Derivative Based Global Sensitivity Analysis using Conjugate Unscented Transforms

Souransu Nandi[1], Tarunraj Singh[2] and Puneet Singla[3]

Abstract— **In this paper, a novel way to compute derivative-based global sensitivity measures is presented. Conjugate Unscented Transform (CUT) is used to evaluate the multidimensional definite integrals which lead to the sensitivity measures. The method is compared with Monte Carlo estimates as well as the screening method of Morris. It is shown that using CUT provides a much more accurate estimate of sensitivity measures as compared to Monte Carlo (with far lesser computational cost) as well as the Morris method (with similar computational cost). Illustrations on three test functions are presented as evidence.**

I. INTRODUCTION

Sensitivity Analysis (SA) is a branch of study in engineering which involves determining quantitatively how much the output of a system varies with its input parameters. This allows one to figure out which input factors significantly impact the output and which factors (if any) are inconsequential. Valuable information can be inferred from these studies and are used when designing experiments, developing surrogate models or even allocating resources to analyze important input variables [1].

SA has been traditionally done in two ways [2]. Local SA which involves calculating the partial derivative of the output with respect to each of its input factors at a nominal point in the input space and global SA where a metric is derived which represents the degree of output variation when all the inputs are varied across their domain. Hence global SA not only looks at the interaction between some of the input factors but also considers the entire input domain as a whole (unlike local SA where the metric is evaluated only at nominal locations).

Global SA (GSA) techniques have been popularly studied in the literature (see articles [1], [2], [3], [4], [5], [6], [7] and references there in). One of the most common indices used for GSA has been the Sobol' Indices [1]. It is a variance based approach where indices pertaining to each input and its combinations are derived based on their fractional contribution to the total output variance. Such indices consider the entire input space as well as the interaction between input factors. However, computing these indices is extremely expensive and could require extensive system realizations.

[1] Souransu Nandi, Graduate Student, Department of Mechanical and Aerospace Engineering, University at Buffalo, Buffalo, NY 14260, USA souransu at buffalo.edu

[2] Tarunraj Singh, Professor, Department of Mechanical and Aerospace Engineering, University at Buffalo, Buffalo, NY 14260, USA tsingh at buffalo.edu

[3] Puneet Singla, Associate Professor, Aerospace Engineering, Pennsylvania State University, University Park, PA 16802, USA pxs433 at engr.psu.edu

In order to navigate the high computational requirements of calculating the Sobol indices, it is often desired to determine a subset of the input variables which can be declared unimportant. If such a subset can be found (i.e. the sensitivity of the output with respect to a subset of inputs is negligible), the dimensionality of the problem can be readily reduced and Sobol' indices could be then determined for a lower-dimensional system.

One such method to eliminate non-crucial inputs has been to use the Morris method of screening [8]. The Morris method crudely approximates the local sensitivity of the output at a certain location in the input space and then averages the local sensitivities evaluated at a finite number of input points. The Morris method has its own set of drawbacks [9]. In order to get a more accurate measure of averaged local sensitivities, Kucherenko et al. in [9] introduced metrics called the Derivative-based Global Sensitivity Measures (DGSM). DGSM basically evaluates the local sensitivity at a particular location and is then averaged over the entire input space using Monte Carlo or quasi Monte Carlo techniques. Their article shows that DGSM is a more reliable way to determine relative importance of input parameters and do so at a comparable computational cost. Sobol and Kucherenko [10] illustrate that the total Sobol indices are bounded by a measure similar to the DGSM which integrate the square of the local sensitivity rather than the absolute value of the local sensitivities. This provides the motivation to use DGSM as a means to pare down and identify the subset of parameters for other higher fidelity global sensitivity analysis.

The work presented in this paper details a method to determine the DGSM even more efficiently than previously quoted. The local sensitivity at any location is determined using finite difference similar to [9] and remains unchanged in this paper as well. However, the methods to find the mean and the standard deviation of those local sensitivities are altered. Conjugate Unscented Transform (CUT) [11] which is a recently developed set of quadrature rules to evaluate multivariate integrals is used to evaluate the necessary mean and variance integrals accurately. It is shown in this paper that CUT greatly reduces the number of function evaluations necessary to derive reliable DGSM metrics.

The paper has been organized in the following way. Section II presents the derivative based global sensitivity measures, section III elaborates the method of Morris, section IV discusses the Monte Carlo and Conjugate Unscented Transform approach towards evaluating the sensitivity measures, section V presents the functions on which our methods

are tested, and finally section VI presents all the results before concluding with section VII.

II. DERIVATIVE-BASED GLOBAL SENSITIVITY ANALYSIS

Consider the equation

$$y = f(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is a bounded input vector which is mapped to a scalar output function y via f. Without loss of generality, it is assumed throughout this paper that the domain of $\boldsymbol{x}$ is an n dimensional hypercube Ω (i.e. $\Omega = [0,1]^n$) and that the i^{th} element of $\boldsymbol{x}$ is written as x_i.

Assuming that the input-output mapping function $f(\boldsymbol{x})$ is differentiable, the local sensitivity at any location in the uncertain space $\boldsymbol{x}$ is given by

$$\boldsymbol{E}(\boldsymbol{x}) = \frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x}) \tag{2}$$

where

$$\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x}) = \left[\frac{\partial f}{\partial x_1}(\boldsymbol{x}), \frac{\partial f}{\partial x_2}(\boldsymbol{x}), \cdots, \frac{\partial f}{\partial x_n}(\boldsymbol{x})\right]^T. \tag{3}$$

Derivative based global sensitivity analysis involves the determination of certain Derivative-based Global Sensitivity Measures (DGSM). One such measure is defined as the average value of the local sensitivities over the domain of the uncertain space Ω [9] and is given by

$$\boldsymbol{M} = \int_\Omega \boldsymbol{E}(\boldsymbol{x}) d\boldsymbol{x}. \tag{4}$$

Another measure often used is the standard deviation of $\boldsymbol{E}$ given by $\boldsymbol{\Sigma}$ where the i^{th} element of $\boldsymbol{\Sigma}$ is

$$\Sigma_i = \sqrt{\int_\Omega (E_i(\boldsymbol{x}) - M_i)^2 d\boldsymbol{x}} \tag{5}$$

where $\boldsymbol{\Sigma} = [\Sigma_1, \Sigma_2, \cdots, \Sigma_n]^T$, M_i and E_i are the i^{th} elements of $\boldsymbol{M}$ and $\boldsymbol{E}$ respectively. However, a major drawback of $\boldsymbol{M}$ is that if the values of $\boldsymbol{E}$ have large variations in magnitude with opposite signs in Ω, the $\boldsymbol{M}$ metric could have a value very close to 0. This would underestimate the true variation of the sensitivities in the domain of interest.

Hence, in order to circumvent the *cancelling effect*, updated measures are considered where the absolute values of local sensitivities are averaged. These metrics are given by

$$\boldsymbol{M^*} = \int_\Omega |\boldsymbol{E}(\boldsymbol{x})| d\boldsymbol{x} \tag{6}$$

and

$$\Sigma_i^* = \sqrt{\int_\Omega (|E_i(\boldsymbol{x})| - M_i^*)^2 d\boldsymbol{x}} \tag{7}$$

where $\boldsymbol{M^*} = [M_1^*, \cdots, M_n^*]^T$ and $\boldsymbol{\Sigma^*} = [\Sigma_1^*, \cdots, \Sigma_n^*]^T$. Similar to [9], metrics $\boldsymbol{M^*}$ and $\boldsymbol{\Sigma^*}$ are considered to be the DGSM.

III. METHOD OF MORRIS

The Morris screening method was first introduced in 1991 [8]. In the same flavour as determining derivatives using finite difference, the method employed observing the changes in the model output when any of the inputs were randomly perturbed while keeping the others constants. This technique has popularly been called as having randomized *one-factor-at-a-time* experiments.

Each individual experiment leads to the determination of an *elementary effect* (d). At the end of all experiments, the statistics of all d are observed to evaluate the relative importance of the input factors.

In this method, the entire input space is uniformly gridded. A grid point $\boldsymbol{x} = [x_1, ..., x_n]^T$ is chosen from that grid at random and the following elementary effect is calculated

$$d_i = \frac{y(x_1, ..., x_i + \Delta, ..., x_n) - y(\boldsymbol{x})}{\Delta} \tag{8}$$

where i represents the elementary effect due to the i^{th} input factor (i.e. i varies between 1 and n). The value of $\boldsymbol{x}$ should be chosen such that the perturbed element $x_i + \Delta$ is within the domain of x_i. The number of levels into which an input factor is divided is usually denoted by p and the total number of samples (or experiments) by N. Δ is typically the relative distance between two grid points in the direction of perturbation. Although Δ could be the distance between any two grid locations, it is typically made equal to the distance between two adjacent grid locations (i.e. $\Delta = \frac{1}{p-1}$). The notation and the nomenclature for the Morris method has been adopted from [1] and can be referred to for more details.

If the elementary effect from the j^{th} experiment or sample is denoted by $d_i^{(j)}$, the final metric of interest is defined to be

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} d_i^{(j)} \tag{9}$$

and

$$\sigma_i = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(d_i^{(j)} - \mu_i)^2}. \tag{10}$$

Once again to circumvent the cancelling effect, a revised version of the metric was presented by Campolongo et al. in [12] and is given by

$$\mu_i^* = \frac{1}{N}\sum_{j=1}^{N}|d_i^{(j)}|. \tag{11}$$

In order to be consistent with the definition in equation (7), we define

$$\sigma_i^* = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(|d_i^{(j)}| - \mu_i^*)^2}. \tag{12}$$

For purposes of brevity we also define the vectors $\boldsymbol{\mu^*} = [\mu_1^*, ..., \mu_n^*]^T$ and $\boldsymbol{\sigma^*} = [\sigma_1^*, ..., \sigma_n^*]^T$. For the rest of the paper, $\boldsymbol{\mu^*}$ and $\boldsymbol{\sigma^*}$ are considered to be the Morris Method (MM) metrics.

IV. COMPUTATIONAL METHODS TO EVALUATE DGSM

The primary goal of this section is to present two methods to evaluate the DGSM numerically. Evaluating the DGSM requires the computation of definite integrals of the form

$$I = \int_{\Omega} g(\boldsymbol{x}) \boldsymbol{dx}. \tag{13}$$

There is a considerable amount of literature that is present which looks at evaluating I. Some of the most popular methods are the Monte Carlo (MC) random sampling [13], Gauss quadrature rules [14], Sparse quadrature [15] and recently developed Conjugate Unscented Transform (CUT) among several others. In this paper, two methods are discussed: namely the MC and the CUT.

A. Monte Carlo Sampling

Monte Carlo sampling has been used extensively for evaluating compact integrals. It involves drawing random samples from the input space, evaluating the integrand at those locations and averaging them, i.e.

$$I_{MC} = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} g(\boldsymbol{x}_i). \tag{14}$$

I_{MC} is an approximation to I and converges on I as N_{MC} tends to infinity. Although the convergence of MC is slow (and is inversely proportional to the square root of N_{MC}), it has the advantage of being independent of the number of variables. Hence, MC has been popular for evaluating high dimension multivariate integrals where quadrature rules encounter the curse of dimensionality [9].

B. Conjugate Unscented Transform

Conjugate Unscented Transform (CUT) like several other numerical integration methods, is a sigma-point based integration scheme where a multivariate integral is estimated by calculating weighted sums of the integrand evaluated at specific points (a.k.a sigma points) inside the input domain. Therefore, I in equation (13) can be approximated as

$$I_{CUT} = \sum_{i=1}^{N_{CUT}} w_i g(\boldsymbol{x_i}). \tag{15}$$

The distinction between different sigma-point based integration schemes lie in the manner by which the sigma points $\boldsymbol{x_i}$ and the weights w_i are selected. For example, in Gauss quadrature, the points and weights for a multidimensional input space are determined simply by taking the tensor product of the points and weights in each univariate direction (where the points and weights for each univariate direction is derived from roots of specific orthogonal polynomials such as Gauss-Legendre polynomials). For Sparse quadrature rules, only a subset of those points are used to determine the multivariate integrals. CUT has developed rules to determine points and weights to evaluate integrals using moment constraint equations [11]. Such a methodology results in far fewer number of sigma points to evaluate multidimensional integrals in comparison to any other known quadrature rules.

CUT was developed with an intention to evaluate primarily weighted integrals where the weighing function would either be a uniform distribution or the standard normal distribution. Since, often in sensitivity analysis, input factors are bounded, CUT rules for uniform distributions can be readily adopted to determine integrals of the form I.

Depending on the order of moments that are satisfied by the sigma points and weights, three sets of sigma points were proposed: namely the CUT4, CUT6 and CUT8 algorithms. The points in CUT4, CUT6 and CUT8, are designed to preserve up to 5^{th}, 7^{th}, and 9^{th} order moments. The number of CUT points in each category is fixed depending on the dimension of the input space. Although with an increase in dimension, the number of points increase, the growth is substantially lower than some of the other popular quadrature schemes. More details regarding the construction of the CUT points, weights and the nomenclature can be found in [11].

Hence, in this paper, we adopt CUT to efficiently determine the mean and standard deviation integrals necessary to evaluate DGSM.

Function	$f(\boldsymbol{x})$
A1	$\sum_{i=1}^{n}(-1)^i \prod_{j=1}^{i} x_j$
B1	$\prod_{i=1}^{n} \frac{n-x_i}{n-0.5}$
C1	$2^n \prod_{i=1}^{n} x_i$

TABLE I
TEST FUNCTIONS

V. TEST FUNCTIONS

The objective of this paper is to compare the metrics of DGSM and MM and present computationally efficient ways to evaluate DGSM over existing known methods. To illustrate the performance of each algorithm, three benchmark test functions are considered.

These functions are listed in Table I. Function A1 is known to have a few dominant input variables. Function B1 has no unimportant subsets and has important low-order interactions. Function C1 also has no unimportant subsets but in contrast to B1 has important high-order interaction terms. The functions have been adopted from [9] and should be referred for more details.

For each of these functions, 5 methods are implemented for effective comparisons.

Method 1: Morris method with $p = 11$ (MM11).

Method 2: Morris method with $p = 101$ (MM101).

Method 3: Monte Carlo estimate of DGSM (MCDGSM).

Method 4: CUT4 estimate of DGSM (CUT4DGSM).

Method 5: Analytical values of DGSM (ADGSM).

For each method, the global sensitivity metrics are evaluated. Method 5 is used as a gold standard for comparison since it is evaluated symbolically (i.e. it is not a numerical estimate).

Computational Efficiency

In order to compare the numerical efficiency, the number of function evaluations (N_f) is used as a metric. It is desired to obtain the relative importance of the input factors x_i with as few function evaluations as possible.

For the Morris methods, each experiment requires $n+1$ function evaluations to compute $\boldsymbol{d}^{(j)}$ where $\boldsymbol{d}^{(j)} = [d_1^{(j)}, d_2^{(j)}, ..., d_n^{(j)}]^T$. If there are a total of N experiments done, $N \times (n+1)$ function calls are required to evaluate the MM metrics.

For the MC and the CUT method, the integrand ($g(\boldsymbol{x})$ in equation (13)) is the local sensitivity $\boldsymbol{E}(\boldsymbol{x})$. In this paper, finite difference (FD) (forward difference specifically) is used to evaluate the local sensitivity at a particular location $\boldsymbol{x}$. Since, each sensitivity evaluation using FD also requires $n+1$ function evaluations, the total number of function calls for methods 3 and 4 is given by $N_{MC/CUT} \times (n+1)$.

VI. RESULTS

This section presents the final simulation results for the 5 methods on the test functions. For all the simulations the number of input factors considered were $n = 6$. Hence, the input domain was a 6 dimensional hypercube (i.e. $\Omega = [0,1]^6$). It should be noted that for $n = 6$, the number of CUT4 points are 252 (i.e. $N_{CUT4} = 252$). To make a fair computational comparison the values of N for methods 1 and 2 are made equal to N_{CUT4}. Since, the convergence of MC is rather slow, N_{MC} is considered to be 100000 for all simulations.

Figures 1(a), 1(b) and 1(c) present the values of $\boldsymbol{\mu}^*$ and $\boldsymbol{M}^*$ determined using the five methods for the test functions A1, B1 and C1 respectively. Similarly, figures 2(a), 2(b) and 2(c) present the corresponding values of $\boldsymbol{\sigma}^*$ and $\boldsymbol{\Sigma}^*$. It is quite evident from these figures that the Morris method repeatedly fails to correctly rank the importance of the input factors x_i especially for functions B1 and C1. Although both functions B1 and C1 have input factors which have an equal influence on the output, the Morris methods MM11 as well as MM101 indicate that some input factors are more important than the others since it gives widely varying estimates for $\boldsymbol{\mu}^*$ and $\boldsymbol{\sigma}^*$. In contrast, we see that the DGSM estimates from CUT4 and MC do much better. For function A1, they both successfully rank the input factors with decreasing influence while for functions B1 and C1, it is correctly identified that all the input factors are of equal relevance.

To present a quantitative comparison, the norm of the error between the analytical results and the other four methods are listed in Table II. The values in the table are calculated using the expression

$$RMSE = ||\boldsymbol{A}_{method} - \boldsymbol{B}_{ADGSM}||_2 \tag{16}$$

where $\boldsymbol{A}_{method}$ refers to the mean and the standard deviation values derived from methods 1 through 4 and $\boldsymbol{B}_{ADGSM}$ refers to the analytical values of $\boldsymbol{M}^*$ and $\boldsymbol{\Sigma}^*$. For each column, the cell with the minimum value has been shaded in gray to highlight it.

Function	A1		B1		C1	
Method	$\boldsymbol{M}^*$ or $\boldsymbol{\mu}^*$	$\boldsymbol{\Sigma}^*$ or $\boldsymbol{\sigma}^*$	$\boldsymbol{M}^*$ or $\boldsymbol{\mu}^*$	$\boldsymbol{\Sigma}^*$ or $\boldsymbol{\sigma}^*$	$\boldsymbol{M}^*$ or $\boldsymbol{\mu}^*$	$\boldsymbol{\Sigma}^*$ or $\boldsymbol{\sigma}^*$
MM11	0.0568	0.0374	0.0205	0.0045	1.6070	1.8844
MM101	0.0211	0.0205	0.0068	0.0019	0.5425	1.3822
MCDGSM	0.0013	0.0014	0.0001	0.0001	0.0218	0.0578
CUT4DGSM	0.0000	0.0011	0.0000	0.0000	0.0000	0.0978

TABLE II

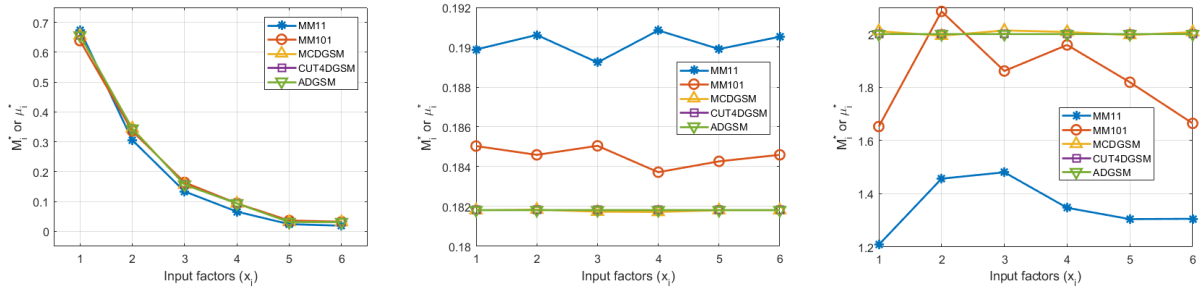
ERROR NORM COMPARISON FOR MEANS AND STANDARD DEVIATIONS OF LOCAL SENSITIVITIES OVER Ω

It is evident from Table II that CUT4DGSM returns values closest to the analytical values of the averaged local sensitivities and their standard deviations. Note that the number of function evaluations required to evaluate each of entries for MM11, MM101 as well as CUT4DGSM are identical and is given by $N_{CUT4/MM} \times (n+1) = 252 \times (6+1) = 1764$. The number of function evaluations for MCDGSM were $N_{MC} \times (n+1) = 100000 \times 7 = 700000$. Hence we see that with a computational effort similar to that of the traditional Morris method, CUT4DGSM can perform far more accurately (with improvements in several orders of magnitude).

We see that CUT4DGSM does much better than all the other methods consistently for all the test functions except for the standard deviation column of C1. As previously mentioned, the output of C1 has important high-order interactions among its input factors. Such high order interactions are difficult to observe and capture accurately in sensitivity measures such as the Morris method which primarily focuses on first -order interactions. Hence, we see that the errors are highest for C1. However, even for this test function we observe CUT4DGSM doing relatively much better compared to the Morris methods but fails to do better than the Monte Carlo method while estimating $\boldsymbol{\sigma}^*$. This can be attributed to the fact that, the CUT4 algorithm produces sigma points which can accurately capture up to the fifth moment of a non-linear transformation. However, evaluating $\boldsymbol{\sigma}^*$ for C1 requires higher moments than that. As a result, we see that CUT4DGSM features a higher error than MCDGSM.

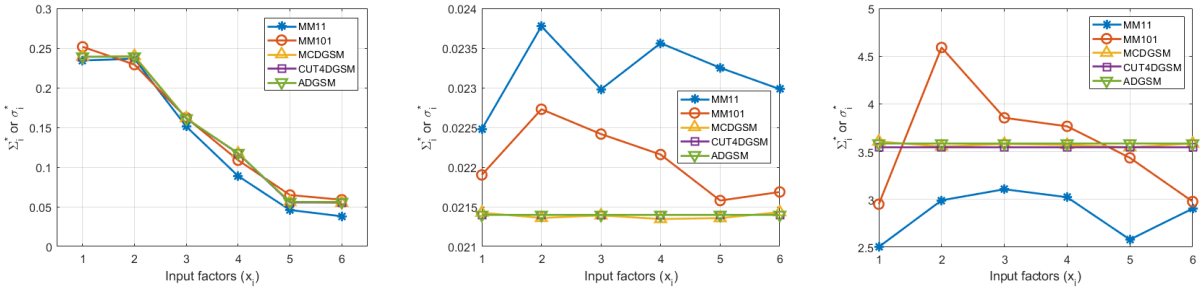
A solution to this problem could simply be to use a higher CUT algorithm such as CUT6 or CUT8 which have the ability to consider higher moments while evaluating multivariate integrals. Table III is used to present the error norm results for the test function C1 when CUT6DGSM as well as CUT8DGSM are employed. Once again, the cell with the minimum value in each column has been highlighted. We see that CUT6DGSM as well as CUT8DGSM successfully converges on the analytical values. Although more number of function evaluations are required for a more accurate estimate from CUT, the N_f value for CUT is almost two orders of magnitude lesser than Monte Carlo.

In order to present the computational benefit of our method with respect to the method of Morris as well as Monte Carlo, Figures 3 and 4 have been shown. Figure 3 presents error norms for the mean while figure 4 presents the error norms



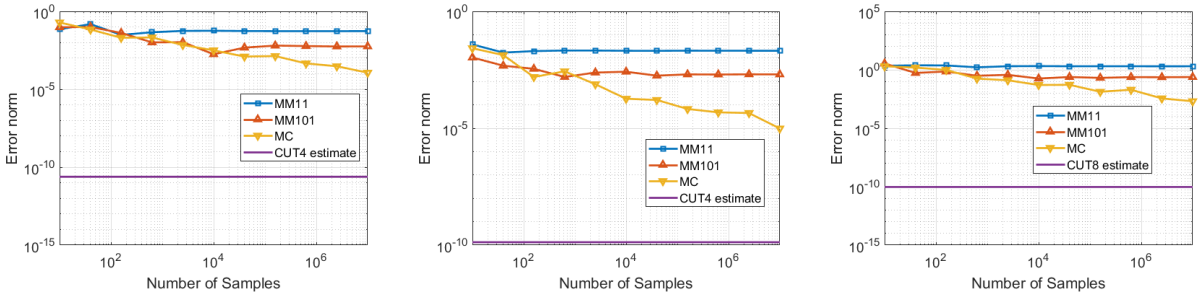
(a) A1 (b) B1 (c) C1

Fig. 1. Comparison of numerical evaluation of $\boldsymbol{M^*}$ or $\boldsymbol{\mu^*}$



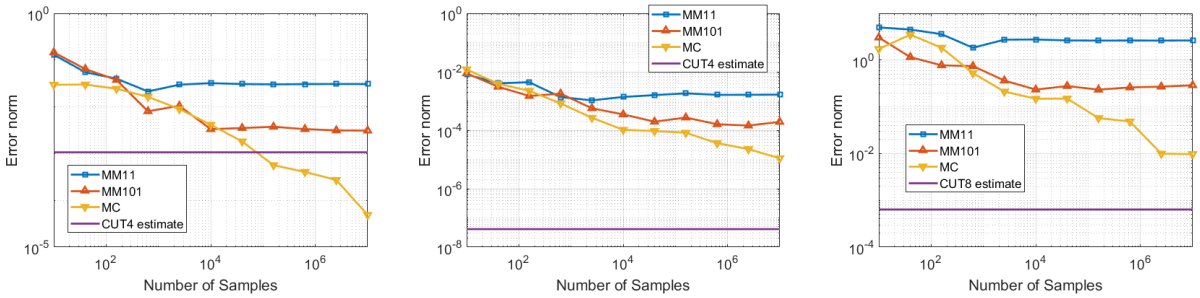
(a) A1 (b) B1 (c) C1

Fig. 2. Comparison of numerical evaluation of $\boldsymbol{\Sigma^*}$ or $\boldsymbol{\sigma^*}$



(a) (b) (c)

Fig. 3. Comparison of RMSE convergence for the mean of local sensitivities



(a) (b) (c)

Fig. 4. Comparison of RMSE convergence for the std of local sensitivities

Function	C1		
Method	$\boldsymbol{M}^*$ or $\boldsymbol{\mu}^*$	$\boldsymbol{\Sigma}^*$ or $\boldsymbol{\sigma}^*$	N_f
CUT4DGSM	0.0000	0.0978	1764
CUT6DGSM	0.0000	0.0152	2107
CUT8DGSM	0.0000	0.0006	6811

TABLE III

ERROR NORM COMPARISON FOR MEANS AND STANDARD DEVIATIONS OF LOCAL SENSITIVITIES OVER Ω CALCULATED USING CUT4, 6 AND 8

for the standard deviation of local sensitivities. The subfigures (a), (b) and (c) correspond to the test functions A1, B1 and C1 respectively.

The horizontal line in each plot marks the error norm derived from CUT while the other curves denote the error norm as the number of samples are increased. The blue and the red curves represent methods 1 and 2. The accuracy of the Morris method always stagnates after a certain number of samples. This is because the value of Δ is too large and the method fails to derive accurate estimates of local sensitivities. Hence, even though the number of local sensitivities being averaged is increased with the number of samples, the quality of each local sensitivity is rather poor: thereby leading to a plateau for all Morris method curves.

On the other hand we observe that the yellow curve steadily keeps improving in its performance with increasing sample size. This observation is expected from MC and also reiterates the fact that a huge number of samples are often required to match the level of accuracy provided by CUT.

We see that, almost categorically CUT performs better even when the number of samples are increased to 10^7.

VII. CONCLUSIONS

Although the method of Morris has been used extensively before to recognize unimportant sets of variables, we see from the illustrated examples that the Morris method could be unreliable (for a certain class of functions at least) in ranking the relevance of inputs. The Morris method has also been favoured over the years for systems where a single function evaluation could be expensive since the method yields a ranking estimate with only a few function evaluations instead of having to determine multidimensional integrals. However, we see that when CUT is used to evaluate those integrals, it can do so accurately with an equivalent number of function evaluations as Morris: thereby not only increasing the reliability of derivative-based global sensitivity measures, but also maintaining a comparable computational cost.

This paper highlights those findings by illustrating them on three test functions quoted in the literature. Comparisons are also made with Monte Carlo simulation estimates and analytical values where it is evident that the CUT based approach requires far less realizations than Monte Carlo. Convergence plots are also presented to show that, at least for the given test functions, it would require an enormous number of function evaluations for Monte Carlo to match the accuracy obtained by CUT.

ACKNOWLEDGMENT

This material is based upon work supported through National Science Foundation (NSF) under Awards No. CMMI-1537210.

REFERENCES

[1] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, *Sensitivity analysis in practice: a guide to assessing scientific models*. John Wiley & Sons, 2004.

[2] B. Sudret, "Global sensitivity analysis using polynomial chaos expansions," *Reliability Engineering & System Safety*, vol. 93, no. 7, pp. 964–979, 2008.

[3] E. H. Sandoval, F. Anstett-Collin, and M. Basset, "Sensitivity study of dynamic systems using polynomial chaos," *Reliability Engineering & System Safety*, vol. 104, pp. 15–26, 2012.

[4] J. Cao, F. Du, and S. Ding, "Global sensitivity analysis for dynamic systems with stochastic input processes," *Reliability Engineering & System Safety*, vol. 118, pp. 106–117, 2013.

[5] F. Sarrazin, F. Pianosi, and T. Wagener, "Global sensitivity analysis of environmental models: convergence and validation," *Environmental Modelling & Software*, vol. 79, pp. 135–152, 2016.

[6] B. Sudret and C. V. Mai, "Computing derivative-based global sensitivity measures using polynomial chaos expansions," *Reliability Engineering & System Safety*, vol. 134, pp. 241–250, 2015.

[7] S. Kucherenko, B. Feil, N. Shah, and W. Mauntz, "The identification of model effective dimensions using global sensitivity analysis," *Reliability Engineering & System Safety*, vol. 96, no. 4, pp. 440–449, 2011.

[8] M. D. Morris, "Factorial sampling plans for preliminary computational experiments," *Technometrics*, vol. 33, no. 2, pp. 161–174, 1991.

[9] S. Kucherenko, M. Rodriguez-Fernandez, C. Pantelides, and N. Shah, "Monte carlo evaluation of derivative-based global sensitivity measures," *Reliability Engineering & System Safety*, vol. 94, no. 7, pp. 1135–1148, 2009.

[10] I. Sobol and S. Kucherenko, "Derivative based global sensitivity measures and their link with global sensitivity indices," *Mathematics and Computers in Simulation*, vol. 79, no. 10, pp. 3009–3017, 2009.

[11] N. Adurthi, P. Singla, and T. Singh, "Conjugate unscented transformation: Applications to estimation and control," *Special section issue of Journal of Dynamic Systems, Measurement, and Control to celebrate Rudolph E. Kalman*, vol. 140, no. 3, 2018.

[12] F. Campolongo, J. Cariboni, and A. Saltelli, "An effective screening design for sensitivity analysis of large models," *Environmental modelling & software*, vol. 22, no. 10, pp. 1509–1518, 2007.

[13] A. H. Stroud, *Approximate calculation of multiple integrals*. Prentice-Hall, 1971.

[14] A. H. Stroud and D. Secrest, *Gaussian quadrature formulas*. Prentice-Hall, 1966.

[15] T. Gerstner and M. Griebel, "Numerical integration using sparse grids," *Numerical algorithms*, vol. 18, no. 3-4, p. 209, 1998.