

LAWRENCE LIVERMORE NATIONAL LABORATORY

# Self-validated Variance-based Methods for Sensitivity Analysis of Model Outputs

C. Tong

April 28, 2009

Reliability Engineering and System Safety

# Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

# Self-validated Variance-based Methods for Sensitivity Analysis of Model Outputs

Charles Tong<sup>1</sup>

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551-0808

#### Abstract

Global sensitivity analysis (GSA) has the advantage over local sensitivity analysis in that GSA does not require strong model assumptions such as linearity or monotonicity. As a result, GSA methods such as those based on variance decomposition are well-suited to multi-physics models, which are often plagued by large nonlinearities. However, as with many other sampling-based methods, inadequate sample size can badly pollute the result accuracies. A natural remedy is to adaptively increase the sample size until sufficient accuracy is obtained. This paper proposes an iterative methodology comprising mechanisms for guiding sample size selection and self-assessing result accuracy. The elegant features in the the proposed methodology are the adaptive refinement strategies for stratified designs. We first apply this iterative methodology to the design of a self-validated first-order sensitivity analysis algorithm. We also extend this methodology to design a self-validated second-order sensitivity analysis algorithm based on refining replicated orthogonal array designs. Several numerical experiments are given to demonstrate the effectiveness of these methods.

Key words: uncertainty quantification, sensitivity analysis, variance decomposition

#### 1. Introduction

Sensitivity analysis (SA) studies how variations of a model output describing certain (for example, physical, biological, or social) processes can be accounted for by variations in the control or model parameters (collectively called input factors or input parameters). In the context of the present discussion, we restrict ourselves to the sensitivity analysis of deterministic simulation models, which give identical results when presented with the same set of parameter values. Sensitivity analysis is increasingly recognized as an important tool for model building and validation. In general, sensitivity analysis is useful for all processes where it is important to know which input factors contribute most to output variability. Sensitivity analysis methods are generally classified as either local or global. Local SA methods compute or approximate the partial derivatives of model outputs with respect to individual input factors at some nominal settings. Global SA, on the other hand, studies the effects of input variations on model outputs in the entire allowable ranges of the input space. Saltelli *et al.* [7,9] have defined global SA methods by two properties:

- (i) The inclusion of influence of scales and shapes of the probability density functions for all inputs; and
- (ii) The sensitivity estimates of individual inputs are evaluated while varying all other inputs (multidimensional averaging).

In this paper we are primarily concerned with global SA methods which can generally be decomposed into four steps:

(i) Define credible ranges and distributions of input factors,

Email address: Email:chtong@llnl.gov (Charles Tong).

<sup>&</sup>lt;sup>1</sup> This work was performed under the auspices of the U.S. Department of Energy Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344.

- (ii) Create a sample of input factor values,
- (iii) Evaluate the model for each sample point, and
- (iv) Estimate the effect of each input factor on the model output.

Global SA methods can further be classified as either qualitative or quantitative. For applications with large number of input factors (tens to hundreds), the "curse of dimensionality" dictates that the computational cost for quantitative global SA becomes insurmountable. The purpose of qualitative SA studies is to identify (as opposed to quantify) the most important input factors using a relatively inexpensive set of simulation experiments, a process called "parameter screening". The goal is to enable the quantitative SA studies to focus on the smaller subset of most important input factors.

Quantitative SA methods, which apportion the output variability to individual input variabilities, typically require large number of simulation runs. When simulation models themselves are computationally intensive, the computational cost of quantitative SA may become prohibitive. To make quantitative SA more tractable, response surface modeling (not within the scope of this paper) is often used to construct inexpensive surrogates in place of the original simulation models.

Among the quantitative SA methods, variancebased methods have received the most attention. The main idea of the variance-based methods is to evaluate the variance components for all of the individual or groups of input factors. By decomposing the model function into a sum of elementary functions, Sobol' [8] has shown that a decomposition of the model output variance is possible (for independent input factors). These variance components are called Sobol' indices, and can be computed for any complex model functions. When the model is purely linear, the Sobol' indices are equivalent to the standardized regression coefficients in classical analysis. For models with Kinputs, the number of Sobol' indices is  $2^{K} - 1$ . In practice, only the first and second-order sensitivity (Sobol') indices are estimated. For large K, Homma and Saltelli [3] proposed the the "total sensitivity indices" which can be computed by using Monte Carlo simulations or the extended Fourier Amplitude Sampling Test (FAST) method.

This paper focuses on efficient and accurate methods for computing the first- and second-order sensitivity indices. Specifically, McKay's [4] main effect analysis is an efficient method for computing the first-order sensitivity indices. However, a difficulty when applying this method is the determination of a suitable sample size to achieve sufficient accuracy. One often resorts to very large samples to ensure accurate results. In this paper we first propose an improved McKay main effect analysis equipped with an adaptive accuracy assessment and improvement capability. We also propose an efficient method for computing the second-order sensitivity indices using replicated orthogonal arrays. Again, an adaptive refinement scheme is incorporated to facilitate accuracy assessment and improvement.

In Section 2 we provide a brief introduction to variance-based sensitivity analyses. Section 3 gives details of McKay's main effect analysis. Section 4 proposes an improved algorithm to more accurately compute the first-order sensitivity indices. Section 5 presents an efficient method based on replicated orthogonal arrays for computing second-order sensitivity indices. Section 6 describes an adaptive strategy similar to the improved main effect analysis for accurately computing the second-order sensitivity indices. Numerical results for simple problems are interspersed in Section 4 and 6. A more elaborative numerical example is given in Section 7.

#### 2. Variance-based Sensitivity Measures

Let Y = F(X) be a mathematical model that maps a set of K input parameters  $X \in \Re^K$  to a scalar output Y. Let E(Y) and V(Y) denote the mean and variance of the distribution of Y given probability distributions of X. A sensitivity measure for a given input  $X_i$  can be obtained by assuming a complete knowledge of the true value of  $X_i$  and assessing the effect of this knowledge on the output variance. To do this, we fix  $X_i$  at  $X_i = X_i^*$  and compute the corresponding conditional variance  $V(Y|X_i = X_i^*)$ . Since this complete knowledge of  $X_i^*$  is in general not available, we compute,  $E(V(Y|X_i))$ , which is the average of the conditional variances given the probability distribution of  $X_i$ . Intuitively,  $E(V(Y|X_i))$  measures the variance of Y when  $X_i$  is known, and so  $V(Y) - E(V(Y|X_i))$  (the additional variance due to the variability of  $X_i$  is a reasonable indicator to quantify the importance of input  $X_i$ . This indicator is equivalent to the statistical quantity called variance of conditional expectation (or VCE) via the following variance decomposition property:

$$V(Y) = V(E(Y|X_i)) + E(V(Y|X_i)).$$

The first term on the right hand side of this relation is the variance of conditional expectation (VCE), conditioned on  $X_i$ ; and the second term is the error or residual term. The residual term represents the variability in Y not accounted for by the input  $X_i$ . McKay defined the correlation ratio [4] (or main effect) by normalizing the VCE's with V(Y):

$$\eta^2(X_i) = \frac{V(E(Y|X_i))}{V(Y)}$$

A high correlation ratio implies that  $X_i$  is important in influencing the output variability. If all input factors are uncorrelated and there are no multi-way interactions, the sum of the correlation ratios is 1.

In [8], Sobol' derived a first-order sensitivity index and his derivation is based on the decomposition of Y = F(X) into a sum of terms of increasing dimensionality:

$$F(X_1, X_2, \dots, X_K) = F_0 + \sum_i F_i(X_i) + \sum_{i < j} F_{ij}(X_i, X_j) + \dots + F_{12\dots, K}(X_1, X_2, \dots, X_K)$$

where the integral of every term over any of its own input variables is zero. Sobol' showed that, when all inputs are orthogonal to each other, this decomposition is unique and that V(Y) is the sum of the variances of each term in the decomposition:

$$V(Y) = \sum_{i} V_i + \sum_{i < j} V_{ij} + \sum_{i < j < k} V_{ijk} + \dots + V_{12\dots K}$$

where  $V_i$  is the variance of  $F_i$ ,  $V_{ij}$  is the variance of  $F_{ij}$ , and so on. The total number of terms for K inputs is thus  $2^K - 1$ . The  $V_i$ 's can be shown to be equivalent to McKay's correlation ratios by the following relationship:

$$V_i = V(Y)\eta^2(X_i) = V(E(Y|X_i)).$$

Similarly,  $V_{ij}$ 's are the (pure) two-way interactions such that

$$V_{ij} = V(E(Y|X_i, X_j)) - V(E(Y|X_i)) - V(E(Y|X_j)).$$

In the event that the inputs are correlated, the above relationships no longer hold. However, variance-based measures are still useful sensitivity indicators. Input correlation will not be covered in this paper.

#### 3. Main Effect Analysis

Main effects (or first-order sensitivity indices) can be computed by directly evaluating the K integrals for the K inputs. McKay [4] proposed a more efficient estimation method based on the use of a single replicated Latin hypercube sampling (r-LHS) design for all K inputs. It should be noted that even with this efficiency improvement the main effect analysis is still very expensive requiring a substantial number (for example, thousands) of model evaluations. For models that are themselves expensive to evaluate, a common strategy to make main effect analysis feasible is to first create a response surface model (also called surrogate model, meta-model, or emulator) and perform subsequent analyses on the substantially less expensive approximate model.

In the r-LHS design, each  $X_i$  takes on distinct values  $X_{ij}, j = 1, \dots, S$  where S is the number of levels (or symbols). These values are to be replicated R times in total so that the final design has N = SR sample points.

Based on this design, the mean and variance of Y can be estimated by, for any i in  $\{1, \dots, K\}$ ,

$$\bar{Y} = \frac{1}{SR} \sum_{j=1}^{S} \sum_{r=1}^{R} Y^{(r)} (X_i = X_{ij}),$$

and

$$V(Y) = \frac{1}{SR} \sum_{j=1}^{S} \sum_{r=1}^{R} \left[ Y^{(r)}(X_i = X_{ij}) - \bar{Y} \right]^2,$$

respectively, where  $Y^{(r)}(X_i = X_{ij})$  is the output corresponding to  $X_i = X_{ij}$  in the *r*-th replication. (that is, the *R* replications amount to keeping input *i* at some fixed value and varying all others). The estimator of the conditional expectation for  $X_i = X_{ij}$  is given by

$$\bar{Y}(X_i = X_{ij}) = \frac{1}{R} \sum_{r=1}^{R} Y^{(r)}(X_i = X_{ij})$$

Finally, the formula for the variance of conditional expectation (VCE) is given by:

$$VCE(X_i) = \frac{1}{S} \sum_{j=1}^{S} \left[ \bar{Y}(X_i = X_{ij}) - \bar{Y} \right]^2 - \frac{1}{SR^2} \sum_{j=1}^{S} \sum_{r=1}^{R} \left[ Y^{(r)}(X_i = X_{ij}) - \bar{Y}(X_i = X_{ij}) \right]^2$$

and the correlation ratio for input i can be computed by normalizing VCE( $X_i$ ) with the output variance. A variant of the VCE is the biased VCE which is defined as:

$$\operatorname{VCE}_{b}(X_{i}) = \frac{1}{S} \sum_{j=1}^{S} \left[ \bar{Y}(X_{i} = X_{ij}) - \bar{Y} \right]^{2}.$$

The correlation ratio is a useful estimator for input importance for general models.

#### 4. An Improved Main Effect Analysis

To create a replicated Latin hypercube sample, both S (number of levels) and R (number of replications)

have to be specified (such that N = SR) by users. In [5], McKay investigated the variability of correlation ratio estimates as a function of sampling variability and concluded that sufficiency of the sampling design (specifically, S and R) is very important to achieve the desired precision. Specifically, large N may be needed to adequately estimate the correlation ratios. In addition, if the biased correlation ratio estimator is used, large bias may result when R is small. Saltelli *et al.* [7] recommended that S should be larger than R to give good accuracy. Despite this recommendation, it should be noted that the adequacy of a sampling design is model dependent and thus not generally known a-priori. In this section we propose a more robust main effect analysis to address this issue.

Our improved main effect analysis is based on an iterative methodology consisting of an adaptive sampling scheme and an accuracy assessment tool to monitor the convergence of the correlation ratios. Our adaptive sampling scheme borrows from our earlier work on refinement of stratified designs [11]. Our improved method currently considers only adaptively increasing S (by a factor of 2 per refinement) for accuracy improvement while keeping R fixed. To offset the effect of bias [5], we use a moderate sized R and also the unbiased correlation ratio estimator.

In the rest of this section, we first show how to adaptively refine a replicated Latin hypercube design. We will then describe the iterative procedure utilizing this adaptive sampling scheme. A few examples will be given to study the effectiveness of this improved method.

#### 4.1. Refinement for Replicated Latin Hypercube

We first denote a replicated Latin hypercube by an 3-tuple LH(N,K,S) where N, K and S are the sample size, number of input parameters, and number of symbols or levels, respectively. The number of replications can be recovered by R = N/S. We begin with a fixed R (for example, R = 50) and an initial S (for example, S = 4). The basic idea in the refinement algorithm follows two major steps. The first step involves refining each occupied grid cell (in a K-dimensional grid with S partitions in each dimension) into an  $2^{K}$ subgrid (note that we only need to examine SR grid cells so the computational cost does not scale with dimensionality). Thus, for each already occupied cell, a LH(2, K, 2) (with size 2, K inputs, and 2 levels) is created. The refined sample can be shown to preserve its original property as a replicated Latin hypercube. A selective random permutation is then applied to the newly created sample points to improve the statistical

property of the entire refined sample while leaving the original sample points unchanged. The detailed refinement algorithm (*Algorithm RefineLH*) consists of the following steps (given an initial replicated LH sample matrix Z):

**Pattern reconstruction:** Transform the sample matrix Z (an  $N \times K$  matrix) to the corresponding LH pattern matrix A by (S is the current number of levels and R is the number of replications)

$$A(i,j) = \lceil (Z(i,j) - L_j)/\delta \hat{X}_j) \rceil,$$
  
$$i = 1, \cdots, N; j = 1, \cdots, K,$$

where  $L_j$  and  $U_j$  are the lower and upper bound of input j, and  $\delta \hat{X}_j = (U_j - L_j)/S$ .

- **Replication separation:** Partition A into R individual Latin hypercube pattern matrices  $A_m, m = 1, \dots, R$  (each  $A_m$  is an  $S \times K$  matrix). Then, for each  $A_m$ ,
- **Level refinement:** Form another pattern matrix  $B_m$  (called base pattern matrix) from  $A_m$  by

$$B_m(i,j) = (\lceil A_m(i,j) \rceil - 1) * 2.$$

- New sample insertion: Create the new pattern matrix  $\tilde{A}_m$ : for each row *i* of *B*,
  - (i) Form a new LH pattern matrix  $C_i$  of size  $2 \times K$ .
  - (ii) Set  $C_i \leftarrow C_i + \begin{bmatrix} 1 & 1 \end{bmatrix}^T B_m(i)$ ,
  - (iii) Permute  $C_i$  to have one row matching  $A_m(i)$ (by first exchanging entries of row 1 of  $C_i$ with entries in the same column so that row 1 matches  $A_m(i)$ ).
  - (iv) Load  $C_i$  into row  $2 \times (i-1) + 1$  to row  $2 \times i$  of  $\tilde{A}_m$ .
- Sample randomization: Perform random permutation to each column of  $\tilde{A}_m$  only to the newly created rows.
- **Sample concatenation:** Append all  $\tilde{A}_m, m = 1, \dots, R$  matrices to form the final  $\tilde{A}$  pattern matrix.
- **Sample Generation:** Map the pattern matrix (which has number of levels = 2S now) to the new sample matrix  $\tilde{Z}$  by scaling and translation with respect to the input ranges

$$\tilde{Z}(i,j) = \tilde{A}(i,j) * \delta X_j + L_j + \epsilon(i,j)$$

where  $\epsilon(i, j)$  is a small random perturbation and its value depends on  $\tilde{A}(i, j)$ .

A detailed example of refining a LH sample is given in [11].

#### 4.2. An Adaptive Algorithm for Main Effect Analysis

The refinement technique can be used in an iterative procedure to improve the accuracy of main effect analysis. The algorithm is as follow:

- (i) Select an initial replicated LH sample with sample size  $N_0 = S_0 R$ . Prescribe a precision  $0 < \epsilon < 1$ . Set lteration = 0.
- (ii) Set Iteration = Iteration + 1. Evaluate the model using the current sample (sample points evaluated in previous iterations are skipped).
- (iii) Use the sample inputs and outputs to compute the VCE's.
- (iv) If Iteration > 1, do the following: for each  $VCE(X_i)$ , compute the error  $e_i$  by finding the difference between the current and the last  $VCE(X_i)$ ; else set  $e_i = \epsilon$ .
- (v) If  $\max e_i < \epsilon$ , terminate.
- (vi) Apply the RefineLH algorithm to create the refined LH sample. Then go back to step 2.

An alternative termination criterion for this procedure may be a prescribed maximum number of model evaluations. Using this criterion, the main effect analysis will compute both the VCE( $X_i$ )'s and the corresponding estimated error bounds. If inexpensive response surfaces are used in this analysis, This algorithm can be executed a number of times (for example, 100 as in our numerical examples) to give an even better estimates of the error bounds for the sensitivity measures.

#### 4.3. Numerical Results

In this section we demonstrate the effectiveness of our improved main effect analysis algorithm on two test examples- a monotonic and a non-monotonic function.

#### 4.3.1. A Monotonic Test Problem

The first test problem is the monotonic Sobol' function [7] given by:

$$Y = \exp\left(\sum_{j=1}^{6} b_j X_j\right) - I_6$$

where  $b_1 = 1.5, b_2 = b_3 = b_4 = b_5 = b_6 = 0.9$ ,

$$I_6 = \prod_{j=1}^6 \frac{e^{b_j} - 1}{b_j},$$

and  $X_j$  is uniformly distributed in [0, 1]. The true correlation ratios for  $X_1$  is 0.287 and 0.1057 for  $X_j$ ,  $j = 2, \dots, 6$ .

We run the adaptive algorithm 100 times, each with an initial S of 4 and R = 50. Figure 1 shows the convergence history of the 6 correlation ratios as a function of N = SR. Due to the randomness in the initial LH design and subsequent refinements, each of the



Fig. 1. Sobol' function: convergence history for the  $\eta^2$ 's (black horizontal lines- true values)

100 sets of runs goes through a different convergence path. The blue 'x' in the plots are actual correlation ratios computed at different refinement levels. We observe firstly the convergence of all sets of runs to the true values as N is increased through refinement. In general, the spread of the correlation ratios (the envelopes) shrinks as N is increased, demonstrating that increasing sample sizes also improve the confidence of the estimates.

#### 4.3.2. A Non-monotonic Test Problem

The second test problem is the Ishigami function [7]:

$$Y = sin(X_1) + 7sin^2(X_2) + 0.1X_3^4 sin(X_1),$$
  
$$X_i \in [-\pi, \pi], i = 1, 2, 3,$$

which has the following statistics

$$V(Y) = \pi^4 / 50 + \pi^8 / 1800 + 1/2 + 49/8 \approx 13.8445$$
  
$$\bar{Y} = 3.5; \quad \eta(X_1) = 0.3139;$$
  
$$\eta(X_2) = 0.4424; \quad \eta(X_3) = 0.0.$$

Again, We run the adaptive algorithm 100 times, each with an initial S of 4 and R = 50. Figure 2 shows the convergence history of the 3 correlation ratios as a function of N. Again, we observe that the correlation ratios converge to their true values as N is increased through refinement.

#### 5. Two-way Interaction Analysis

In this section we extend the idea for main effect analysis to two-way interaction (or second-order sensitivity) studies for uncorrelated inputs. In this case, we employ the following relationship

$$V(Y) = V(E(Y|X_i, X_k)) + E(V(Y|X_i, X_k))$$



Fig. 2. Ishigami function: convergence history for the  $\eta^2$ 's (black horizontal lines- true values)

where  $X_i$  and  $X_k$  are two distinct inputs under consideration. The first term on the right hand side is the variance of the conditional expectation VCE $(X_i, X_k)$ of Y, conditioned on  $X_i$  and  $X_k$ . Again, the second term is the error or residual term measuring the estimated variance of Y by fixing  $X_i$  and  $X_k$ . In addition, the correlation ratio for the input pair  $(X_i, X_k)$  is

$$\eta^2(X_i, X_k) = V(E(Y|X_i, X_k))/V(Y)$$

A high correlation ratio shows that  $X_i$  and  $X_k$  taken together are important contributors to the output variability. The variance due to the interaction term alone is defined as

$$V(X_i, X_k) = V(E(Y|X_i, X_k)) - V(E(Y|X_i)) - V(E(Y|X_k)).$$

 $V(X_i, X_k)$  can be computed using many different techniques, for example, by directly evaluating the corresponding integral. Here we illustrate its evaluation with the use of replicated orthogonal array sampling. Using orthogonal array designs with a strength of 2,  $X_i$ and  $X_k$  take on values  $X_{ij}, j = 1, \dots, S$  and  $X_{kl}, l =$  $1, \dots, S$  where S is the number of symbols (or levels). Based on this design, the mean and variance of Y can be estimated by, for any i and k in  $\{1, \dots, K\}, i \neq k$ ,

$$\bar{Y} = \frac{1}{S^2 R} \sum_{j=1}^{S} \sum_{l=1}^{S} \sum_{r=1}^{R} Y^{(r)}(X_i = X_{ij}, X_k = X_{kl})),$$

and

$$V(Y) = \frac{1}{SR} \sum_{j=1}^{S} \sum_{l=1}^{S} \sum_{r=1}^{R} \left[ Y^{(r)}(X_i = X_{ij}, X_k = X_{kl}) - \bar{Y} \right]^2$$

where  $Y^{(r)}(X_i = X_{ij}, X_k = X_{kl})$  is the output corresponding to  $X_i = X_{ij}$  and  $X_k = X_{kl}$  in the *r*-th replication (that is, keeping the two inputs at some fixed values and varying all others). The variance estimator for the expectation conditioned on  $X_i = X_{ij}$  and  $X_k = X_{kl}$  is

$$\bar{Y}(X_i = X_{ij}, X_k = X_{kl}) =$$
  
 $\frac{1}{R} \sum_{r=1}^{R} Y^{(r)}(X_i = X_{ij}, X_k = X_{kl})$ 

To approximate the variance of conditional expectation  $VCE(X_i, X_k)$ , we use

$$VCE(X_i, X_k) = \frac{1}{S^2} \sum_{j=1}^{S} \sum_{l=1}^{S} \left[ \bar{Y}(X_i = X_{ij}, X_k = X_{kl}) - \bar{Y} \right]^2 - \frac{1}{S^2 R^2} \sum_{j=1}^{S} \sum_{l=1}^{S} \sum_{r=1}^{R} \left[ Y^{(r)}(X_i = X_{ij}, X_k = X_{kl}) - \bar{Y}_{jl}) \right]^2$$

where  $Y_{jl}$  is the mean of Y at  $X_i = X_{ij}$  and  $X_k = X_{kl}$ . The two-way correlation ratio for input pair (i, k) is then obtained by normalizing VCE $(X_i, X_k)$  with the output variance V(Y). Again, we can also compute the corresponding biased estimator by ignoring the second term in the above equation.

Finally, we arrive at the following pure two-way interaction effect

$$V(X_i, X_k) = \text{VCE}(X_i, X_k) - \text{VCE}(X_i) - \text{VCE}(X_k)$$

where  $VCE(X_i)$  and  $VCE(X_i)$  can be obtained from the main effect analysis or estimated from the same replicated orthogonal array sample.

This same idea can be applied to the analysis of higher-order interaction. For example, to analyze 3way interaction, replicated orthogonal array designs of strength 3 can be used together with the corresponding formulas for computing the variance of conditional expectations.

#### 5.1. An Improved Two-way Interaction Analysis

Our improved two-way interaction analysis is again based on an iterative methodology consisting of an adaptive orthogonal array sampling scheme (based on our earlier work in [11]) and an accuracy assessment tool (similar to the one in our improved main effect analysis) to monitor the convergence of the correlation ratios. As opposed to replicated Latin hypercube designs which have a sample size growth factor of  $\approx 2$ per refinement, the sample size growth factor for orthogonal arrays is  $O(K^2)$ . Therefore, this improved algorithm is less practical than the improved main effect analysis for large K (for example, K > 5) unless the model evaluations are cheap (for example, through using response surfaces).

In the rest of this section, we first present the refinement algorithm for orthogonal arrays. We will then describe how to embed this refinement algorithm in the iterative procedure. An example will be given to assess the effectiveness of the improved method.

#### 5.2. Refinement for Replicated Orthogonal Arrays

We first denote a replicated orthogonal array by an 4-tuple OA(N,K,S,t) where N, K, S and t are the sample size, number of parameters, number of symbols or levels, and strength, respectively. The number of replications can be recovered by  $R = N/(S^t)$ . We begin with a fixed R (for example, R = 50) and an initial S (the minimum S depends on K). The basic idea in the refinement algorithm is similar to that of the Latin hypercube and it consists of the following two steps: (1) refine each grid cell (in a K-dimensional grid with S partitions in each dimension) into an  $S^{K}$ subgrid; and (2) for each grid cell that already contains a sample point, an  $OA(S^2, K, S, t)$  including the existing sample point is created. The refined sample can be shown to preserve its property as a replicated orthogonal array. A selective random permutation is then applied to the newly created sample to improve the statistical property of the refined sample while leaving the original sample points unchanged. The refinement algorithm (Algorithm RefineOA) consists of the following steps:

- **Pattern reconstruction:** same as in *Algorithm RefineLH*.
- **Replication separation:** same as in *Algorithm RefineLH*.
- **Level refinement:** For each pattern matrix  $A_m, m = 1, \dots, R$ , for another pattern matrix  $B_m$  (called base pattern matrix) from  $A_m$  by

$$B_m(i,j) = (\lceil A_m(i,j) \rceil - 1) * 2.$$

- New sample insertion: Create the new pattern matrix  $\tilde{A}_m$ : for each row *i* of *B*,
  - (i) Form a new OA pattern matrix  $C_i$  with  $OA(S^2, K, S, 2)$ .
  - (ii) Set  $C_i \leftarrow C_i + \begin{bmatrix} 1 & 1 \end{bmatrix}^T B_m(i)$ ,
  - (iii) Permute  $C_i$  to have one row matching  $A_m(i)$ (by first exchanging entries of row 1 of  $C_i$ with entries in the same column so that row 1 matches  $A_m(i)$ ).
  - (iv) Load  $C_i$  into row  $S^2 \times (i-1) + 1$  to row  $S^2 \times i$  of  $\tilde{A}_m$ .

- **Sample randomization:** same as in *Algorithm RefineLH*.
- **Sample concatenation:** same as in *Algorithm RefineLH*.
- **Sample Generation:** same as in *Algorithm RefineLH*.

A detailed example of refining an OA sample is given in [11].

5.3. An Adaptive Algorithm for Two-way Interaction Analysis

The OA refinement technique can be used in an iterative procedure to improve the accuracy of interaction analysis. The algorithm is as follow:

- (i) Select an initial replicated OA sample with sample size  $N_0 = S_0^2 R$ . Prescribe a precision  $0 < \epsilon < 1$ . Set Iteration = 0.
- (ii) Set Iteration = Iteration + 1. Then evaluate the model using the current sample.
- (iii) Use the sample inputs and outputs to compute the VCE's.
- (iv) If Iteration > 1, do the following: for each  $VCE(X_i, X_k)$ , compute the error  $e_{ik}$  by finding the difference between the current and the last  $VCE(X_i, X_k)$ ; else set  $e_{ik} = \epsilon$ ..
- (v) If  $\max e_{ik} < \epsilon$ , terminate.
- (vi) Apply the RefineOA algorithm to create a refined OA sample. Then go back to step 2.

# 5.4. Numerical Results

We tested the improved interaction analysis algorithm on the non-monotonic function described previously:

$$Y = sin(X_1) + 7sin^2(X_2) + 0.1X_3^4 sin(X_1),$$
$$X_i \in [-\pi, \pi], i = 1, 2, 3.$$

We ran the iterative algorithm 100 times, each with an initial S of 2 and R = 50. Figure 3 shows the convergence history of the 3 two-way correlation ratios as a function of sample size  $N = S^2 R$ . Once again, we observe that the correlation ratios of all 100 simulations converge to their true values as S is increased through refinement.

#### 6. An Application

In this section we present the results of applying the above two iterative algorithms to the study of a two-dimensional soil-foundation-structure-interaction



Fig. 3. Ishigami function: convergence history for the  $\eta^2$ 's (black horizontal lines- true values)



Fig. 4. A 2D soil-foundation-structure-interaction system

(2DSFSI) system (obtained from Professor Conte at UCSD) subject to earthquake excitation. The structure is a two-story two-bay reinforced concrete frame. The foundations consist of reinforced concrete squat footings at the bottom of each column. The soil is a layered clay, with stiffness properties varying along the depth (see Figure 4, Courtesy of Professor Conte and Dr. Quan Gu at UCSD).

The frame consists of two stories each of height h = 3.6m and L = 20m across (horizontal direction). The columns are modeled using displacement-based Euler-Bernoulli frame elements with distributed plasticity. Foundation footings and soil layers are modeled through isoparametric four-node quadrilateral finite elements with bilinear displacement interpolation. The constitutive behavior of the steel reinforcement is modeled by using a one-dimensional  $J_2$  plasticity model with both kinematic and isotropic linear hardening. The concrete is modeled by using a Kent-Scott-Park model with zero tension stiffening. Different material parameters are used for confined (core) and unconfined (cover) concrete in the columns. The soil is modeled by using a pressure-independent multi-yield

surface  $J_2$  plasticity material model. Different material parameters are used for each of the four layers considered. The soil under a condition of simple shear has its bottom nodes fixed and the corresponding boundary nodes at same depth tied together. The node of the beam (3 degrees of freedom) and the corresponding node on the foundation concrete block (2 degrees of freedom), at the same location, are tied together in both the horizontal and vertical directions.

After static application of the gravity loads, the structure is subjected to a base excitation taken as three times the recorder data of the 1940 Elcentro earthquake.

The original model has 19 material parameters (cover and core concretes, steel, foundation, and soil layers), 4 of which have been identified as important after a pre-screening analysis (not reported here). The ranges and distributions of these 4 parameters are given in Table 6. The model output of interest is the roof drift ratio, which is defined as the ratio of the maximum absolute displacement at the roof to the building height.

rable r	
Material	parameters

T. I. I. 1

	Parameter	Distribution	Mean $(\mu)$	range
Cover	$f_c$	Uniform	27588.5	$\pm 0.2~\mu$
Steel	$S_y$	Uniform	248200	$\pm 0.106 \ \mu$
Soil Layer #3	$ au_{\max}$	Uniform	35.0	$\pm 0.25~\mu$
Soil Layer #4	$ au_{ m max}$	Uniform	44.0	$\pm 0.25~\mu$

For these 4 parameters, we used a quasi-Monte Carlo sample of size 800 together with Friedman's multivariate adaptive regression splines (MARS) [1] interpolation scheme to create an approximate model or response surface, which is subsequently probed repeatedly to compute the first- and second-order sensitivity indices.

For the main effect analysis, we ran our iterative algorithm 100 times, each with an initial S of 4 and R = 10. Figure 5 shows the convergence history of the 4 correlation ratios as a function of sample size. The analysis shows that the soil material parameters  $\tau_3$ and  $\tau_4$  are key contributors to the output variability.

For two-way interaction analysis, we simulate our iterative algorithm 100 times, each with an initial S of 3 and R = 10. Figure 7 shows the convergence history of the 4 two-parameter correlation ratios as a function of sample size  $N = S^2 R$ . Again, we observe convergence of all the 100 runs toward 0 for most of the pairs except a small interaction between  $\tau_3$  and  $\tau_4 (\approx 7\%)$ .



Fig. 5. 2D SFSI: convergence history for the  $\eta^2$ 's



Fig. 6. 2D SFSI: convergence history for the  $\eta^2$  's (brute force)



Fig. 7. 2D SFSI: convergence history for the two-way  $\eta^2$ 's

# 7. Summary

In this paper we propose iterative schemes to more accurately compute the first- and second-order sensitivity indices. Specifically, the use of refinement techniques in stratified sampling methods such as Latin hypercube and orthogonal array together with the corresponding convergence analyses has enabled the accuracy assessment and improvement of the computed sensitivity indices. By running these improved algorithms we obtain not only accurate sensitivity measures, but also their error bounds. The effectiveness of the proposed algorithms have been demonstrated on a few numerical examples.

# References

- J. H. Friedman, Multivariate adaptive regression splines, Annals of Statistics 19.1, 1-141, 1991.
- [2] A. S. Hedayat, N. J. a. Sloane, and John Stufken, Orthogonal Arrays: Theory and Applications, Springer Series in Statistics, 1999.
- [3] T. Homma and A. Saltelli, Importance measures in global sensitivity analysis of non linear models, Reliability Engineering and System Safety 52 (1996), pp. 117.
- [4] M. D. McKay, Evaluating Prediction Uncertainty, Los Alamos National Laboratory Technical Report NUREG/CR-6311, LA-12915-MS, 1995.
- [5] M. D. McKay, Nonparametric Variance-based Methods of Assessing Uncertainty Importance, Reliability Engineering and System Safety 57(1997), pp. 267-279.
- [6] M. McKay, M. A. Fitzgerald, and R. J.Beckman, Sample Size Effects when Using R<sup>2</sup> to Measure Model Input Importance, LANL technical report.
- [7] A. Saltelli, K. Chan, E. M. Scott (editors), Sensitivity Analysis, Wiley Series in Probability and Statistics, 2000.
- [8] I. M. Sobol, Sensitivity estimates for nonlinear mathematical models, Mathematical modeling and computational experiments, 1993.
- [9] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, Sensitivity Analysis in Practice, Wiley, 2004.
- [10] M. Stein, Large Sample Properties of Simulations Using Latin Hypercube Sampling, Technometrics, 29(2), pp. 143-151.
- [11] C. Tong, Refinement Strategies for Stratified Sampling Methods, Reliability Engineering and System Safety 91 (2006), Issue 10-11, pp 1257-1265.