

The use of sensitivities in performing uncertainty analysis is highlighted in Figure 4 and Table 1. As shown in the figure, the cumulative distribution function (CDF), which gives the probability that the Elmore delay is less than a particular value, is almost the same for an LHS sample of size 1000 and all of the PCE methods. It is very hard to see differences: the CDF curves for LHS 1000 and all of the PCE variants overlay each other. The one that is different is LHS based only on 100 samples. Figure 4 shows that this CDF is not as resolved as the others. Table 1 shows that the mean values of the Elmore delay are very similar, differing only in the fifth significant figure. Finally, the standard deviations show a little more variability, but again are reasonably close. We conclude that a polynomial chaos expansion using sensitivities from Xyce (the 10 PCE with regression case) performs comparably to 1000 samples from LHS.

6 Conclusions

This paper explored a new approach to circuit level uncertainty quantification, based on gradient-enhanced Polynomial Chaos Expansions (PCE). PCE is a non-sampling, projection-based technique, in which circuit parametric uncertainties are approximated using an expansion of orthogonal polynomials, and the specific choice of polynomial is determined by the assumed functional form of the uncertain inputs. Regression-based PCE can be enhanced by parametric derivative information from

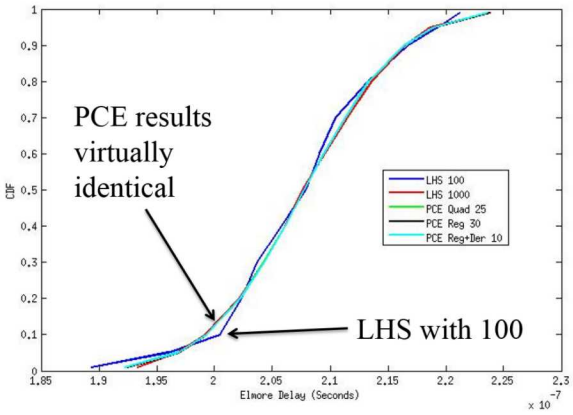

Fig. 4 Cumulative Distribution Function of Elmore Delay using Various UQ Approaches

Table 1 Comparison Results from UQ Approaches

Number of samples and UQ Method	Mean	Std Dev.
100 LHS	2.0781E-7	6.6309E-9
1000 LHS	2.0782E-7	6.6935E-9
25 PCE Quadrature	2.0783E-7	6.6954E-9
30 PCE Regression	2.0783E-7	6.7131E-9
10 PCE Regression with derivatives	2.0782E-7	6.7035E-9

the simulator. Derivative-based enhancements to PCE offers the possibility of similar accuracy for a smaller number of samples. In this paper, the development of direct transient sensitivities in a circuit simulator is described, and the successfull application of these sensitivities to gradient-enhanced PCE has been demonstrated.

Acknowledgements This work was sponsored by the Laboratory Directed Research and Development (LDRD) Program at Sandia National Laboratories. Sandia National Laboratories is a multiprogram laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

1. R. A. Bartlett, "A derivation of forward and adjoint sensitivities for odes and daes," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2007-6699, June 2009.
2. A. Meir and J. Roychowdhury, "Blast: Efficient computation of nonlinear delay sensitivities in electronic and biological networks using barycentric lagrange enabled transient adjoint analysis," in *DAC '12: Proceedings of the 2012 Design Automation Conference*. New York, NY, USA: ACM, 2012.
3. F. Liu and P. Feldmann, "A time-unrolling method to compute sensitivity of dynamic systems," in *Design Automation Conference (DAC), 2014 51st ACM/EDAC/IEEE*, June 2014, pp. 1–6.
4. E. R. Keiter, T. Mei, T. V. Russo, R. L. Schiek, P. E. Sholander, H. K. Thornquist, J. C. Verley, and D. G. Baur, "Xyce Parallel Electronic Simulator: User's Guide, Version 6.4," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2015-10527, December 2015.
5. B. M. Adams, L. E. Bauman, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, P. D. Hough, K. T. Hu, J. D. Jakeman, L. P. Swiler, S. J. A., D. M. Vigil, and T. M. Wildey, "Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.0 users manual," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2014-4633, Updated May 2014, available online from http://dakota.sandia.gov/documentation.html.
6. E. R. Keiter and L. P. Swiler, "Advanced uncertainty quantifiation methods for circuit simulation: Interim report LDRD 2014-0788," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2014-20238, December 2014.
7. B. Gu, K. Gullapalli, Y. Zhang, and S. Sundareswaran, "Faster statistical cell characterization using adjoint sensitivity analysis," in *Custom Integrated Circuits Conference, 2008. CICC 2008. IEEE*, Sept 2008, pp. 229–232.
8. E. R. Keiter, T. Mei, T. V. Russo, R. L. Schiek, P. E. Sholander, H. K. Thornquist, J. C. Verley, and D. G. Baur, "Xyce Parallel Electronic Simulator: User's Guide, Version 6.1," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2014-2405, March 2014.
9. ——, "Xyce Parallel Electronic Simulator: Reference Guide, Version 6.1," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2014-2406, March 2014.
10. D. E. Hocevar, P. Yang, T. N. Trick, and B. D. Epler, "Transient sensitivity computation for mosfet circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. CAD-4, no. 4, October 1985.
11. Z. Ilievski, H. Xu, A. Verhoeven, E. ter Maten, W. Schilders, and R. Mattheij, "Adjoint transient sensitivity analysis in circuit simulation," in *Scientific Computing in Electrical Engineering*, ser. Mathematics in Industry, G. Ciuprina and D. Ioan, Eds. Springer Berlin Heidelberg, 2007, vol. 11, pp. 183–189.
12. Y. Cao, S. Li, L. Petzold, and R. Serban, "Adjoint sensitivity analysis for differential-algebraic equations: The adjoint dae system and its numerical solution," *SIAM Journal on Scientific Computing*, vol. 24, no. 3, pp. 1076–1089, 2003. [Online]. Available: http://dx.doi.org/10.1137/S1064827501380630
13. S. Li and L. Petzold, "Adjoint sensitivity analysis for time-dependent partial differential equations with adaptive mesh refinement," *Journal of Computational Physics*, vol. 198, no. 1, p. 310325, 2004.
14. R. Mrz, "Differential algebraic systems anew," *Applied Numerical Mathematics*, vol. 42, no. 13, pp. 315 – 335, 2002, numerical Solution of Differential and Differential-Algebraic Equations, 4-9 September 2000, Halle, Germany. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168927401001581
15. A. Haldar and S. Mahadevan, *Probability, Reliability, and Statistical Methods in Engineering Design*. New York: Wiley, 2000.
16. M. S. Eldred, L. P. Swiler, and G. Tang, "Mixed aleatory-epistemic uncertainty quantification with stochastic expansions and optimization-based interval estimation," *Reliability Engineering and System Safety*, vol. 96, no. 9, pp. 1092–1113, 2011.

17. N. Wiener, "The homogoeneous chaos," *Amer. J. Math.*, vol. 60, pp. 897–936, 1938.
18. D. Xiu and G. M. Karniadakis, "The wiener-askey polynomial chaos for stochastic differential equations," *SIAM J. Sci. Comput.*, vol. 24, no. 2, pp. 619–644, 2002.
19. S. Smolyak, "Quadrature and interpolation formulas for tensor products of certain classes of functions," *Dokl. Akad. Nauk SSSR*, vol. 4, pp. 240–243, 1963.
20. A. Stroud, *Approximate Calculation of Multiple Integrals*. Prentice Hall, 1971.
21. R. W. Walters, "Towards stochastic fluid mechanics via polynomial chaos," in *Proceedings of the 41st AIAA Aerospace Sciences Meeting and Exhibit*, no. AIAA-2003-0413, Reno, NV, January 6–9, 2003.
22. M. Tatang, "Direct incorporation of uncertainty in chemical and environmental engineering systems," Ph.D. dissertation, MIT, 1995.
23. S. Hosder, R. W. Walters, and M. Balch, "Efficient sampling for non-intrusive polynomial chaos applications with multiple uncertain input variables," in *Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, no. AIAA-2007-1939, Honolulu, HI, April 23–26, 2007.
24. B. M. Adams, L. E. Bauman, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, P. D. Hough, K. T. Hu, J. D. Jakeman, L. P. Swiler, S. J. A., D. M. Vigil, and T. M. Wildey, "Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.2 reference manual," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2014-5015, Updated May 2015, available online from http://dakota.sandia.gov/documentation.html.
25. G. Blatman and B. Sudret, "Adaptive sparse polynomial chaos expansion based on least angle regression," *Journal of Computational Physics*, vol. 230, no. 6, pp. 2345 – 23:67, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021999110006856
26. A. . Doostan and H. Owhadi, "A non-adapted sparse approximation of PDEs with stochastic inputs," *Journal of Computational Physics*, vol. 230, no. 8, pp. 3015 – 3034, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021999111000106
27. B. M. Adams, L. E. Bauman, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, P. D. Hough, K. T. Hu, J. D. Jakeman, L. P. Swiler, S. J. A., D. M. Vigil, and T. M. Wildey, "Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.2 theory manual," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2014-4253, Updated May 2015, available online from http://dakota.sandia.gov/documentation.html.
28. Y. S. Chauhan, S. Venugopalan, M. A. Chalkiadaki, M. A. U. Karim, H. Agarwal, S. Khandelwal, N. Paydavosi, J. P. Duarte, C. C. Enz, A. M. Niknejad, and C. Hu, "Bsim6: Analog and rf compact model for bulk mosfet," *IEEE Transactions on Electron Devices*, vol. 61, no. 2, pp. 234–244, Feb 2014.
29. E. R. Keiter, T. V. Russo, E. L. Rankin, R. L. Schiek, H. K. Thornquist, D. A. Fixel, T. S. Coffey, R. P. Pawlowski, K. R. Santarelli, and C. E. Warrender, "Xyce parallel electronic simulator: User's guide, version 5.2," Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2011-2515, 2011.
30. B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response," *IEEE Journal of Solid-State Circuits*, vol. 3, no. 4, pp. 365–373, Dec 1968.