Validating the Assumptions of Sequential Bifurcation in Factor Screening

Shi, W.; Kleijnen, J.P.C.

Publication date:
2015

Document Version
Early version, also known as pre-print

CentER Discussion Paper

No. 2015-034

VALIDATING THE ASSUMPTIONS OF SEQUENTIAL BIFURCATION IN FACTOR SCREENING

By

Wen Shi, Jack P.C. Kleijnen

17 June 2015

ISSN 0924-7815
ISSN 2213-9532

Tilburg University

Validating the Assumptions of Sequential Bifurcation in Factor Screening

Wen Shi†
School of Logistics and Engineering Management, Hubei University of Economics, Wuhan 430205, China, shi3wen@163.com

Jack P.C. Kleijnen
Department of Management/Center for Economic Research (CentER), Tilburg University, 5000 LE Tilburg, The Netherlands, kleijnen@uvt.nl

June 17, 2015

Sequential bifurcation (SB) is a very efficient and effective method for identifying the important factors (inputs) of simulation models with very many factors, provided the SB assumptions are valid. A variant of SB called multiresponse SB (MSB) can be applied to simulation models with multiple types of responses (outputs). The specific SB and MSB assumptions are: (i) a second-order polynomial per output is an adequate approximation (valid metamodel) of the implicit input/output function of the underlying simulation model; (ii) the directions (signs) of the first-order effects are known (so the first-order polynomial approximation per output is monotonic); (iii) heredity applies; i.e., if an input has no important first-order effect, then this input has no important second-order effects. To validate these three assumptions, we develop new methods. We compare these methods through Monte Carlo experiments and a case study.

Key words: simulation; sensitivity analysis; design of experiments; statistical analysis
JEL: C0, C1, C9, C15, C44

---

1. Introduction

By definition, *factor screening*—or briefly *screening*—means searching for the really important factors—or inputs—among the many inputs that can be varied in an experiment with a given simulation model. Such screening assumes that input effects are *sparse*; i.e., only a few inputs among these many inputs are really important (we shall define "important" below). The *Pareto* principle and the *20-80* rule imply that only a few inputs (e.g., 20% of the inputs) are really important. The *parsimony* principle or *Occam's razor* implies that a simpler explanation with fewer "factors" is preferred to a more complex explanation—all other things being equal. So we conclude that there is really a need for screening in the design and analysis of simulation experiments. In simulation, the

†Corresponding author

importance of inputs depends on the *experimental domain*—or *experimental area* or *experimental frame*. Information on this domain should be given by the users of the given simulation model, including realistic ranges of the individual inputs.

The statistical theory on *design of experiments* (DOE) provides several types of screening designs; see Kleijnen (2015) for details. We focus on *sequential bifurcation* (SB) that was originally developed in Bettonvil (1990) and summarized in Bettonvil and Kleijnen (1997). Later on, other authors extended SB; see the many references in Kleijnen (2015). Recently, Shi et al. (2014a) extended SB to simulation models with multiple responses, and call this *multiresponse SB* (MSB). SB and MSB are very efficient and effective if the following assumptions are satisfied:

1. A valid metamodel is a *second-order polynomial* plus approximation error e with zero mean so $E(e) = 0$:
$$y = \beta_0 + \sum_{j=1}^{k} \beta_j x_j + \sum_{j=1}^{k-1} \sum_{j'=j+1}^{k} \beta_{j;j'} x_j x_{j'} + \sum_{j=1}^{k} \beta_{j;j} x_j^2 + e \tag{1}$$
where y denotes the the metamodel's response (output, predictor), x_j the standardized (coded, scaled) input j ($j = 1, \ldots, k$) so $-1 \leq x_j \leq 1$—if an original input is qualitative, then we randomly associate its levels with the standardized values -1 and 1—β_0 the intercept, β_j the first-order or main effect of x_j, $\beta_{j;j'}$ the interaction between the x_j and $x_{j'}$, and $\beta_{j;j}$ the purely quadratic effect of x_j.

2. The *signs* of the first-order effects β_j are known so that we can define the low and the high bounds l_j and u_j of the original (nonstandardized) input z_j such that all k first-order effects are nonnegative: $\beta_j \geq 0$ (without this assumption first-order effects might cancel each other within a group, as (3) will show).

3. If input j has no first-order effect (so $\beta_j = 0$), then this input has no second-order effects either (so $\beta_{j;j}^2 = 0$ and $\beta_{j;j'} = 0$ with $j' \neq j$). Wu and Hamada (2009) calls this the *heredity* assumption.

Section 2 summarizes SB and MSB, focusing on definitions and symbols needed for our validation methods. Section 3 details three methods for validating the three assumptions of SB and MSB. Section 4 compares three validation methods through a Monte Carlo study that does satisfy all SB or MSB assumptions. Section 5 details a case study concerning a logistics system in China. Section 6 summarizes the major conclusions.

2. SB and MSB: summary

The details of SB and MSB are given in Shi et al. (2014a). Obviously, to *estimate* the first-order effects β_j, it is most efficient to experiment with only two levels (values) per input; these levels should be realistic extreme values. The *efficiency* of SB and MSB is measured by the number of simulated input combinations and the number of replications m_i for combination i (i = 1, 2, ...); deterministic simulation (often used in engineering) implies $m_i = 1$. We try to use the same symbols as in Shi et al. (2014a)—as far as we think is reasonable; e.g., we use the symbol k instead of K.

2.1. SB summary

Let w_j denote the simulation output when the first j simulation inputs are "high" and the remaining inputs are "low" (also see Assumption 2 above):

$$w_j = f_{\text{sim}}(x_1 = 1, \ldots, x_j = 1, x_{j+1} = -1, \ldots, x_k = -1) \tag{2}$$

where f_{sim} denotes the input/output (I/O) function implicitly specified by the simulation model. Analogously, w_{-j} denotes the simulation output when the inputs 1 through j are low and the remaining inputs are high. Obviously, $w_0 = w_{-k}$ and $w_{-0} = w_k$. Let $\beta_{j'–j}$ (with endash –, not minus sign −) denote the *first-order group effect* of the standardized inputs j' through j:

$$\beta_{j'–j} = \sum_{h=j'}^{j} \beta_h. \tag{3}$$

SB applies the *foldover principle*; i.e., we simulate the *mirror* input combination besides the original combination, so −1 and 1 in the original combination become 1 and −1 in the mirror combination. Now we prove that this principle enables SB to estimate the first-order effects unbiased by the second-order effects (obviously, the "price" is that applying this principle also doubles the number of simulated combinations); we shall use this proof to develop new methods to validate the SB assumptions. The polynomial in (1) gives

$$\begin{aligned} E(w_j) = \beta_0 + \sum_{h=1}^{j} \beta_h - \sum_{h=j+1}^{k} \beta_h + \sum_{h=1}^{k} \beta_{h;h}^2 + \sum_{h=1}^{j-1} \sum_{h'=h+1}^{j} \beta_{h;h'} \\ + \sum_{h=j+1}^{k-1} \sum_{h'=h+1}^{k} \beta_{h;h'} - \sum_{h=1}^{j} \sum_{h'=j+1}^{k} \beta_{h;h'} \end{aligned} \tag{4}$$

and

$$E(w_{-j}) = \beta_0 - \sum_{h=1}^{j} \beta_h + \sum_{h=j+1}^{k} \beta_h + \sum_{h=1}^{k} \beta_{h;h}^2 + \sum_{h=1}^{j-1} \sum_{h'=h+1}^{j} \beta_{h;h'}$$
$$+ \sum_{h=j+1}^{k-1} \sum_{h'=h+1}^{k} \beta_{h;h'} - \sum_{h=1}^{j} \sum_{h'=j+1}^{k} \beta_{h;h'}, \tag{5}$$

so an unbiased estimator of the first-order group effect $\beta_{j'-j}$ is

$$\widehat{\beta}_{j'-j} = \frac{(w_j - w_{-j}) - (w_{j'-1} - w_{-(j'-1)})}{4}. \tag{6}$$

Obviously, substituting $j' = j$ into (3) and (6) gives an unbiased estimator of the *individual* first-order effect of standardized input j:

$$\widehat{\beta}_j = \frac{(w_j - w_{-j}) - (w_{j-1} - w_{-(j-1)})}{4}. \tag{7}$$

Next we consider *random simulation*, which includes discrete-event simulation and stochastic differential equation simulation. Let $w_{j;r}$ denote replication r of w_j defined in (2) with $r = 1$, ..., m_i and $m_i > 1$. We then obtain the following unbiased estimator of $\beta_{(j'-j)}$ that is the analogue of (6):

$$\widehat{\beta}_{(j'-j);r} = \frac{(w_{j;r} - w_{(-j);r}) - (w_{(j'-1);r} - w_{-(j'-1);r})}{4}. \tag{8}$$

This estimator gives the sample averages and variances

$$\overline{\widehat{\beta}}_{(j'-j)} = \frac{\sum_{r=1}^{m} \widehat{\beta}_{(j'-j);r}}{m_i} \text{ and } s^2(\overline{\widehat{\beta}}_{(j'-j)}) = \frac{\sum_{r=1}^{m} (\widehat{\beta}_{(j'-j);r} - \overline{\widehat{\beta}}_{(j'-j)})^2}{m_i(m_i - 1)}. \tag{9}$$

Note that this variance estimator allows *variance heterogeneity* of the simulation outputs (i.e., variances change as the input combinations change) and *common random numbers* (CRN) so the simulation outputs of combinations i and i' may be correlated. Based on (9), SB gives a *confidence interval* (CI) for $\beta_{(j'-j)}$ through the classic Student t-statistic with $m_i - 1$ degrees of freedom:

$$t_{m_i-1} = \frac{\overline{\widehat{\beta}}_{(j'-j)} - \beta_{(j'-j)}}{s(\overline{\widehat{\beta}}_{(j'-j)})}. \tag{10}$$

This CI enables the following test of the one-sided *null hypothesis* (H_0), which implies that Assumption 2 in Section 1 holds:

$$H_0 : \beta_{(j'-j)} > 0 \text{ versus } H_1 : \beta_{(j'-j)} = 0. \tag{11}$$

The t-statistic in (10) assumes a fixed m_i, and H_0 in (11) is the "favorite" hypothesis, which is rejected only if the statistics $\overline{\widehat{\beta}}_{(j'-j)}$ and $s(\overline{\widehat{\beta}}_{(j'-j)})$ provide serious counterevidence.

Wan et al. (2010), however, develops a *sequential probability ratio test* (SPRT) that selects m_i such that it improves the control of the type-I or α error rate ("false positive") and has no favorite null-hypothesis but considers two comparable hypotheses. This SPRT adds one replication at a time, and terminates as soon as a conclusion can be reached. This SPRT classifies inputs with $\beta_j \leq \Delta_0$ as *unimportant* and inputs with $\beta_j \geq \Delta_1$ as *important* where Δ_0 and Δ_1 are determined by the users. For these unimportant inputs, the type-I error probability is controlled such that it does not exceed α; for important inputs, the statistical power of the test should be at least γ. For *intermediate* inputs—which have $\Delta_0 < \beta_j < \Delta_1$—the power should be "reasonable". This SPRT is further discussed in Shi et al. (2014a), including experimental results and the correction of an error in Wan et al. (2010).

2.2. MSB summary

Shi et al. (2014a) extends SB to MSB for *multiresponse simulation*. MSB allows $n > 1$ output types—sometimes we briefly write n "outputs" instead of "output types". MSB declares an input group to be important if that group is important for at least one of the n output types. For simulation output l ($l = 1, ..., n$) we add the superscript l to the preceding symbols if needed; e.g., (1) becomes

$$y^{(l)} = \beta_0^{(l)} + \sum_{j=1}^{k} \beta_j^{(l)} x_j + \sum_{j=1}^{k} \sum_{j' \geq j}^{k} \beta_{j;j'}^{(l)} x_j x_{j'} + e^{(l)} \ (l = 1, ..., n). \tag{12}$$

MSB selects *input groups* such that within such a group all inputs have the same sign for a specific type of output (so no cancellation of first-order effects for this output occurs). By definition, changing the level of input j from the "low" level $L_j^{(l)}$ to the "high" level $H_j^{(l)}$ increases output l. This change, however, may decrease output $l' \neq l$. So, $L_j^{(l)}$ equals either $L_j^{(l')}$ or $H_j^{(l')}$; e.g., $L_j^{(l)} = H_j^{(l')}$ if input j has opposite effects on the outputs l and l'. Fig. 1 illustrates three situations labeled (a) through (c) when $n = 2$. Part (a) shows that all k inputs have positive—see the $+$ signs—first-order effects on both output types; so a single input group suffices ($q = 1$), because no cancellation of individual effects within the input group can occur. Part (b) shows that all k inputs have positive effects on output type 1, and negative effects—see the $-$ signs—on output type 2, so a single input group still suffices. Part (c) shows that $q = 2$ input groups are needed; input group 1 consists of the individual inputs labeled from 1 through k_1 and input group 2 consists of the individual inputs labeled from $k_1 + 1$ through k.

Figure 2 illustrates various situations for the general case of n output types requiring q input groups. To form such input groups as fewer as we could, we provides a brief description in Fig. 3.

Figure 1: Number of input groups q for two output types

(a) $q=1$

Input	Input values for $w^{(1)}$				
	Low level for $w^{(1)}$	High level for $w^{(1)}$	$w^{(1)}$	$w^{(2)}$	Input group p
1	$L_1^{(1)}$	$H_1^{(1)}$	+	+	$p=1$
2	$L_2^{(1)}$	$H_2^{(1)}$	+	+	
⋮	⋮	⋮	⋮	⋮	
k	$L_k^{(1)}$	$H_k^{(1)}$	+	+	

(b) $q=1$

Input	Input values for $w^{(1)}$				
	Low level for $w^{(1)}$	High level for $w^{(1)}$	$w^{(1)}$	$w^{(2)}$	Input group p
1	$L_1^{(1)}$	$H_1^{(1)}$	+	-	$p=1$
2	$L_2^{(1)}$	$H_2^{(1)}$	+	-	
⋮	⋮	⋮	⋮	⋮	
k	$L_k^{(1)}$	$H_k^{(1)}$	+	-	

(c) $q=2$

Input	Input values for $w^{(1)}$				
	Low level for $w^{(1)}$	High level for $w^{(1)}$	$w^{(1)}$	$w^{(2)}$	Input group p
1	$L_1^{(1)}$	$H_1^{(1)}$	+	+	$p=1$
2	$L_2^{(1)}$	$H_2^{(1)}$	+	+	
⋮	⋮	⋮	+	+	
k_1	$L_{k_1}^{(1)}$	$H_{k_1}^{(1)}$	+	+	
k_1+1	$L_{k_1+1}^{(1)}$	$H_{k_1+1}^{(1)}$	+	-	$p=2$
⋮	⋮	⋮	⋮	⋮	
k	$L_k^{(1)}$	$H_k^{(1)}$	+	-	

Figure 2: Number of input groups q for n output types

Input	Input values for $w^{(1)}$: Low level for $w^{(1)}$	High level for $w^{(1)}$	$w^{(1)}$	$w^{(2)}$	…	$w^{(n)}$	Input group p	
1	$L_1^{(1)}$	$H_1^{(1)}$	+	+	+	+		
2	$L_2^{(1)}$	$H_2^{(1)}$	+	+	+	+	$p=1$	
⋮	⋮	⋮	+	+	+	+		
k_1	$L_{k_1}^{(1)}$	$H_{k_1}^{(1)}$	+	+	+	+		Boundary 1 ($w_{k_1}^{(1)}$ and $w_{-k_1}^{(1)}$)
k_1+1	$L_{k_1+1}^{(1)}$	$H_{k_1+1}^{(1)}$	+	+	+	-		
⋮	⋮	⋮	+	+	+	-	$p=2$	
k_2	$L_{k_2}^{(1)}$	$H_{k_2}^{(1)}$	+	+	+	-		Boundary 2 ($w_{k_2}^{(1)}$ and $w_{-k_2}^{(1)}$)
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	Boundary p-1 ($w_{k_{p-1}}^{(1)}$ and $w_{-k_{p-1}}^{(1)}$)
$k_{p-1}+1$	$L_{k_{p-1}+1}^{(1)}$	$H_{k_{p-1}+1}^{(1)}$	+	⋮	⋮	⋮		
⋮	⋮	⋮	+	⋮	⋮	⋮	p	
k_p	$L_{k_p}^{(1)}$	$H_{k_p}^{(1)}$	+	⋮	⋮	⋮		Boundary p ($w_{k_p}^{(1)}$ and $w_{-k_p}^{(1)}$)
⋮	⋮	⋮	+	⋮	⋮	⋮	⋮	Boundary q-1 ($w_{k_{q-1}}^{(1)}$ and $w_{-k_{q-1}}^{(1)}$)
$k_{q-1}+1$	$L_{k_{q-1}+1}^{(1)}$	$H_{k_{q-1}+1}^{(1)}$	+	-	-	-		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	$p=q$	
k	$L_k^{(1)}$	$H_k^{(1)}$	+	-	…	-		Boundary q ($w_{k_q}^{(1)}$ and $w_{-k_q}^{(1)}$)

Figure 3: Guidelines for designing input group

(1) Select the most interesting output type among the n output types, as $w^{(1)}$.
(2) Define the values of all k inputs such that changing each individual input from $L^{(1)}$ to $H^{(1)}$ makes $w^{(1)}$ increase; i.e., all signs for $w^{(1)}$ are +.
(3) Determine the signs for the remaining output types using the signs of $w^{(1)}$ in (2).
(4) Assign a smaller index to input j, if this input has more + signs for various output types.
(5) Assign a smaller index to the superscript of output type l, if it has more + signs.

Initially, we decide to select $w^{(1)}$ as the output type that we are most interested in; this output type gets only + signs for all k inputs; see column 4 in Fig. 2. If there is no output type that we are most interested in, then we arbitrarily select one output type as $w^{(1)}$. Given our choice of $w^{(1)}$, the signs + and − of the k inputs for the $n-1$ remaining output types are known. To an output type with relatively many + signs we assign a relatively small superscript—which must be higher than the supercript (1) in $w^{(1)}$; e.g., the last output type $w^{(n)}$ in Fig. 2 has fewer + signs than the first output type $w^{(1)}$. Fig. 2 illustrates that each input group p ($p = 1, ..., q$) has two adjacent input groups $p-1$ and $p+1$, so each input group shares a boundary with its two adjacent input groups; see the thick lines of Fig. 2. We notice that input group 1 is adjacent to the input groups 2 and q, and input group q is adjacent to input groups $q-1$ and 1, so the uppermost thick line actually coincides with the lowermost thick line and, thus, we choose to label it as the last boundary; i.e., boundary q.

3. Three methods for validating the assumptions of SB and MSB

Now we present three methods for the validation of the three assumptions listed in Section 1 that are the basis of SB and MSB. The first method focuses on the important inputs and has already been detailed in Shi et al. (2014a), whereas the other two methods focus on the unimportant inputs and are not mentioned in Shi et al. (2014a).

3.1. Method 1: important inputs

We denote the number of *important inputs* by $k_{\mathrm{I}} \ll k$ where the subscript "I"stands for important. Each of these k_{I} inputs has its own magnitude for the effects in the second-order polynomial for output type l ($l = 1, ..., n$). We denote this number of important effects for output l by $q(k_{\mathrm{I}}) = 1 + k_{\mathrm{I}} + k_{\mathrm{I}}(k_{\mathrm{I}}-1)/2 + k_{\mathrm{I}}$. The case study in Section 5 is a relatively small screening experiment with $k_{\mathrm{I}} = 5$ important inputs and $k = 26$ inputs in total; yet, the number of important effects is $q(5) = 21$. To validate these screening results, Shi et al. (2014a) uses a method that is inspired by Bettonvil and Kleijnen (1997); we call this Method 1.

3.1.1 Estimating the metamodels for important inputs

To validate the three assumptions through Method 1, we first estimate the individual effects of the second-degree polynomial with the k_{I} inputs declared to be important. This estimation does not require screening, but "classic" DOE; i.e., we use a *central composite design* (CCD), which has (say) n_{CCD} combinations (CCDs are extensively discussed in the DOE literature; see again Kleijnen

(2015)). Unfortunately, a CCD is rather inefficient: $q(5) = 21 \ll n_{\text{CCD}} = 43$. Furthermore, we need to select the number of replications for the CCD; we denote this number by m_{CCD}.

Moreover, to run the simulation with these k_{I} important inputs, we also need values for all the $k - k_{\text{I}}$ *unimportant inputs*; e.g., the case study has $k - k_{\text{I}} = 26 - 5 = 21$ unimportant inputs. In Method 1, we keep the unimportant inputs *constant*; i.e., if these unimportant inputs are quantitative, then we keep them at their coded value 0 (intermediate standardized value) and if these inputs are qualitative, then—rather arbitrarily—we keep them at +1. We also check whether the $k - k_{\text{I}}$ inputs declared to be unimportant are indeed unimportant. We select (say) n_{val} combinations of the k inputs—unimportant or important inputs. Our selection of a value for n_{val} depends on the computer time required per replication and the available computer budget. We select these n_{val} combinations such that they are space-filling for the quantitative inputs (important or unimportant); i.e., we use a *Latin hypercube sample* (LHS). For a qualitative input we sample without replacement its -1 and 1 values with equal probabilities (of 0.5); i.e., $n_{\text{val}}/2$ values are -1 and the other $n_{\text{val}}/2$ values are 1. We randomly combine the n_{val} combinations of the quantitative inputs with the n_{val} values of the qualitative inputs.

Next we simulate these n_{val} input combinations, using m_{val} replications. To select a value for this m_{val}, we examine the final number of replications that the SPRT needed to test the significance of individual inputs. We use CRN when running the simulation model for these n_{val} input combinations.

3.1.2 Testing the validity of the metamodels for important inputs

In Method 1, we test the validity of the second-degree polynomial with the k_{I}-dimensional vector with estimated parameters $\widehat{\boldsymbol{\beta}}^{(l)}$ for the k_{I} important inputs, while the remaining $k - k_{\text{I}}$ unimportant inputs have zero effects of first order and second order. We therefore predict the output of type l for the n_{val} input combinations, and compare the average regression predictions $\overline{\widehat{y}}_i^{(l)} = \sum_{r=1}^{m_{\text{CCD}}} \widehat{y}_{i;r}^{(l)}/m_{\text{CCD}}$ ($i = 1, ..., n_{\text{val}}$) with the corresponding average simulated output values $\overline{w}_i^{(l)} = \sum_{r=1}^{m_{\text{val}}} w_{i;r}^{(l)}/m_{\text{val}}$. Because MSB should declare an input j to be important if $\beta_j^{(l)} \geq \Delta_1^{(l)}$, we might accept the regression predictor as valid if $|\overline{w}_i^{(l)} - \overline{\widehat{y}}_i^{(l)}| \leq \Delta_1^{(l)}$; however, this comparison is scale dependent. The Studentized statistic defined in (16) below is scale-independent because it accounts for the estimated noises $s^2(\overline{w}_i^{(l)})$ and $s^2(\overline{\widehat{y}}_i^{(l)})$. This $s^2(\overline{w}_i^{(l)})$ is the classic estimator

$$s^2(\overline{w}_i^{(l)}) = \frac{\sum_{r=1}^{m_{\text{val}}} (w_{i;r}^{(l)} - \overline{w}_i^{(l)})^2}{(m_{\text{val}} - 1)m_{\text{val}}}. \tag{13}$$

Furthermore, $\widehat{y}_{i;r}^{(l)} = \mathbf{x}_i' \widehat{\boldsymbol{\beta}}_r^{(l)}$ where $\mathbf{x}_i$ denotes the vector with the values of the independent variables determined by the CCD for the important inputs, and $\widehat{\boldsymbol{\beta}}_r^{(l)}$ denotes the $q(k_{\mathrm{I}})$-dimensional vector with the the estimated effect j for output l computed from replication r ($j = 1, \ldots, q(k_{\mathrm{I}})$, $l = 1, \ldots, n$, $r = 1, \ldots, m_{\mathrm{CCD}}$). Consequently, the following variance estimator allows unequal output variances and CRN:

$$s^2(\overline{\widehat{y}}_i^{(l)}) = \frac{\sum_{r=1}^{m_{\mathrm{CCD}}} (\widehat{y}_{i;r}^{(l)} - \overline{\widehat{y}}_i^{(l)})^2}{(m_{\mathrm{CCD}} - 1) m_{\mathrm{CCD}}}. \tag{14}$$

To validate the regression metamodel, we use the Studentized statistic with v degrees of freedom:

$$t_{i;v}^{(l)} = \frac{\max\ (|\overline{w}_i^{(l)} - \overline{\widehat{y}}_i^{(l)}| - \Delta_1^{(l)},\ 0)}{\sqrt{s^2(\overline{w}_i^{(l)}) + s^2(\overline{\widehat{y}}_i^{(l)})}} \tag{15}$$

where we select $v = \min\ (m_{\mathrm{val}} - 1,\ m_{\mathrm{CCD}} - 1)$. Because $i = 1, \ldots, n_{\mathrm{val}}$, (15) gives n_{val} observations on t for output l ($l = 1, \ldots, n$). Therefore we use Bonferroni's inequality; i.e., we replace the classic α value by $\alpha/(n_{\mathrm{val}} \times n)$ and accept the metamodel if

$$\max_{i;l}\ t_{i;v}^{(l)} \leq t_{\nu;\alpha/(n_{\mathrm{val}} \times n)}\ (i = 1, \ldots, n_{\mathrm{val}}; l = 1, \ldots, n). \tag{16}$$

If we accept this estimated second-order polynomial with k_{I} important inputs as a valid metamodel, then we may test the remaining two assumptions; namely, *known signs* of all first-order effects in this polynomial—so $\beta_j^{(l)} \geq 0$ with $j = 1, \ldots, q(k_{\mathrm{I}})$—and *heredity*—so $\beta_{j;j'} = 0$ with $j' \geq j$ and $j' = 1, \ldots, k_{\mathrm{U}}$ where k_{U} denotes the number of unimportant inputs and the subscript "U" stands for unimportant (obviously, screening implies $k_{\mathrm{U}} \gg k_{\mathrm{I}}$). We shall illustrate Method 1 in Sections 4 and 5.

3.2. Method 2: unimportant inputs

Each of the k_{U} unimportant inputs has nearly the same magnitude for its effects in the second-order polynomial for output type l; namely virtually zero (more precisely, $\beta_j \leq \Delta_0$ with $j = 1, \ldots, k_{\mathrm{U}}$; see the SPRT discussed below (11)) and thereby no significant second-order effects if the heredity assumption holds. So in Method 2 we do not need to estimate the $q(k_{\mathrm{U}})$ individual effects; it suffices to test that these k_{U} inputs have virtually no effects. So now we test the effects of the k_{U} unimportant inputs through the simulation of only *extreme* combinations of these inputs. First we explain Method 2 for simulation models with a single output type ($n = 1$) so SB suffices, for the test of first-order and second-order effects respectively; next we explain this method for MSB with two output types, as in the Chinese case-study.

3.2.1 Testing the first-order effects of the unimportant inputs

In SB with a single output type, we distinguish only the following two extreme combinations of the unimportant inputs:

(a) All k_{U} unimportant inputs are at their *low* levels (coded -1), while the k_{I} important inputs are kept fixed; e.g., the important inputs are fixed at their base levels (so the important "inputs" become "constants").

(b) All k_{U} unimportant inputs are at their *high* levels (coded 1), while keeping the k_{I} important inputs fixed at the same values as in combination (a).

To simplify our explanation, we assume that all k_{I} important inputs are *quantitative* and are fixed at their coded values 0. Furthermore, we relabel the k inputs such that SB declared the first k_{U} inputs to be unimportant. Consequently, the metamodel assumed in (1) gives the following results for combinations (a) and (b), respectively:

$$E(y \mid \mathbf{x}_{\mathrm{U}} = -\mathbf{1}) = \beta_0 - \sum_{j=1}^{k_{\mathrm{U}}} \beta_j + \sum_{j=1}^{k_{\mathrm{U}}} \sum_{j'=j}^{k_{\mathrm{U}}} \beta_{j;j'} \tag{17}$$

and

$$E(y \mid \mathbf{x}_{\mathrm{U}} = \mathbf{1}) = \beta_0 + \sum_{j=1}^{k_{\mathrm{U}}} \beta_j + \sum_{j=1}^{k_{\mathrm{U}}} \sum_{j'=j}^{k_{\mathrm{U}}} \beta_{j;j'}. \tag{18}$$

These two equations together give

$$\frac{E(y \mid \mathbf{x}_{\mathrm{U}} = \mathbf{1}) - E(y \mid \mathbf{x}_{\mathrm{U}} = -\mathbf{1})}{2} = \sum_{j=1}^{k_{\mathrm{U}}} \beta_j \tag{19}$$

We assume that the number of replications for these two combinations is m_{val}; this m_{val} may have the same value as in (13). We use CRN, to reduce the noise in our estimator of the difference

$$\delta = \frac{E(w|\mathbf{x}_{\mathrm{U}} = \mathbf{1}) - E(w|\mathbf{x}_{\mathrm{U}} = -\mathbf{1})}{2} \tag{20}$$

where w still denotes the simulation output. So we compute the m_{val} differences between the simulation outputs of the two combinations (a) and (b):

$$d_r = \frac{w_r(\mathbf{x}_{\mathrm{U}} = \mathbf{1}) - w_r(\mathbf{x}_{\mathrm{U}} = -\mathbf{1})}{2} \ (r = 1, \ldots, m_{\mathrm{val}}). \tag{21}$$

These differences give the *t-statistic for paired differences*:

$$t_{m_{\mathrm{val}}-1} = \frac{\overline{d} - E(d)}{s(d)/\sqrt{m_{\mathrm{val}}}} \tag{22}$$

with the classic estimators of the mean and standard deviation of d:

$$\overline{d} = \frac{\sum_{r=1}^{m_{\text{val}}} d_r}{m_{\text{val}}} \tag{23}$$

and

$$s(d) = \sqrt{\frac{\sum_{r=1}^{m_{\text{val}}} (d_r - \overline{d})^2}{m_{\text{val}} - 1}}. \tag{24}$$

The t-statistic defined in (22) gives a CI for the mean difference δ defined in (20). This CI may be used to test the following null-hypothesis:

$$H_0 : E(d) \leq \Delta \;\; \text{versus} \;\; H_1 : E(d) > \Delta \tag{25}$$

where $\leq$ implies that we use a one-sided hypothesis, because we assume that the first-order effects are not negative; we select

$$\Delta = k_{\text{U}} \Delta_0 \tag{26}$$

where Δ_0 was used to define unimportant inputs below (11). So we expect that an individual input is declared unimportant if its effect is Δ_0; together, the k_{U} unimportant inputs might have a total effect of $k_{\text{U}}\Delta_0$. Altogether, we accept bigger differences between the outputs for the extreme input combinations, as the number of unimportant inputs increases; also see (19). We reject H_0 defined in (25) only if $t_{m_{\text{val}}-1}$ defined in (22) with $E(d)$ following from (25) and (26) is "too high"; i.e., we reject this H_0 if

$$t_{m_{\text{val}}-1} = \frac{\overline{d} - k_{\text{U}}\Delta_0}{s(d)/\sqrt{m_{\text{val}}}} \tag{27}$$

is higher than $t_{m_{\text{val}}-1;1-\alpha}$ where $t_{m_{\text{val}}-1;1-\alpha}$ is the classic symbol for the $1 - \alpha$ quantile (or upper α point) of $t_{m_{\text{val}}-1}$.

3.2.2 Testing the second-order effects of the unimportant inputs

We also test whether the *heredity* assumption holds. This assumption implies that the k_{U} unimportant inputs have no second-order effects $\beta_{j;j'}$ ($j, j' = 1, ..., k_{\text{U}}$). Unfortunately, our test of the two extreme combinations (a) and (b), is completely insensitive to these $\beta_{j;j'}$; i.e., even if $\beta_{j;j'} \neq 0$, then these $\beta_{j;j'}$ do not affect our test (also see (19)). Therefore we now consider the *center combination* $\mathbf{x}_0 = \mathbf{0}$ where $\mathbf{0}$ denotes the k_{U}-dimensional vector with all elements equal to zero. (A center combination is also part of a CCD discussed in Section 3.1, and DOE may use this combination to test the the validity of the fitted second-order polynomial through the so-called lack-of-fit F-statistic.) Obviously, if the heredity assumption does not hold, then $E(y \mid \mathbf{x}_{\text{U}} = \mathbf{0}) \neq E(y \mid \mathbf{x}_{\text{U}} = -\mathbf{1}) = E(y \mid \mathbf{x}_{\text{U}} = \mathbf{1})$. To test the heredity assumption we assume that the number of replications for the central combination equals m_{val} (the same m_{val} is used for the two extreme combinations). We again use the CRN that are also used for the two extreme combinations. This gives the following difference:

$$\delta_0 = [\frac{E(w \mid \mathbf{x}_{\text{U}} = \mathbf{1}) + E(w \mid \mathbf{x}_{\text{U}} = -\mathbf{1})}{2}] - E(w \mid \mathbf{x}_{\text{U}} = \mathbf{0}). \tag{28}$$

We observe that—whatever the magnitudes and signs of the first-order effects are—if the second-order polynomial for the k_{U} unimportant inputs holds, then (28) becomes

$$\delta_0 = \sum_{j=1}^{k_{\text{U}}} \sum_{j'=j}^{k_{\text{U}}} \beta_{j;j'}. \tag{29}$$

Some of these $(k_{\text{U}}(k_{\text{U}} - 1)/2 + k_{\text{U}})$ second-order effects $\beta_{j;j'}$ may be negative and some may be positive, so we do not make any assumptions about the magnitude of the sum in (29). To estimate δ_0 defined in (28), we compute the m_{val} differences

$$d_{0;r} = [\frac{w_r(\mathbf{x}_{\text{U}} = -\mathbf{1}) + w_r(\mathbf{x}_{\text{U}} = \mathbf{1})}{2}] - w_r(\mathbf{x}_{\text{U}} = \mathbf{0}) \; (r = 1, ..., m_{\text{val}}). \tag{30}$$

These differences give the analogue of (22):

$$t_{0;m_{\text{val}}-1} = \frac{\overline{d}_0 - E(d_0)}{s(d_0)/\sqrt{m_{\text{val}}}}. \tag{31}$$

We use this t-statistic to test

$$H_0 : E(d_0) = 0 \text{ versus } H_1 : E(d_0) \neq 0 \tag{32}$$

where we now use a two-sided hypothesis, because the individual second-order effects may be negative or positive. Note that H_0 in (25) uses $\Delta = k_{\text{U}}\Delta_0$, whereas H_0 in (32) uses 0. We reject the latter H_0 if $|t_{0;m_{val}-1}| > t_{0;m_{val}-1;1-\alpha/2}$.

We may wish to preserve the *experimentwise* type-I error rate; experimentwise, per comparison, and familywise error rates are discussed in Miller (1981). We may then apply Bonferroni's inequality; i.e., we replace α by $\alpha/2$ because we test two null-hypotheses—namely, the hypotheses defined in (25) and (32).

3.2.3 Multiple output types and testing the effects of the unimportant inputs

Now we explain Method 2 for MSB, in case of $n \geq 2$ output types (Note that $n = 2$ in the Chinese case-study). If there were a single input group ($q = 1$), then Method 2 would be the same as what has been explained for SB in the preceding section. Therefore we suppose that there are $q \geq 2$ input groups (but Method 2 does not use these input groups, whereas Method 3 does). Fig. 4

Figure 4: Method 2 procedure for multiple output types

Testing the first-order effects:

(1) For output type l, simulate the two *extreme* input combinations $w^{(l)}(\mathbf{X}_{\mathrm{U}} = -1)$ and $w^{(l)}(\mathbf{X}_{\mathrm{U}} = 1)$.

(2) Estimate the whole group effect $d^{(l)}$; i.e., $\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}}}$.

(3) Repeat steps (1)–(2) m_{val} times, and compute the average $\bar{d}^{(l)}$ and its standard deviation $s(d^{(l)})$; i.e., $\bar{\widehat{\beta}}^{(l)}_{1-k_{\mathrm{U}}}$ and $s\left(\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}}}\right)$.

(4) Repeat steps (1)–(3) for the other output types.

(5) Test the first-order effects of unimportant inputs for output l.

Testing the second-order effects:

(6) Simulate the input combination $w(\mathbf{X}_{\mathrm{U}} = \mathbf{0})$, and repeat m_{val} times.

(7) Estimate the sum of second-order effects $d_0^{(l)} = \sum_{j=1}^{k_{\mathrm{U}}} \sum_{j'=j}^{k_{\mathrm{U}}} \widehat{\beta}^{(l)}_{j;j'}$, and compute $\bar{d}_0^{(l)}$ and $s(d_0^{(l)})$ $(l = 1, ..., n)$.

(8) Repeat step (7) for the other output types.

(9) Test the second-order effects of unimportant inputs.

provides a formal description of Method 2 for multiple output types. Initially, Method 2 simulate the two extreme combinations for each the output type successively; i.e., for output type l with $l = 1, \ldots, n$, we simulate the following two combinations:

(a) All k_{U} unimportant inputs are at their *low* levels (coded -1) for output type l, while the k_{I} important inputs are kept fixed (e.g., fixed at their base levels); we denote this combination by $\mathbf{x}_{\mathrm{U}}^{(l)} = -\mathbf{1}$.

(b) All k_{U} unimportant inputs are at their *high* levels (coded 1) for output type l, while the k_{I} important inputs are still fixed at the same values as in combination (a); we denote this combination by $\mathbf{x}_{\mathrm{U}}^{(l)} = \mathbf{1}$.

Analogously to (21) we define

$$d_r^{(l)} = \frac{w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = \mathbf{1}) - w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = -\mathbf{1})}{2} \; (r = 1, \ldots, m_{\mathrm{val}}; l = 1, \ldots, n). \tag{33}$$

We replace H_0 defined in (25) for SB by

$$H_0 : E(d^{(l)}) \leq \Delta^{(l)} \text{ versus } H_1 : E(d^{(l)}) > \Delta^{(l)} \tag{34}$$

Using (27) and Bonferroni's inequality, we reject this H_0 if

$$\max_l \frac{\overline{d}^{(l)} - k_{\mathrm{U}} \Delta_0^{(l)}}{s(d^{(l)})/\sqrt{m_{\mathrm{val}}}} > t_{m_{\mathrm{val}}-1;1-\alpha/n} \tag{35}$$

where $\Delta_0^{(l)}$ was used to define inputs that are unimportant for output l; see the text below (11).

To test the *heredity* assumption we simulate the center combination $\mathbf{x}_0 = \mathbf{0}$ (see the text above (28), defining $\mathbf{0}$ as the k_{U}-dimensional vector with all elements equal to zero; no superscript (l) is needed). We obtain m_{val} replications. Analogously to (30) we define

$$d_{0;r}^{(l)} = [\frac{w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = -\mathbf{1}) + w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = \mathbf{1})}{2}] - w_r(\mathbf{x}_{\mathrm{U}} = \mathbf{0}) \ (r = 1, ..., m_{\mathrm{val}}). \tag{36}$$

We formulate the following H_0:

$$H_0 : E(d_0^{(l)}) = 0 \text{ versus } H_1 : E(d^{(l)}) \neq 0 \tag{37}$$

To test this H_0, we compute

$$t_{0;m_{\mathrm{val}}-1}^{(l)} = \frac{\overline{d}_0^{(l)}}{s(d_0^{(l)})/\sqrt{m_{\mathrm{val}}}} \tag{38}$$

We reject this H_0 if

$$\max_l \mid t_{0;m_{\mathrm{val}}-1}^{(l)} \mid > t_{m_{\mathrm{val}}-1;1-(\alpha/n)/2} \tag{39}$$

where we use (α/n) because of n output types and /2 because (37) is two-sided (so we use the absolute value of the t-statistic).

The whole experiment is meant to test H_0 defined in (34) and H_0 defined in (37). Both hypotheses concern multiple outputs, so these hypotheses are "composite"; see again Miller (1981).

We conclude that Method 2 requires only $2n$ (extreme) combinations plus the center combination, whereas Method 1 requires $n_{\mathrm{CCD}} + n_{\mathrm{val}}$ combinations; this n_{CCD} is determined through the (rather inefficient) CCD for the k_{I} important inputs, and n_{val} is selected to make the design (possibly determined through LHS) space filling so this n_{val} will be rather arbitrary and high.

3.3. Method 3: input groups and unimportant inputs

As we did in Section 3.2, we focus on the unimportant inputs—but now we take advantage of the existence of input groups, which we illustrated in Figs. 1 and 2. As we shall see, these input groups enable us to save simulation effort because Method 3 estimates the effects of each input group for all n output types *simultaneously*. Fig. 5 gives a formal description of Method 3.

3.3.1 Testing the first-order effects of the unimportant inputs

The formation of the original q input groups may change when MSB finishes and declares inputs to be either important or unimportant. We use the symbol q_{U} to denote the number of input groups formed *only* by the k_{U} unimportant inputs. Let $\beta_{k_{p-1}+1-k_p}^{(1)}$ denote the sum of the first-order effects

Figure 5: Method 3 procedure for multiple output types

Testing the first-order effects:

(1) Simulate the two input combinations $w_{k_p}^{(1)}$ and $w_{-k_p}^{(1)}$ at boundary p ($p=1, ..., q_U$) while recording the observations on output l ($l=2, ..., n$); i.e., $w_{k_p}^{(1\to l)}$ and $w_{-k_p}^{(1\to l)}$.

(2) Estimate the input group p's effect $\widehat{\beta}_{k_{p-1}+1-k_p}^{(l)}$ ($p=1, ..., q_U$; $l=1, ..., n$) for all output types, through Theorems 1 and 2 in Shi et al. (2014a).

(3) Compute the whole group effect $d^{(l)}=\widehat{\beta}_{1-k_U}^{(l)}$ by aggregating the q_U input-group effects obtained in (2) together.

(4) Repeat steps (1)–(3) for m_{val} times, and compute the average $\bar{d}^{(l)}$ and its standard deviation $s(d^{(l)})$; i.e., $\bar{\widehat{\beta}}_{1-k_U}^{(l)}$ and $s\left(\widehat{\beta}_{1-k_U}^{(l)}\right)$.

(5) Test the first-order effects of unimportant inputs.

Testing the second-order effects:

(6) Simulate the input combination $w(\mathbf{X}_U=\mathbf{0})$, and repeat for m_{val} times.

(7) Estimate the sum of second-order effects $d_0^{(l)}=\sum_{j=1}^{k_U}\sum_{j'=j}^{k_U}\widehat{\beta}_{j;j'}^{(l)}$, and compute $\bar{d}_0^{(l)}$ and $s(d_0^{(l)})$ for all output types.

(8) Repeat step (7) for the other output types.

(9) Test the second-order effects of unimportant inputs.

for output type 1 (which is the type in which we are most interested) of input group p ($p = 1$, ..., q_U) (the second $-$ in the subscript is the endash, not the minus sign); i.e., input group p contains inputs $k_{p-1} + 1$, $k_{p-1} + 2$, ..., k_p (so the individual input k_{p-1} is the last individual input of input group $p - 1$). Shi et al. (2014a) proves two theorems (called Theorems 1 and 2) that enable the estimation of this sum for all output types simultaneously, using replication r. Let the superscript $1 \to l$ in (41) and (42) denote that output type l is observed "for free" when observing output type 1; i.e., running an input combination to observe output type 1 also generates an observation on the other output type l, so (40) and (41) or (42) have completely corresponding terms. Then these theorems imply

$$\widehat{\beta}_{k_{p-1}+1-k_p;r}^{(1)} = \frac{[w_{k_p;r}^{(1)} - w_{-k_p;r}^{(1)}] - [w_{k_{p-1};r}^{(1)} - w_{-(k_{p-1});r}^{(1)}]}{4}, \tag{40}$$

and

$$\widehat{\beta}_{k_{p-1}+1-k_p;r}^{(l)} = \frac{[w_{k_p;r}^{(1\to l)} - w_{-k_p;r}^{(1\to l)}] - [w_{k_{p-1};r}^{(1\to l)} - w_{-(k_{p-1});r}^{(1\to l)}]}{4} \; (l = 2, 3, \ldots, n), \tag{41}$$

or

$$\widehat{\beta}_{k_{p-1}+1-k_p;r}^{(l)} = -\frac{[w_{k_p;r}^{(1\to l)} - w_{-k_p;r}^{(1\to l)}] - [w_{k_{p-1};r}^{(1\to l)} - w_{-(k_{p-1});r}^{(1\to l)}]}{4} \; (l = 2, 3, \ldots, n). \tag{42}$$

If—within input group p—output types 1 and l have identical signs (either + or −), then $\beta^{(1)}_{k_{p-1}+1-k_p}$ and $\beta^{(l)}_{k_{p-1}+1-k_p}$ are estimated by (40) and (41); else (so they have opposite signs + and −), their estimators are obtained by (40) and (42).

Altogether, we can compute the unbiased estimator $\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}};r}$ by adding the q_{U} effects $\widehat{\beta}^{(l)}_{k_{p-1}+1-k_p;r}$ ($p = 1, ..., q_{\mathrm{U}}$):

$$\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}};r} = \sum_{p=1}^{q_{\mathrm{U}}} \widehat{\beta}^{(l)}_{k_{p-1}+1-k_p;r} \ (l = 1, 2, \ldots, n). \tag{43}$$

Because (40) has four terms in the numerator, it might seem that we need to simulate four input combinations for the estimation of a single input group p ($p = 1, ..., q_{\mathrm{U}}$); i.e., we seem to need $4q_{\mathrm{U}}$ input combinations to compute $\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}};r}$ in (43). In MSB, however, some input combinations applied for one input group are also used for another input group. These input combinations are identified through the boundaries between the input groups (the thick lines in Fig. 2). More specifically, two adjacent input groups—sharing a boundary—use two common input combinations; e.g., to estimate the effects of input groups $p+1$ and p, we need $w^{(l)}_{k_p;r}$ and $w^{(l)}_{-(k_p);r}$ so we save half of the simulation effort. In general, suppose that there are k individual inputs, n output types, and q input groups. We can then prove the following two theorems that state that the simulation effort depends only on q, not on n.

Theorem 3. *The total number of input combinations needed to estimate each individual input group effect* $\beta^{(l)}_{k_{p-1}+1-k_p}$ *(p = 1, ..., q; l = 1, ..., n) and their sum* $\beta^{(l)}_{1-k}$ *is* $2q$.

Theorem 4 . *The following relationship holds:* $q \leq n$.

The proofs of the theorems are given in the Online Supplement.

3.3.2 Testing the second-order effects of the unimportant inputs

Like in Method 2, the key to testing the heredity assumption is the estimator $d^{(l)}_{0;r}$ defined in (36). Unlike Method 2, Method 3 does not simulate the two extreme input combinations $\mathbf{x}^{(l)}_{\mathrm{U}} = -\mathbf{1}$ and $\mathbf{x}^{(l)}_{\mathrm{U}} = \mathbf{1}$ *for each output type* l, but computes $d^{(l)}_{0;r}$ through *the input groups*; so Method 3 requires less simulation effort.

To estimate $\beta^{(l)}_{1-k_{\mathrm{U}}}$ in (43), Method 3 simulates the two input combinations on the boundaries. Now we also use these input combinations to obtain $d^{(l)}_0$. Fig. 2 showed that the k inputs form q groups determined by q boundaries. However, there is an *exclusive* boundary for a specific output type that partitions the k inputs into two opposite groups; namely, the inputs in the group above the boundary that have have plus signs only, and the remaining inputs in the group below the boundary that have minus signs only. For example, Boundary 1 (immediately after input k_1) is the exclusive boundary for output n (last output type of the Fig. 2), because the inputs above this boundary all have plus signs and the remaining inputs below this boundary all have minus signs. If the exclusive boundary of output type l is p, then an estimator based on replication r is

$$d_{0;r}^{(1\to l)} = [\frac{w_{k_p;r}^{(1\to l)} + w_{-k_p;r}^{(1\to l)}}{2}] - w_r(\mathbf{x}_{\mathrm{U}} = \mathbf{0}) \ (r = 1, ..., m_{\mathrm{val}}; l = 2, 3, \ldots, n), \tag{44}$$

where $w_{k_p;r}^{(1\to l)}$ and $w_{-k_p;r}^{(1\to l)}$ are the two observations at boundary p on output type l (see the last column of Fig. 2) when the inputs 1 through k_p are at output 1's high level and the remaining inputs are at output 1's low level, and when the input 1 through k_p are at output 1's low level and the remaining inputs are at output 1's high level; $w_r(\mathbf{x}_{\mathrm{U}} = \mathbf{0})$ still denotes the observation at the center point. Note that output 1 and l have same signs (+) above the boundary p, whereas have opposite signs below the boundary. So, $w_{k_p;r}^{(1\to l)}$ and $w_{-k_p;r}^{(1\to l)}$ are actually the two extreme combinations (all inputs are high, and all inputs are low) for output l and thereby are identical to $w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = \mathbf{1})$ and $w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = -\mathbf{1})$ in (36).

Therefore, Methods 2 and 3 give the same test, using the same estimators for validating the first-order and second-order effects of unimportant inputs. Using input groups, Method 3 may give lower simulation cost; namely, $2q$ input combinations plus the center combination. However, this does not necessarily mean that Method 3 is always preferred: the cost of sorting inputs to form input groups as in Fig. 2 may be relatively high, especially when n is small. Moreover, simulation practitioners may find Method 3 less easy to understand and implement.

4. Monte Carlo experiment

In general, the advantage of Monte Carlo (MC) experiments is that they ensure that all SB/MSB assumptions are satisfied, so such experiments can provide information on which factors are truly important (Kleijnen, 2015). In this section, we present a MC experiment that quantifies the performance of the three validation methods labeled Methods 1, 2, and 3 described in the preceding section. Our MC experiment resembles the experiment in Shi et al. (2014a).

4.1. Designing the Monte Carlo experiments

In our MC experiment we use the second-order polynomial (12) with $k = 100$ simulation inputs, no CRN, a prespecified (nominal) type-I error $\alpha = 0.05$, and 1,000 macroreplications. Furthermore, we control the magnitudes of the effects (polynomial coefficients) and the heterogeneous response variances. Shi et al. (2014a) considers four characteristics of screening experiments in random simulation; namely, (i) sparsity of input effects, (ii) signal-noise ratio, (iii) variability of effects, and (iv) clustering of effects. Combining these four characteristics, Shi et al. (2014a) investigates sixteen combinations of these "characteristics" or "MC factors" (in this section we speak of "MC factors" and "simulation inputs"). We, however, use only one of these combinations (namely, combination 3 in Shi et al. (2014a)) as our basis. This combination implies that the noise $\epsilon^{(l)}$ in (12) is normally distributed with mean 0 and standard deviation 5; furthermore, 4 of the $k = 100$ first-order effects are "important"; namely, the effects of the simulation inputs 1, 2, 99, and 100. For Method 1 we select m_{CCD}, n_{val}, and m_{val} equal to 10. Unlike Shi et al. (2014a), we investigate the following three MC factors (starting from our basis combination); also see Table 1 (the last three columns will be discussed later):

1. Number of output types, n: We select either $n = 2$ or $n = 3$; see the combinations 1 through 8 and the combinations 9 through 15, respectively. For output type $l = 1$, we select the same thresholds as Shi et al. (2014a) do: $\Delta_0^{(1)} = 2$ and $\Delta_1^{(1)} = 4$. In practice, the thresholds for different output types may differ; i.e., if $n = 2$, then we select $\Delta_0^{(2)} = 2\Delta_0^{(1)} = 4$ and $\Delta_1^{(2)} = 2\Delta_1^{(1)} = 8$; if $n = 3$, then we select $\Delta_0^{(2)} = 2\Delta_0^{(1)} = 4$, $\Delta_0^{(3)} = 3\Delta_0^{(1)} = 6$, $\Delta_1^{(2)} = 2\Delta_1^{(1)} = 8$, and $\Delta_1^{(3)} = 3\Delta_1^{(1)} = 12$. Using Bonferroni's inequality, we replace α by $\alpha/2$ or $\alpha/3$. Notice that $\Delta_1^{(l)}$ is used by Method 1, and $\Delta_0^{(l)}$ is used by Methods 2 and 3.

2. Number of unimportant input groups, q_{U}: For simplicity's sake we make q_{U} equal to q (in practice, $q_{\text{U}} \leq q$). Furthermore, Theorem 4 stated $q \leq n$ so $q_{\text{U}} \leq n$. So if $n = 2$, then q_{U} can be either 1 or 2; actually, the combinations 1 through 4 have $q_{\text{U}} = 1$ and the combinations 5 through 8 have $q_{\text{U}} = 2$. If $n = 3$, then q_{U} can be 1, 2, or 3; e.g., the combinations 17 through 20 have $q_{\text{U}} = 3$.

3. Magnitude and sign of first-order effect, $\beta_j^{(l)}$: If there are $n = 3$ output types, then there may be $q_{\text{U}} = 3$ input groups for the unimportant inputs. Therefore we partition the $k = 100$ inputs into five clusters (subsets); namely, the inputs 1-2, 3-80, 81-90, 91-98, and 99-100 where the inputs 1-2 and 99-100 refer to important inputs and the middle three clusters include only unimportant inputs. Given these clusters, we proceed as follows with our MC design.

    - Important inputs: We select a constant value for all first-order effects of the important inputs $j = 1, 2, 99$, and 100 for a specific output type l; i.e., $|\beta_j^{(1)}| = 5$ ($> \Delta_1^{(1)} = 4$; see above), $|\beta_j^{(2)}| = 10$ ($> \Delta_1^{(2)} = 8$), and $|\beta_j^{(3)}| = 15$ ($> \Delta_1^{(3)} = 12$). Shi et al. (2014a) reports that the estimated probability of declaring input j to be important—denoted by

$\widehat{\Pr}$(DI)—is 1 for $|\beta_j^{(1)}| = 5$ and $\Delta_1^{(1)} = 4$; so $\widehat{\Pr}$(DI) is still 1 for $|\beta_j^{(2)}| = 10$ and $\Delta_1^{(2)} = 8$, and for $|\beta_j^{(3)}| = 15$ and $\Delta_1^{(3)} = 12$.

- Unimportant inputs: Whereas Shi et al. (2014a) selects the value zero for all effects of the unimportant inputs, we investigate various values for the unimportant inputs. For example, the first-order effects of the unimportant inputs 3, 4, ..., 98 are 0, 1, 2, 3 for output type 1, and 0, 2, 4, 6 for output type 2; see the combinations 1–4 where we use semicolons to distinguish effect values for different output types so $(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5}; \mathbf{10}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{10})$ in combination 1 means that the effects of the five input clusters for output type 1 are $(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5})$, and for output type 2 they are $(\mathbf{10}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{10})$ where boldface denotes vectors. For output l, the unimportant inputs within combinations {1, 5, 9, 13, 17}, {2, 6, 10, 14, 18}, {3, 7, 11, 15, 19}, and {4, 8, 12, 16, 20} have first-order effects with the values 0, $\frac{1}{2}\Delta_0^{(l)}$, $\Delta_0^{(l)}$, and $\frac{3}{2}\Delta_0^{(l)}$ respectively, so the sums of their first-order effects become 0, $\frac{1}{2}\Delta^{(l)}$, $\Delta^{(l)}$, and $\frac{3}{2}\Delta^{(l)}$ respectively. Consequently, the combinations {1, 5, 9, 13, 17}, {2, 6, 10, 14, 18}, {3, 7, 11, 15, 19} imply that H_0 in (25) does hold, where combinations {3, 7, 11, 15, 19} implies that this H_0 holds with an equality sign; combination {4, 8, 12, 16, 20} implies that H_0 does not hold. Moreover, some rows have $-$ signs, which mean that the first-order effects are negative for the corresponding output type; e.g., in combination 20 the inputs 99 and 100 have effects with the value -15 for output type 3.

4.2. Efficiency of Methods 1, 2, and 3 in MC experiment

The last three columns of Table 1 display the total number of simulation observations required by Methods 1, 2, and 3; this number quantifies the *efficiency* of the method. Obviously, Method 1 is less efficient than Method 2 or Method 3 in all combinations. Our explanation is that Method 1 includes the fitting of a second-order polynomial through a CCD; e.g., combination 1 displays 350, which is the sum of $n_{\text{CCD}} \times m_{\text{CCD}} = 25 \times 10 = 250$ and $n_{\text{val}} \times m_{\text{val}} = 10 \times 10 = 100$ where 250 is the number required by the fitting a second-order polynomial to the 5 important inputs found by MSB; this number explains the simulation effort in Method 1. Furthermore, the number of simulation observations in Method 2 is never smaller than the number in Method 3; e.g., combinations 1 through 4 show $q_{\text{U}} = 1 < n = 2$ so the number of replications in Method 2 is $2n \times m_{\text{val}} = 4 \times 10 = 40$, which is double the number in Method 3, which is $2q_{\text{U}} \times m_{\text{val}} = 2 \times 10 = 20$. In general, Method 3 is more efficient than Method 2 if $q_{\text{U}} < n$; Methods 2 and 3 are equally efficient if $q_{\text{U}} = n$. In practice it is clear whether $q_{\text{U}} = n$ or $q_{\text{U}} < n$, so we do know which method is more efficient.

Table 1: Combinations of Monte Carlo (MC) factors and resulting number of replications

	MC factors			Number of replications		
Combi	n	q_{U}	Inputs (1-2, 3-80, 81-90, 91-98, 99-100)	Method 1	Method 2	Method 3
1	2	1	$(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5}; \mathbf{10}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{10})$	350	40	20
2	2	1	$(\mathbf{5}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{5}; \mathbf{10}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{10})$	350	40	20
3	2	1	$(\mathbf{5}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{5}; \mathbf{10}, \mathbf{4}, \mathbf{4}, \mathbf{4}, \mathbf{10})$	350	40	20
4	2	1	$(\mathbf{5}, \mathbf{3}, \mathbf{3}, \mathbf{3}, \mathbf{5}; \mathbf{10}, \mathbf{6}, \mathbf{6}, \mathbf{6}, \mathbf{10})$	350	40	20
5	2	2	$(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5}; \mathbf{10}, \mathbf{0}, \mathbf{0}, -\mathbf{0}, -\mathbf{10})$	350	40	40
6	2	2	$(\mathbf{5}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{5}; \mathbf{10}, \mathbf{2}, \mathbf{2}, -\mathbf{2}, -\mathbf{10})$	350	40	40
7	2	2	$(\mathbf{5}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{5}; \mathbf{10}, \mathbf{4}, \mathbf{4}, -\mathbf{4}, -\mathbf{10})$	350	40	40
8	2	2	$(\mathbf{5}, \mathbf{3}, \mathbf{3}, \mathbf{3}, \mathbf{5}; \mathbf{10}, \mathbf{6}, \mathbf{6}, -\mathbf{6}, -\mathbf{10})$	350	40	40
9	3	1	$(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5}; \mathbf{10}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{10}; \mathbf{15}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{15})$	350	60	20
10	3	1	$(\mathbf{5}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{5}; \mathbf{10}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{10}; \mathbf{15}, \mathbf{3}, \mathbf{3}, \mathbf{3}, \mathbf{15})$	350	60	20
11	3	1	$(\mathbf{5}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{5}; \mathbf{10}, \mathbf{4}, \mathbf{4}, \mathbf{4}, \mathbf{10}; \mathbf{15}, \mathbf{6}, \mathbf{6}, \mathbf{6}, \mathbf{15})$	350	60	20
12	3	1	$(\mathbf{5}, \mathbf{3}, \mathbf{3}, \mathbf{3}, \mathbf{5}; \mathbf{10}, \mathbf{6}, \mathbf{6}, \mathbf{6}, \mathbf{10}; \mathbf{15}, \mathbf{9}, \mathbf{9}, \mathbf{9}, \mathbf{15})$	350	60	20
13	3	2	$(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5}; \mathbf{10}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{10}; \mathbf{15}, \mathbf{0}, \mathbf{0}, -\mathbf{0}, -\mathbf{15})$	350	60	40
14	3	2	$(\mathbf{5}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{5}; \mathbf{10}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{10}; \mathbf{15}, \mathbf{3}, \mathbf{3}, -\mathbf{3}, -\mathbf{10})$	350	60	40
15	3	2	$(\mathbf{5}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{5}; \mathbf{10}, \mathbf{4}, \mathbf{4}, \mathbf{4}, \mathbf{10}; \mathbf{15}, \mathbf{6}, \mathbf{6}, -\mathbf{6}, -\mathbf{15})$	350	60	40
16	3	2	$(\mathbf{5}, \mathbf{3}, \mathbf{3}, \mathbf{3}, \mathbf{5}; \mathbf{10}, \mathbf{6}, \mathbf{6}, \mathbf{6}, \mathbf{10}; \mathbf{15}, \mathbf{9}, \mathbf{9}, -\mathbf{9}, -\mathbf{15})$	350	60	40
17	3	3	$(\mathbf{5}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{5}; \mathbf{10}, \mathbf{0}, \mathbf{0}, -\mathbf{0}, -\mathbf{10}; \mathbf{15}, \mathbf{0}, -\mathbf{0}, -\mathbf{0}, -\mathbf{15})$	350	60	60
18	3	3	$(\mathbf{5}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{5}; \mathbf{10}, \mathbf{2}, \mathbf{2}, -\mathbf{2}, -\mathbf{10}; \mathbf{15}, \mathbf{3}, -\mathbf{3}, -\mathbf{3}, -\mathbf{15})$	350	60	60
19	3	3	$(\mathbf{5}, \mathbf{2}, \mathbf{2}, \mathbf{2}, \mathbf{5}; \mathbf{10}, \mathbf{4}, \mathbf{4}, -\mathbf{4}, -\mathbf{10}; \mathbf{15}, \mathbf{6}, -\mathbf{6}, -\mathbf{6}, -\mathbf{15})$	350	60	60
20	3	3	$(\mathbf{5}, \mathbf{3}, \mathbf{3}, \mathbf{3}, \mathbf{5}; \mathbf{10}, \mathbf{6}, \mathbf{6}, -\mathbf{6}, -\mathbf{10}; \mathbf{15}, \mathbf{9}, -\mathbf{9}, -\mathbf{9}, -\mathbf{15})$	350	60	60

Note. Symbol "–" before a number means negative effect on output l.

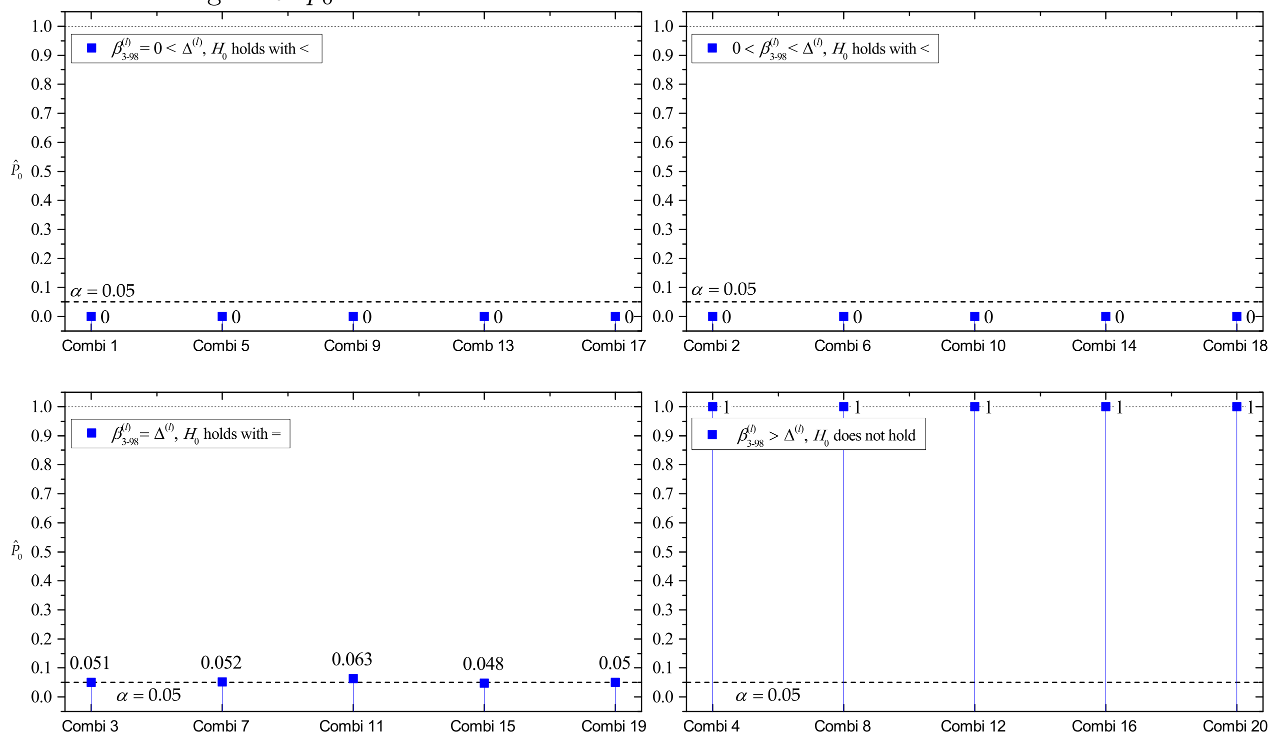
4.3. Effectiveness of Methods 1, 2, and 3 in MC experiment

To quantify the *effectiveness* of the three methods, we estimate p_0, which denotes the probability of rejecting H_0 defined in (34). More precisely, we obtain 1,000 macroreplications and record the percentage of macroreplications that rejects H_0. Ideally a method should have $\widehat{p}_0 \leq \alpha$ if H_0 holds, where α is the nominal type-I error probability. First we compute $\widehat{p}_0$ for various magnitudes of the first-order effects of the inputs declared to be "unimportant"; next we compute $\widehat{p}_0$ for various magnitudes of the second-order effects of these "unimportant" inputs. Obviously, $1 - \widehat{p}_0$ estimates the Type-II error rate.

4.3.1 First-order effects of unimportant inputs

Figure 6 presents $\widehat{p}_0$ in combinations {1, 5, 9, 13, 17}, {2, 6, 10, 14, 18}, {3, 7, 11, 15, 19}, and {4, 8, 12, 16, 20} for Method 2 only, because Method 3 gives similar results. The x-axis lists specific combinations and the y-axis gives the corresponding $\widehat{p}_0$; e.g., the upper-left plot gives $\widehat{p}_0$ for combinations 1, 5, 9, 13, and 17 which have zero aggregated effects for the $k_{\mathrm{U}} = 96$ unimportant inputs so these aggregated effects are much smaller than $\Delta^{(l)} = 96\Delta_0^{(l)}$. This figure shows that $\beta^{(l)}_{1-k_U}$ strongly influences $\widehat{p}_0$; e.g., the upper two plots show $\widehat{p}_0 = 0$ if $\beta^{(l)}_{3-98} < \Delta^{(l)}$, the lower-right plot shows $\widehat{p}_0 = 1$ if $\beta^{(l)}_{3-98} > \Delta^{(l)}$ and the lower-left plot shows $\widehat{p}_0 \approx \alpha = 0.05$ if $\beta^{(l)}_{3-98} = \Delta^{(l)}$ (so

Figure 6: $\widehat{p}_0$ of Method 2 for various combinations of first-order effects



$\beta^{(l)}_{3-98}$ reaches its maximum while H_0 still holds).

From these plots we conclude that Methods 2 and 3 give appropriate type-I and type-II error rates. We do not display results for Method 1, because this method turns out to give relatively high $\widehat{p}_0$ when there are considerably many unimportant inputs. Our explanation is that Method 1 uses (15), which has the term $\Delta^{(l)}_1$ so it does not consider the *aggregated* effects of the unimportant simulation inputs. These aggregated effects may increase the difference between $\overline{w}^{(l)}_i$ and $\overline{\widehat{y}}^{(l)}_i$; this difference increases the probability of rejecting H_0 as more unimportant inputs are involved.

4.3.2 Various second-order effects of unimportant inputs

Whereas Table 1 implies that all second-order effects of the unimportant inputs are exactly zero, we now allow non-zero second-order effects so that we can investigate the heredity assumption for Methods 2 and 3. We start from combination 17 in Table 1, so all first-order effects of the unimportant inputs for the three output types are exactly zero. Next we investigate the following four cases for second-order effects of these unimportant inputs; also see Table 2.

Case 1: The heredity assumption does hold, so input j has the first-order effect $\beta^{(l)}_j = 0$ and this input also has zero second-order effects $\beta^{(l)}_{j;j'} = 0$ with $j \leq j' = 1, \ldots, k_{\mathrm{U}}$. Actually, Case 1

Table 2: Second-order effects of unimportant inputs

Case	The values of $\beta^{(l)}_{j;j'}$	δ_0	Heredity holds?
1	Each unimportant input has zero $\beta^{(l)}_{j;j'}$ ($j \leq j' = 1, 2, \ldots, k_{\mathrm{U}}$)	$= 0$	Yes
2	One unimportant input has positive $\beta^{(l)}_{j;j}$; i.e., $\beta^{(l)}_{10;10} = c\Delta^{(l)}_1$ ($c = 0.01, 0.1, 1, 25, 50, 100$)	> 0	No
3	Two unimportant inputs have non-zero opposite $\beta^{(l)}_{j;j}$; i.e., $\beta^{(l)}_{10;10} = c\Delta^{(l)}_1$, $\beta^{(l)}_{20;20} = -c\Delta^{(l)}_1$ ($c = 0.01, 0.1, 1, 25, 50, 100$)	$= 0$	No
4	Each unimportant input has small positive $\beta^{(l)}_{j;j'}$; i.e., $\beta^{(l)}_{j;j'} = c\Delta^{(l)}_0$ ($j \leq j' = 1, 2, \ldots, k_{\mathrm{U}}; k_{\mathrm{U}} = 10, 20, 40, 80$; $c = 0.0001, 0.001, 0.002, 0.005, 0.01, 0.05$)	> 0	No

is identical to combination 17, in which the unimportant inputs labeled 3 through 98 have zero first-order and second-order effects. Consequently, the definition in (29) gives $\delta^{(l)}_0 = 0$. Therefore, we expect $\widehat{p}_0 \leq \alpha$ for Case 1.

Case 2: The unimportant input #10 has the first-order effect $\beta^{(l)}_{10} = 0$, but its purely quadratic effect is $\beta^{(l)}_{10;10} = c\Delta^{(l)}_1$ where c is one of the following six values: 0.01, 0.1, 1, 25, 50, 100 ($\Delta^{(l)}_1$ was defined below (11), and denotes the threshold exceeded by the first-order effects of important inputs). We expect that if c increases, then $\widehat{p}_0$ exceeds α more and more—until $\widehat{p}_0$ reaches its maximum value of 1.

Case 3: The unimportant inputs #10 and #20 have purely quadratic effects that may be much higher than their first-order effects—but these quadratic effects have opposite signs; i.e., $\beta^{(l)}_{10;10} = c\Delta^{(l)}_1$ and $\beta^{(l)}_{20;20} = -c\Delta^{(l)}_1$ where c and $\Delta^{(l)}_1$ are defined as in Case 2. Consequently, these quadratic effects cancel out so $\delta^{(l)}_0 = 0$. Therefore, we expect $\widehat{p}_0 = \alpha$ even if c is high; i.e., our test has little power in this case.

Case 4: Each unimportant input has a zero first-order effect (so $\beta^{(l)}_j = 0$ with $j = 1, \ldots, k_{\mathrm{U}}$) and has relatively small second-order effects so $\beta^{(l)}_{j;j'} = c\Delta^{(l)}_0$ where c has (very small) values from 0.0001 to 0.05 ($\Delta^{(l)}_0$ was defined below (11) as the threshold not exceeded by the first-order effects of unimportant inputs). The aggregated effects of the unimportant inputs may still be high, if there are many unimportant inputs so k_{U} is high so $\delta^{(l)}_0$ is high; see (29). Therefore, we expect $\widehat{p}_0$ to vary between the desired value α and the maximum value 1, as c or k_{U} increases.

For each of these four cases the 1,000 macroreplications give $\widehat{p}_0$, which now denotes the percentage of macroreplications that rejects H_0 in (32). Because Methods 2 and 3 give almost the same $\widehat{p}_0$, we display $\widehat{p}_0$ only for Method 2. Case 1 gives $\widehat{p}_0 = 0.049$, which is very close to the desired value $\alpha = 0.05$—as we expected. Cases 2 through 4 give the six estimated power curves in Fig. 7

for various c; the first two plots (in the first row) display $\widehat{p}_0$ for Cases 2 and 3, while the remaining four plots give $\widehat{p}_0$ for Case 4 with $k_{\rm U} = 10, 20, 40, 80$. This figure demonstrates that $\delta_0^{(l)}$ has a profound effect on $\widehat{p}_0$, as we detail next.

- If $\delta_0^{(l)} = 0$, then $\widehat{p}_0 \approx \alpha = 0.05$; e.g., in Case 3 (upper-right plot), the two quadratic effects cancel out so $\delta_0^{(l)} = 0$ and all observed values for $\widehat{p}_0$ (see the squares) are close to the dashed line that corresponds with $\alpha = 0.05$—no matter how much c changes.
- If $\delta_0^{(l)} \geq \Delta_1^{(l)}$, then $\widehat{p}_0 \uparrow 1$; e.g., in Case 2 (upper-left plot), c= 1 so $\delta_0^{(l)} = \beta_{10;10}^{(l)} = \Delta_1^{(l)}$ (the threshold for importance) gives an estimated probability of rejecting H_0 that is as high as 0.994. Moreover, in Case 4 (each "unimportant" input has second-order effects) even a small $k_{\rm U}$ or c gives $\widehat{p}_0 = 1$ because the sum $\delta_0^{(l)}$ may still exceed $\Delta_1^{(l)}$; e.g., $k_{\rm U} = 10$ and $c = 0.05$ (the rightmost point in the middle-left plot) gives $\delta_0^{(l)} = [(10 \times 9)/2 + 10] \times 0.05 \times \Delta_0^{(l)} = 2.75\Delta_0^{(l)} > \Delta_1^{(l)}$, so $\widehat{p}_0 = 1$.
- If $0 < \delta_0^{(l)} \leq \Delta_1^{(l)}$, then $\alpha < \widehat{p}_0 \leq 1$; e.g., Case 4 with $k_{\rm U} = 40$ and $c = 0.001$ (the second point in the lower-left plot) gives $\delta_0^{(l)} = [(40 \times 39)/2 + 40] \times 0.001 \times \Delta_0^{(l)} = 0.82\Delta_0^{(l)} < \Delta_1^{(l)}$, and $\widehat{p}_0 = 0.341 < 1$. Similarly, $\widehat{p}_0 = 0.731 < 1$ if $k_{\rm U} = 20$ and $c = 0.005$ (the fourth point in the middle-right plot) so $\delta_0^{(l)} = 1.05\Delta_0^{(l)} < \Delta_1^{(l)}$.
- The more $\delta_0^{(l)}$ approximates $\Delta_1^{(l)}$, the bigger $\widehat{p}_0$ becomes; e.g., $\delta_0^{(l)} = 0.82\Delta_0^{(l)} < \delta_0^{(l)} = 1.05\Delta_0^{(l)}$ gives $\widehat{p}_0 = 0.341 < \widehat{p}_0 = 0.731$. Similar results are found for other $\widehat{p}_0$-values between 0.05 and 1.

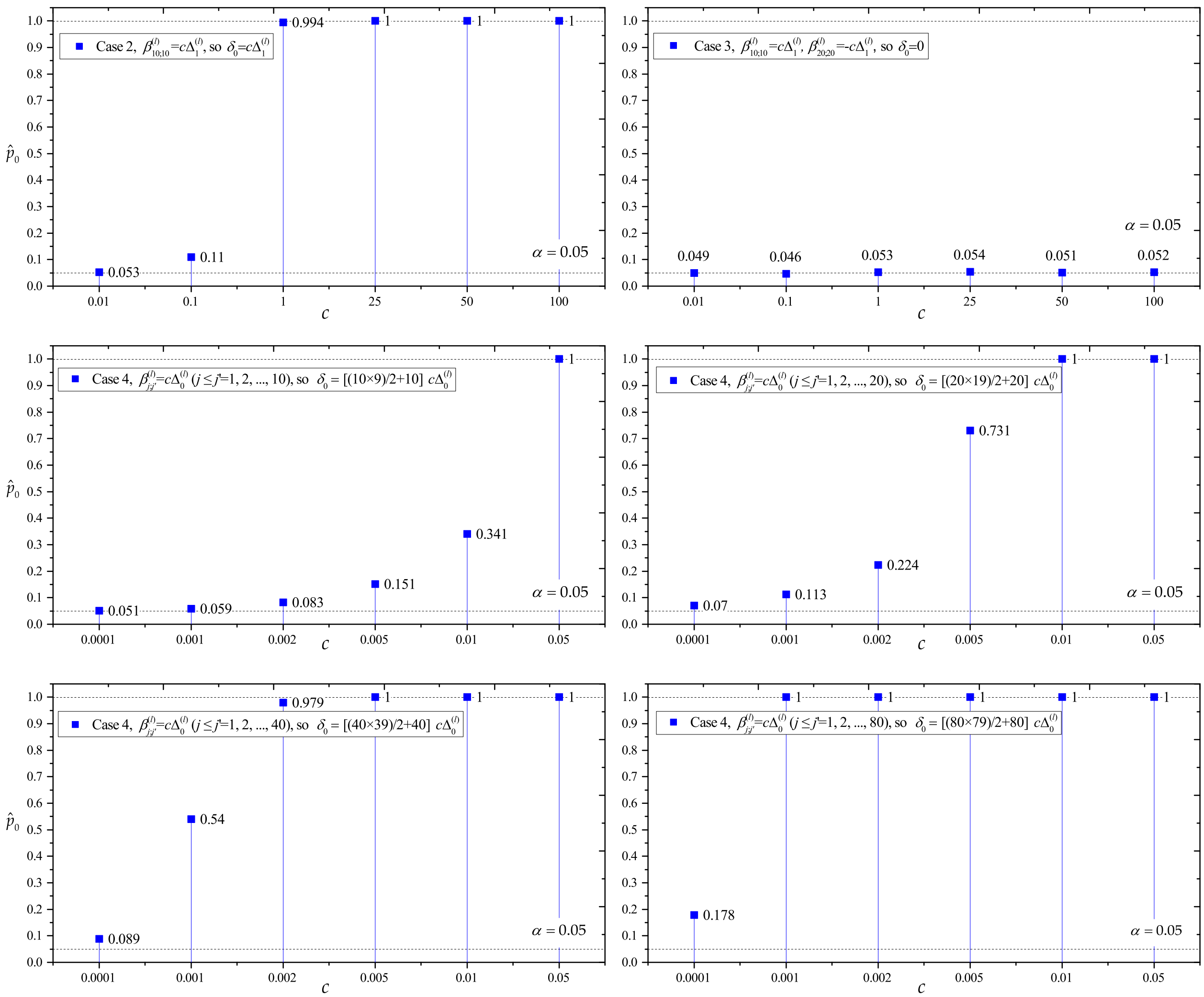
We conclude that Method 2 and Method 3 have so much power that they detect most cases that violate the heredity assumption—except for Case 3, in which two quadratic effects cancel out exactly. We find Case 3 rather pathological, so we do not further discuss Case 3.

5. Case study: a logistics simulation in China

Whereas the MC experiments in the preceding section enabled us to check the *validity* of the proposed three methods, the case study in the present section enables us to investigate the *robustness* of these methods in practice.

Shi et al. (2014a) presents a case study concerning a discrete-event simulation model for a Chinese *third-party logistics* (TPL) company that wants to improve the *just-in-time* (JIT) system for its customer, a car manufacturer; details are given in Shi et al. (2014b). We summarize this case study as follows.

Figure 7: Estimated power $\widehat{p}_0$ of Method 2 for various cases with second-order effects



The simulation includes a flow of parts, truck scheduling, etc. The TPL company expects to open another assembly plant; when this new plant will open, the current TPL capacity will not meet the logistic needs. Management wants to maintain the current logistic performance, measured through the *average cycle time* (CT) of a part and the *number of throughput* (NT) per "month" or 30-day period. A high CT conflicts with the JIT philosophy. NT is the sum of the shipments collected at the part suppliers and delivered to the assembly plants within a production cycle of 30 days. The goal of this case study is to identify the inputs that are important for one or both performance measures (CT, NT).

Table 3 describes the $k = 26$ simulation inputs, and their "high" and "low" values; i.e., changing an input from $L^{(\text{CT})}$ to $H^{(\text{CT})}$ makes CT increase—such an increase is denoted by the plus sign in the next-to-last column. All inputs are quantitative, except for input 23 which is the queueing discipline. Inputs 1 through 5 are known to have the same (plus) signs for both outputs, whereas the remaining 21 inputs have opposite signs for the two outputs (namely, plus for CT and minus for NT). Consequently, we can define two *input groups*; namely, group 1 with inputs 1 though 5, and group 2 with the remaining inputs (labeled 6 through 26).

Shi et al. (2014a) uses the SPRT with $\Delta_1^{(\text{CT})} = 5$ and $\Delta_1^{(\text{NT})} = 3,000$ (performance improvement not to be missed), and $\Delta_0^{(\text{CT})} = 2.5$ and $\Delta_0^{(\text{NT})} = 2,000$ (minimum critical values). SB and MSB find five important inputs; namely, the inputs 4, 5, 14, 17, and 20—see the inputs printed in boldface in Table 3. Obviously, inputs 4 and 5 are in input group 1, and inputs 14, 17, and 20 are in input group 2. Furthermore, SB and MSB declare the same inputs to be important; i.e., SB identifies the inputs 4, 5, 14, 17, and 20 for CT and input 17 for NT. SB requires 355 replications, whereas MSB requires only 233 replications.

5.1. Method 1: case-study results

Shi et al. (2014a) applies Method 1 to test the SB and MSB assumptions. For this case study we summarize Method 1 and its results, as follows. A CCD is used for the $k_{\text{I}} = 5$ inputs that SB and MSB declare to be important. This CCD enables the estimation of the 21 individual effects in a second-degree polynomial. The number of replications per CCD combination is selected to be $m_{\text{CCD}} = 10$; this number is based on the number of replications in the last stages of the SPRT that is used in SB and MSB. The unimportant quantitative inputs are fixed, at their coded value 0, and the one unimportant qualitative input that is fixed at +1 which denotes FIFO (FIFO is the default queueing rule of the current supply-chain). Replication r ($r = 1, ..., m_{\text{CCD}} = 10$) of all the CCD combinations use CRN.

Table 3: Inputs of the Chinese logistics simulation; inputs printed in bold are found to be important

ID	Description	$L^{(\mathrm{CT})}$	$H^{(\mathrm{CT})}$	CT	NT
1	Pick-up orders time from the 1th milk-run (hours)	2.2	1.1	+	+
2	Pick-up orders time from the 2th milk-run (hours)	1.8	1	+	+
3	Pick-up orders time from the 3th milk-run (hours)	1.5	1	+	+
4	**Pick-up orders time from the 4th milk-run (hours)**	2.5	1.2	+	+
5	**Pick-up orders time from the 5th milk-run (hours)**	1.6	0.8	+	+
6	Setup time in a part supplier (minutes)	10	15	+	−
7	Loading time of unit parts in part supplier (minutes)	2	3	+	−
8	Unloading time of unit parts in CDDC (minuters)	2	4	+	−
9	Scanning tiem of unit parts in CDDC (minutes)	20	30	+	−
10	Loading time of unit parts in CDDC (minutes)	2	4	+	−
11	Unloading time of unit part in factory warehouse (minutes)	1.5	2.5	+	−
12	Ratio between pick-up suppliers and in milk-run i	40%	60%	+	−
13	Passing rate of scanning	1%	2%	+	−
14	**Number of receiving doors**	30	10	+	−
15	Number of shipping doors	30	10	+	−
16	Number of forklifts	20	10	+	−
17	**Number of LTL trucks**	40	20	+	−
18	Number of TL trucks	60	50	+	−
19	Velocity of forklifts	30	20	+	−
20	**Velocity of LTL transportation**	100	75	+	−
21	Velocity of TL transportatioin	100	75	+	−
22	Threshold time at temporary storage area	20	24	+	−
23	Queue discipline of LTL trucks	SPT	FIFO	+	−
24	Velocity of trailers	10	5	+	−
25	Number of trailers	20	10	+	−
26	Velocity of conveyors in CDDC	24	12	+	−

Analysis of variance (ANOVA) of the resulting simulation I/O data gives second-order polynomials with $R^2 = 0.9608$ and $R^2_{\mathrm{adj}} = 0.9519$ for CT, and $R^2 = 0.9641$, and $R^2_{\mathrm{adj}} = 0.9588$ for NT, whereas the first-order polynomials have only $R^2 = 0.7022$, $R^2_{\mathrm{adj}} = 0.6683$ for CT, and $R^2 = 0.6988$ and $R^2_{\mathrm{adj}} = 0.6733$ for NT. So, the second-order polynomials are much better, and these metamodels seem adequate for predicting the outputs.

Given that these second-order polynomials are adequate, it makes sense to examine their individual estimated coefficients. It turns out that the signs of the estimated first-order effects of the important inputs agree with the assumed signs; namely, inputs 4 and 5 have minus signs for both CT and NT (these inputs have $L^{(\mathrm{CT})} > H^{(\mathrm{CT})}$ in Table 3), and inputs 14, 17, and 20 have opposite signs for these outputs. So the assumption of known signs for all first-order effects seems to hold for the important inputs.

To test that all first-order and second-order effects of all *unimportant* inputs are zero, $n_{\mathrm{val}} = 10$ new combinations are selected. These combinations are selected through LHS with uniform sampling of values between -1 and 1 for all 25 quantitative inputs, and sampling the two values -1 and 1 for the qualitative input. For each combination, the number of replications is selected to be $m_{\mathrm{val}} = 20$. Altogether, these 10 LHS combinations with their 20 replications give the simulated

Table 4: Method 1 for validating the SB and MSB assumptions in new combination i

i	1	2	3	4	5	6	7	8	9	10
$\overline{w}_i^{CT}$	28.09	31.27	25.06	45.22	42.02	67.57	24.30	38.63	27.25	58.65
$s^2(\overline{w}_i^{CT})$	1.38	4	0.38	0.80	0.02	1.70	0.26	1.92	1.04	1.30
$\overline{\widehat{y}}_i^{CT}$	34.03	30.01	26.53	49.40	40.78	61.26	26.26	34.37	24.96	54.58
$s^2(\overline{\widehat{y}}_i^{CT})$	0.28	0.57	0.77	1.22	0.50	1.76	0.64	0.72	0.41	0.38
t_i^{CT}	2.28			0.83		1.78		0.78		
$\overline{w}_i^{NT}$	49,387	53,664	53,122	45,513	51,563	38,952	51,003	44,424	48,402	51,562
$s^2(\overline{w}_i^{NT})$	64,407	209,913	36,151	9,250	52,819	23,850	97,016	30,896	21,505	3,876
$\overline{\widehat{y}}_i^{NT}$	46,738	52,531	51,475	43,665	50,991	40,323	51,397	45,007	51,669	48,317
$s^2(\overline{\widehat{y}}_i^{NT})$	52,303	35,195	32,936	43,594	38,718	26,054	49,384	35,292	52,609	22,877
t_i^{NT}	0	0	0	0	0	0	0	0	0.98	1.50

Table 5: Validation results of Methods 2 and 3

	$d^{(\mathrm{CT})}$	$d^{(\mathrm{NT})}$	$d_0^{(\mathrm{CT})}$	$d_0^{(\mathrm{NT})}$
$\overline{d}$	17.05	10,223.73	10.43	-4,643.33
$s(d)$	1.92	922.18	2.92	915.24

$\overline{w}$ and the predicted $\overline{\widehat{y}}$ and their estimated variances $s^2(\overline{w})$ and $s^2(\overline{\widehat{y}})$ displayed in Table 4. These prediction errors are tested through a t_v-statistic with degrees of freedom v = min $(10-1, 20-1)$ = 9. Furthermore, $\alpha = 0.20$; such a relatively high α value is typical when applying Bonferroni's inequality. This test uses the critical value $t_{10-1;(0.20/(10\times2))} = t_{9;0.01} = 2.821$. This table shows that $\max_{i;l}\ t_i^{(l)} = 2.28$ (this 2.28. is found in column $i = 1$ for CT), but this maximum value is not significant so the two metamodels are accepted. We conclude that in this case study Method 1 does not reject the three assumptions of SB and MSB.

5.2. Methods 2 and 3: case-study results

Table 3 showed that SB and MSB find $k_{\mathrm{U}} = 21$ *unimportant* inputs. These inputs imply $q_{\mathrm{U}} = 2$ *input groups*; namely, input group 1 comprising inputs 1 through 3 and input group 2 comprising the remaining 18 unimportant inputs. Because q_{U} equals the number of output types ($n = 2$), Methods 2 and 3 require the same number of input combinations—namely, $2n = 2q = 4$—to test the first-order effects. To test the second-effects (featuring in the heredity assumption), Methods 2 and 3 need one more combination; namely, the center point. Like Method 1, we use Methods 2 and 3 with $m_{\mathrm{val}} = 20$ replications per simulated combination. Altogether, we require $(4+1)\times 20 = 100$ simulation observations. We use a "per comparison" error rate $\alpha = 0.05$ (Method 1 used a "familywise" rate of $\alpha = 0.20$).

Equation (33) implies the definition of $d_r^{(\mathrm{CT})}$ and $d_r^{(\mathrm{NT})}$ $(r = 1, ..., m_{\mathrm{val}} = 20)$. Their sample averages $\overline{d}$ and their standard deviations $s(d)$ follow from (23) and (24), and are displayed in the columns 2 and 3 of Table 5; we shall discuss $d^{(\mathrm{CT})}_{0;r}$ and $d^{(\mathrm{NT})}_{0;r}$ in columns 4 and 5, below.

Using (35) with $\Delta^{(\mathrm{CT})} = k_{\mathrm{U}}\Delta^{(\mathrm{CT})}_0 = 52.5$ and $\Delta^{(\mathrm{NT})} = k_{\mathrm{U}}\Delta^{(\mathrm{NT})}_0 = 42,000.00$, we obtain $t^{(\mathrm{CT})}_{19}$ $= -82.6$ and $t^{(\mathrm{NT})}_{19} = -154.1$. These negative values are much smaller than the positive critical value $t_{20-1;1-0.05/2} = t_{19;0.975} = 2.093$ where we use /2 because $n = 2$ and we use Bonferroni's inequality. So, we do not reject H_0 in (25) for CT and NT.

Note: Obviously, the preceding test neglects the possibility of a first-order effect on (say) CT of one input declared to be unimportant, that is actually higher than the threshold $\Delta^{(\mathrm{CT})}_0 = 2.5$ while the other unimportant inputs have zero first-order effects. The sum of the $k_{\mathrm{U}} = 21$ unimportant inputs equals $\overline{d}^{(\mathrm{CT})} = 17.05$, so if these inputs had the same first-order effects, then these estimated effects would be $17.05/21 = 0.81$—which is considerably less than $\Delta^{(\mathrm{CT})}_0 = 2.5$. Furthermore, $\overline{d}^{(\mathrm{CT})}$ $= 17.05$ is small compared with $\widehat{\beta}^{(\mathrm{CT})}_{1-26} = 46.41$ (sum of first-order effects of $k = k_{\mathrm{I}} + k_{\mathrm{U}} = 26$ inputs), whereas $\widehat{\beta}^{(\mathrm{CT})}_{4,5,14,17,20} = 31.71$ (sum of first-order effects of the $k_{\mathrm{I}} = 5$ important inputs). Analogous results hold for the other output, NT.

The last two columns of Table 5 display $\overline{d}_0$ and $s(d_0)$ for CT and NT, respectively; see (36). The two-sided test in (39) gives $|t^{(\mathrm{CT})}_{0;19}| = 15.93$ and $|t^{(\mathrm{NT})}_{0;19}| = 22.69$. Selecting $\alpha = 0.05$, we obtain the critical value $t_{20-1;1-0.05/(2\times 2)} = t_{19;0.9875} = 2.4334$, which is much smaller so we reject H_0 in (25).

The results of Method 1 suggested that a second-order polynomial with only the important inputs adequately explains the effects of the inputs on the outputs; i.e., the unimportant inputs seem to have small second-order (and first-order) effects. Methods 2 and 3, however, suggest that there are many of these small second-order effects so their sum is statistically significant. Altogether, Methods 2 and 3 require only a few simulation observations, but may give misleading results; so, next we may apply the more expensive Method 1 to validate the assumptions of SB and MSB.

6. Conclusions

We proposed two novel methods (called Method 2 and 3) for the validation of the assumptions of SB and MSB. These assumptions are: (i) a second-order polynomial is an adequate approximation of the implicit I/O function defined by the simulation model; (ii) this polynomial has known signs of its first-order effects; (iii) if an input has no important first-order effect, then it has no important second-order effects either (heredity assumption).

Originally, Shi et al. (2014a) proposed a method—that we call Method 1—to validate these three assumptions, and focuses on the important inputs only (screening assumes that only "a few" inputs are really important). A CCD enables the estimation of a second-order polynomial—per output type—with the important inputs determined by SB or MSB. This estimated polynomial is tested through a comparison of the simulation outputs and the metamodel predictions, for several input combinations selected through LHS.

Instead of estimating a second-order polynomial for the important inputs, the new methods called Method 2 and Method 3 test the aggregated first-order effects of the unimportant inputs, simulating only a few combinations. Method 2 simulates two extreme combinations for a given output type, to estimate the aggregated first-order effects of the unimportant inputs; Method 2 also simulates the center combination to estimate the aggregated second-order effects of the unimportant inputs. Whereas Method 2 applies its validation procedure to each type of output successively, Method 3 uses a more efficient procedure that uses input groups formed by the unimportant inputs in the presence of multiple output types. If the number of these input groups is less than the number of output types, Method 3 saves input combinations compared with Method 2.

Our Monte Carlo experiments—based on one of Shi et al. (2014a)'s scenarios—show that Method 2 and Method 3 require very few replications compared with Method 1. Comparing Method 2 and Method 3 when the number of input group is smaller than the number of output types shows that Method 3 requires fewer input combinations and thus a smaller total number of replications than Method 2. Our experiments also compare the effectiveness of the three methods. The experiment with various magnitudes of the first-order effects shows that Method 1 gives bad results when there are many unimportant inputs, whereas Method 2 and Method 3 give appropriate type-I error and power. The experiment with second-order effects shows that Method 2 and Method 3 have enough power to detect most cases that violate the heredity assumption.

We also experiment with the logistics case-study in Shi et al. (2014a), with its 5 important inputs and 21 unimportant inputs. Method 1 accepts the three assumptions of SB or MSB. Methods 2 and 3 require only a few simulation observations, and reject the heredity assumption. Next, the results of Methods 2 and 3 may be double-checked through the more expensive Method 1.

Acknowledgments

This work is partly supported by the National Natural Sciences Foundation of China under Grants (NSFC) 71402048 and 71372134.

References

Bettonvil, B., Kleijnen, J. P. C., 1997. Searching for important factors in simulation models with many factors: Sequential bifurcation. European Journal of Operational Research 96 (1), 180–194.

Bettonvil, B. W. M., 1990. Detection of important factors by sequential bifurcation. Ph.D. thesis, Universiteit van Tilburg.

Kleijnen, J. P. C., 2015. Design and Analysis of Simulation Experiments, 2nd Edition. Springer US, New York.

Miller, R. G., 1981. Simulation Statistical Inference, 2nd Edition. McGraw-Hill, New York.

Shi, W., Kleijnen, J. P., Liu, Z., 2014a. Factor screening for simulation with multiple responses: Sequential bifurcation. European Journal of Operational Research 237 (1), 136 – 147.

Shi, W., Shang, J., Liu, Z.-X., Zuo, X.-L., 2014b. Optimal design of the auto parts supply chain for jit operations: Sequential bifurcation factor screening and multi-response surface methodology. European Journal of Operational Research 236, 664–676.

Wan, H., Ankenman, B. E., Nelson, B. L., 2010. Improving the efficiency and efficacy of controlled sequential bifurcation for simulation factor screening. INFORMS Journal on Computering 22 (3), 482–492.

Wu, J. C. F., Hamada, M., 2009. Experiments: Planning, Analysis, and Optimization, 2nd Edition. Wiley, New York.