Factor Screening for Simulation with Multiple Responses

Shi, W.; Kleijnen, Jack P.C.; Liu, Zhixue

Publication date:
2012

CentER

Discussion Paper

No. 2012-032

FACTOR SCREENING FOR SIMULATION WITH MULTIPLE RESPONSES: SEQUENTIAL BIFURCATION

By

Wen Shi, Jack P.C. Kleijnen, Zhixue Liu

April 16, 2012

ISSN 0924-7815

TILBURG UNIVERSITY

Factor Screening for Simulation with Multiple Responses: Sequential Bifurcation

Wen Shi [1], Jack P.C. Kleijnen [2], Zhixue Liu [1]
[1]School of Management, Huazhong University of Science and Technology, 430080 Wuhan, China
[2]CentER and Department of Information Management, Tilburg University, Post Box 90153, 5000 LE Tilburg, The Netherlands

April 10, 2012

Abstract

Factor screening searches for the really important inputs (factors) among the many inputs that are changed in a realistic simulation experiment. Sequential bifurcation (or SB) is a sequential method that changes groups of inputs simultaneously. SB is the most efficient and effective method if the following assumptions are satisfied: (i) second-order polynomials are adequate approximations of the input/output (I/O) functions implied by the simulation model; (ii) the signs of all first-order (or main) effects are known; (iii) if two inputs have no important first-order effects, then they have no important second-order effects either (heredity property). This paper examines SB for random simulation with multiple responses (outputs), called multi-response SB (MSB). This MSB selects "batches" of inputs such that within a batch all inputs have the same sign for a specific type of output, so no cancellation of main effects occurs. MSB also applies Wald's sequential probability ratio test (SPRT) to obtain enough replicates for correctly classifying a group effect or an individual effect as important or unimportant. MSB enables efficient selection of the initial number of replicates in SPRT. The paper also proposes a procedure to validate the three assumptions of MSB. The performance of MSB is examined through extensive Monte Carlo experiments that satisfy all MSB assumptions, and through a case study representing a logistic system in China; MSB performance is very promising.

Keywords: design of experiments; curse of dimensionality; sparse effects
JEL: C0, C1, C9, C15, C44

1 Introduction

Whereas real-world experiments typically have only a small number of factors (less than ten?), *simulation* experiments may involve numerous factors or (simulation) inputs; e.g., the ecological case study by Bettonvil and Kleijnen (1997) has 281 inputs. Screening is defined as searching for the really important inputs among the many inputs that are changed in a simulation experiment (so-called Pareto or sparsity-of-effects principle; also see the general review of design and analysis of simulation experiments by Kleijnen et al. 2005). This search should also cure "the curse of dimensionality", so that after the screening phase the analysts can apply classic metamodels such as low-order polynomial regression and Kriging models .

There are several *screening* methods, including classic two-level designs, frequency domain experimentation, supersaturated designs, and group-screening designs; details are given by Kleijnen (2008) and Kleijnen (2009). In this paper, however, we focus on the screening method called *sequential bifurcation* (SB), because it seems the most efficient and effective method—if its assumptions are satisfied. The most important assumptions are: (i) a second-order polynomial is an adequate approximation of the input/output (I/O) function implied by the simulation model**;** **(**ii) the signs of all first-order (or main) effects in this polynomial are known; (iii) if two inputs have no important first-order effects, then they have no important second-order effects either (so-called "heredity property"; see Wu and Hamada 2009). Several case studies illustrate that these assumptions may be realistic; see again Bettonvil and Kleijnen (1997) and also Kleijnen, Bettonvil, and Persson (2006), Wan, Ankenman, and Nelson (2010), and the case study in Section 4.

SB is a sequential group-screening method. By definition, *sequential* designs select the input combinations (also called design or inputs points) as the experimental results become available: sequential methods learn from preceding experimental results. *Group* screening means that individual inputs are treated as a group; i.e., if the group changes from one value to another value, then all its individual inputs do so. SB's first (or initial) step aggregates all inputs of the simulation model into a single group. In each step, SB tests whether the current group has a significant main effect. SB discards non-significant subgroups. SB splits a group with a significant effect into two (smaller) subgroups; i.e., it uses bifurcation. In the next step SB tests whether these subgroups have significant main effects; SB splits significant subgroups into smaller subgroups—until the main effects of all significant individual inputs are estimated.

The first paper on SB is Bettonvil and Kleijnen (1997), assuming deterministic simulation. Next, SB is extended to random (stochastic) discrete-event simulation by Cheng (1997). SB for random simulation is also discussed by Kleijnen et al. (2006), illustrating SB through a supply-chain case study with 92 inputs. Wan, Ankenman, and Nelson (2006) combine Bettonvil and Kleijnen (1997)'s SB with two hypothesis-testing procedures, to control the type-I and type-II error probabilities; next, Wan et al. (2010) generalize their procedure to improve its efficiency and efficacy. Shen and Wan (2009) develop a controlled sequential factorial design (CSFD) that combines a traditional factorial design with sequential hypothesis testing; Shen, Wan, and Sanchez (2010) further improve the efficiency by combining CSFD and the design in Wan et al. (2010).

Based on this literature survey, we conclude that SB is a relatively new method that has attracted the attention of several researchers. However, no researchers examine SB (or other screening methods) for simulation with *multiple* responses (outputs, performance measures). In practice, a simulation model does provide multiple responses; e.g., supply chain management (SCM) simulation gives multiple performance measures; see the surveys by Kleijnen (2003), Kleijnen (2008), and Kleijnen and Smits (2003). Examples of SCM case studies with multiple simulation outputs are (in historical order) Shang and Tadikamalla (1998), Chan and Spedding (2001), Dabbas et al. (2001), Shang and Tadikamalla (2004), Kumar and Nottestad (2006), Yalcnkaya and Mirac Bayhan (2009), and Ekren et al. (2010). Note that these case studies do not use screening; i.e., they assume a very limited number of inputs, all of which are important. (The famous psychological study Miller (1956) reports that human's capacity for processing information is limited to seven plus or minus two responses.)

Naive SB for simulation with multiple outputs applies SB to each type of output successively. We, however, propose *multi-response SB* (MSB). Our main conclusion will be that MSB is more efficient (fewer simulated input combinations and replicates, so less computer time) and more effective (higher probability of finding important inputs). Note that efficiency is crucial for computationally expensive simulation.

This paper is organized as follows. Section 2 extends SB to multiple outputs, and uses similar assumptions as SB does. Because an individual input may increase some type of output and decrease another type of output, MSB selects "batches" of inputs such that within a batch all inputs have the same sign for a specific type of output. We detail MSB for only two output types. To determine the number of replicates, MSB applies the sequential probability ratio test (SPRT). This section also gives a procedure to validate the three assumptions of MSB. Section 3 compares the performance of MSB and SB through Monte Carlo experiments that satisfy all (M)SB assumptions; this section also includes a more efficient rule for selecting the initial number of replicates per stage. Section 4 evaluates the robustness of MSB through a case study representing a logistic system in China; MSB turns out to require fewer replicates than SB. Section 5 presents the main conclusions.

2 Multi-response Sequential Bifurcation (MSB)

The basic idea of our MSB is inherited from SB. So, MSB contains a sequence of steps in which the main effects of input groups are estimated and tested. Specifically, if a group is declared to be non-significant, then all inputs in the group are classified as unimportant and discarded in the next steps. However, if a group is declared significant, then this group is split into two subgroups for further evaluation. A basic MSB rule is to declare a group of inputs to be important if at least one of the (multiple) outputs shows significant changes. Moreover, the unique feature of MSB is its attempt to estimate the main effects of groups for all outputs, while minimizing the experimental effort compared with SB for a single output. Details are given in the next subsections.

2.1 MSB Symbols and Definitions

Table 1 gives the major symbols and their definitions in MSB. These symbols slightly differ from those in Bettonvil and Kleijnen (1997). Their definitions will become clear in the next subsections.

2.2 MSB Assumptions

We use the following three basic assumptions for MSB, which Bettonvil and Kleijnen (1997) also use for SB.

Assumption 1: An adequate metamodel (of the I/O function implied by the underlying simulation model) for output l is a second-order polynomial.

This metamodel is denoted by

$$y_l = \beta_{(l);0} + \sum_{j=1}^{K} \beta_{(l);j} x_j + \sum_{j=1}^{K} \sum_{j'=j}^{K} \beta_{(l);j;j'} x_j x_{j'} + \epsilon_l \tag{1}$$

where the K inputs are standardized (coded, normalized) such that $-1 \leq x_j \leq 1$ $(j = 1, \ldots, K)$; for output l the intercept is $\beta_{(l);0}$, the K main effects are $\beta_{(l);j}$, the $K(K-1)/2$ two-factor interactions are $\beta_{(l);j;j'}$ with $j < j'$, the K purely quadratic effects are $\beta_{(l);j;j}$; ϵ_l is the metamodel residual for output l with zero mean (because the metamodel is assumed to be "adequate"). Note that this standardization $(-1 \leq x_j \leq 1)$ makes the input effects scale-free so they are comparable when determining the important inputs. (Bettonvil and Kleijnen (1997) assume a metamodel without purely quadratic effects so $\beta_{(l);j;j} = 0$). To estimate the main effects in (1), it is efficient to select only two values per input. In practice, the users of the underlying simulation model should provide these values and ensure that these values are realistic extreme values, given the goal of the simulation model.

Assumption 2: The signs of all main effects are known; i.e., it is known that either $\beta_{(l);j} \geq 0$ or $\beta_{(l);j} \leq 0$ $(j = 1, \ldots K)$ for any given j and l.

Assumption 2 is a basic assumption of all group-screening methods, because this assumption avoids cancellation of individual main effects within the group effect.

In practice, Assumption 2 may indeed hold, as the case study in Section 4 will illustrate. This case study concerns a Chinese automobile-parts supply-chain in which some inputs are logistic resources (e.g., the number of trucks in trunk-line transportation, the number of trucks in branch-line transportation, and the number of receiving doors in the cross-docking distribution center). Obviously, the more logistic resources are available, the lower is the waiting time of auto parts in each logistic node so the lower is the cycle time (CT); CT is one of

Table 1: Major MSB symbols and their definitions

Symbol	Definition
α	Type I error probability
$\beta_{(l);j}$	Main (first-order) effect of input j for output l
$\beta_{(l);j'-j}$	Sum of main effects of inputs $j\prime$ through j for output l
γ	Power (complement of type II error probability)
Δ_0	Threshold such that $\beta_j \leq \Delta_0$ is "unimportant"
Δ_1	Threshold such that $\beta_j \geq \Delta_1$ is "important"
ϵ_l	Residual in metamodel for output l with zero mean
$\sigma^2_{(l)}(\mathbf{x})$	Variance of ϵ_l for input combination $\mathbf{x}$
$H_{(l):i}$	Value of input i that gives highest value for output l
K	Total number of inputs in simulation experiment
k_p	Number of inputs in batch p
$L_{(l):i}$	Value of input i that gives lowest value for output l
$m_{j'-j}$	Final number of replicates when estimating $\beta_{(l);j'-j}$ through SPRT
$M_{(l);j'-j}$	Maximum number of replicates when estimating $\beta_{(l);j'-j}$ through SPRT
n	Number of simulation output types
$N_{0;j'-j}$	Initial sample size when estimating $\beta_{(l);j'-j}$
p	Identity number of batch ($p = 1, ..., q$)
q	Number of batches
r	Current number of replicates in a given stage
$w_{(l);(j)}$	Simulation output l when inputs 1 through j are at $H_{(l)}$ and inputs $(j+1)$ through K are at $L_{(l)}$
$w_{(l);-(j);r}$	Simulation output l when inputs 1 through j are at $L_{(l)}$ and inputs $(j+1)$ through K are at $H_{(l)}$
$w_{(l\to l');(j)}$	Simulation output l' when inputs 1 through j are at $H_{(l)}$ and inputs $(j+1)$ through K are at $L_{(l)}$
$w_{(l\to l');-(j)}$	Simulation output l' when inputs 1 through j are at $L_{(l)}$ and inputs $(j+1)$ through K are at $H_{(l)}$
x_j	Standardized value of input j
y_l	Metamodel output for simulation output l ($l = 1, ...n$)

Table 2: Input values for two output types

Input	(a) Input values for w_1				(b) Input values for w_2			
	Low level for w_1	High level for w_1	w_1	w_2	Low level for w_2	High level for w_2	w_1	w_2
1	$L_{(1);1}$	$H_{(1);1}$	+	+	$L_{(2);1}$	$H_{(2);1}$	+	+
2	$L_{(1);2}$	$H_{(1);2}$	+	+	$L_{(2);2}$	$H_{(2);2}$	+	+
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
k_1	$L_{(1);k_1}$	$H_{(1);k_1}$	+	+	$L_{(2);k_1}$	$H_{(2);k_1}$	+	+
k_1+1	$L_{(1);k_1+1}$	$H_{(1);k_1+1}$	+	−	$L_{(2);k_1+1}$	$H_{(2);k_1+1}$	−	+
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
K	$L_{(1);K}$	$H_{(1);K}$	+	−	$L_{(2);K}$	$H_{(2);K}$	−	+

the outputs of interest. Another output is number of throughput (NT) of parts; obviously, the number of resources have positive effects on this other output (NT). This case study illustrates that the means of the simulation output types may either decrease or increase *monotonically* as a specific simulation input increases.

Assumption 3: If two inputs have no important first-order effects, then they have no important second-order effects either.

2.3 MSB Mathematical Details

Definition 1 *Let $\beta_{(l);j'-j}$ be the sum of the main effects of inputs j' through j for output l:*

$$\beta_{(l);j'-j} = \sum_{i=j'}^{j} \beta_{(l);i}. \tag{2}$$

Definition 2 *Changing the level of input i from $L_{(l);i}$ to $H_{(l);i}$ makes output l increase.*

Note that this change may make another output l' decrease, so $L_{(l);i}$ equals either $L_{(l');i}$ or $H_{(l');i}$ where $l \neq l'$; e.g., $L_{(l);i} = H_{(l');i}$ if input i has opposite effects on the outputs l and l'. Table 2 gives an example with K inputs and $n = 2$ outputs; columns 4 and 5 show that the inputs 1 through k_1 have the same signs for both outputs, while inputs $k_1 + 1$ through K have opposite signs; changing from $L_{(l);i}$ to $H_{(l);i}$ with $l = 1, 2$ and $1 \leq i \leq k_1$ increases both outputs, whereas changing from $L_{(l);i}$ to $H_{(l);i}$ with $l = 1$ and $k_1 + 1 \leq i \leq K$ increases output w_1 but decreases output w_2. If we wish to increase w_2 for all K inputs, then we should use part (b) of this table; i.e., $1 \leq i \leq k_1$ implies $L_{(1);i} = L_{(2);i}$ and $H_{(1);i} = H_{(2);i}$, but $k_1 + 1 \leq i \leq K$ implies $L_{(1);i} = H_{(2);i}$ and $H_{(1);i} = L_{(2);i}$.

Definition 3 *Let $w_{(l);(j)}$ denote output l when inputs 1 through j are at $H_{(l)}$ and the remaining inputs ($j+1$ through K) are at $L_{(l)}$. Let $w_{(l);-(j)}$ denote output l when inputs 1 through j are at $L_{(j)}$ and the remaining inputs are at $H_{(l)}$.*

The example of Table 2 implies that $w_{(1);(K)}$ is the output of w_1 when all K inputs have the values $H_{(1)}$; $w_{(1);-(K)}$ is the output when all K inputs have the values $L_{(1)}$. Following Bettonvil and Kleijnen (1997), we call $w_{(l);-(j)}$the *mirror* observation of $w_{(l);(j)}$. The definition applies the so-called *foldover* principle, which was originally developed for real-life experiments with a few factors such that two-factor interactions do not bias the estimators of the main effects (see Montgomery 2007); in SB and MSB the foldover principle ensures that second-order effects (two-factor interactions and purely quadratic effects) do not bias the first-order (main effect) estimators—as we shall prove in the next paragraph.

The metamodel assumed in (1) implies

$$\begin{aligned} E(w_{(l);(j)}) &= \beta_{(l);0} + \beta_{(l);1} + \cdots + \beta_{(l);j} - \beta_{(l);j+1} - \cdots - \beta_{(l);K} + \\ & \quad \beta_{(l);1;2} + \cdots - \beta_{(l);1;K} + \beta_{(l);1;1} + \cdots + \beta_{(l);K;K} \end{aligned} \tag{3}$$

and

$$\begin{aligned} E(w_{(l);-(j)}) &= \beta_{(l);0} - \beta_{(l);1} - \cdots - \beta_{(l);j} + \beta_{(l);j+1} + \cdots + \beta_{(l);K} + \\ & \quad \beta_{(l);1;2} + \cdots - \beta_{(l);1;K} + \beta_{(l);1;1} + \cdots + \beta_{(l);K;K}. \end{aligned} \tag{4}$$

Hence, $w_{(l);(j)}$ and $w_{(l);-(j)}$ enable the following estimator of the aggregated main effect $\beta_{(l);j'-j}$ (defined in (2)) that is not biased by second-order effects (also see Bettonvil and Kleijnen 1997):

$$\widehat{\beta_{(l);j'-j}} = \frac{[w_{(l);(j)} - w_{(l);-(j)}] - [w_{(l);(j'-1)} - w_{(l);-(j'-1)}]}{4}. \tag{5}$$

Consequently, the estimator of the individual main effect $\beta_{(l);j}$ that is not biased by second-order effects, is

$$\widehat{\beta_{(l);j}} = \frac{[w_{(l);(j)} - w_{(l);-(j)}] - [w_{(l);(j-1)} - w_{(l);-(j-1)}]}{4}. \tag{6}$$

Note that this bias elimination in (5) and (6) doubles the number of simulation observations, because it implies mirror observations.

Let the symbol "$l \to l'$" in a subscript mean that the output l' is observed "for free" when we are interested in output l; i.e., running the simulation model to observe output l also gives an observation on the other output l'. For example, $w_{(1\to2);(K)}$ denotes the output w_2 when all K inputs are at $H_{(1)}$; $w_{(1\to2);-(K)}$ denotes the output of w_2 when all K inputs are at $L_{(1)}$. Therefore, $w_{(1\to2);(K)}$ and $w_{(1);(K)}$ are observed for the same input combination $H_{(1);1-K}$. This gives the following definition.

Definition 4 *Let* $w_{(l \to l');(j)}$ *denote output* l' *when inputs* 1 *through* j *are at* $H_{(l)}$ *and the remaining inputs are at* $L_{(l)}$*; likewise, the mirror output* $w_{(l \to l');-(j)}$ *denotes output* l' *when inputs* 1 *through* j *are at* $L_{(l)}$ *and the remaining inputs are at* $H_{(l)}$.

Next we give the following definition of a batch of inputs such that there is no cancellation of individual effects within the batch; e.g., Table 2 gave an example of two batches with batch 1 containing inputs 1 through k_1 so both outputs increase and batch 2 containing inputs $k_1 + 1$ through K so output 1 increases and output 2 decreases.

Definition 5 *A* batch *is a group (of inputs) in which each of the* n *outputs either increases or decreases when changing all the individual inputs in this group from -1 to 1.*

The following theorems and their corollaries represent the main contribution of this paper; their proofs are given in the appendixes, and examples are given in the next subsections.

Theorem 6 *If inputs* j' *through* j *are in the same batch and they have the same signs for outputs* l *and* l'*, then the unbiased estimators of the group main effects for outputs* l *and* l' *are*

$$\widehat{\beta_{(l);j'-j}} = \frac{[w_{(l);(j)} - w_{(l);-(j)}] - [w_{(l);(j'-1)} - w_{(l);-(j'-1)}]}{4} \tag{7}$$

and

$$\widehat{\beta_{(l');j'-j}} = \frac{[w_{(l \to l');(j)} - w_{(l \to l');-(j)}] - [w_{(l \to l');(j'-1)} - w_{(l \to l');-(j'-1)}]}{4} \tag{8}$$

where $j' \leq j$ *and corresponding terms in (7) and (8) are observed for the same input combination.*

Note that (7) is identical to (5). An example of corresponding terms in (7) and (8) is $w_{(l);(j)}$ and $w_{(l \to l');(j)}$. The proof of this theorem is given in Appendix 1.

Corollary 7 *If* w_l *and* $w_{l'}$ *either increase or decrease for the individual input* j*, then the unbiased main effect estimators are*

$$\widehat{\beta_{(l);j}} = \frac{[w_{(l);(j)} - w_{(l);-(j)}] - [w_{(l);(j-1)} - w_{(l);-(j-1)}]}{4} \tag{9}$$

$$\widehat{\beta_{(l');j}} = \frac{[w_{(l \to l');(j)} - w_{(l \to l');-(j)}] - [w_{(l \to l');(j-1)} - w_{(l \to l');-(j-1)}]}{4} \tag{10}$$

Proof. Equations (9) and (10) follow from (7) and (8) when j' equals j. ∎

Now we give a theorem if the batch of inputs has *opposite* signs for the outputs l and l' (instead of the same signs as in the preceding theorem).

Theorem 8 *If inputs j' through j are in the same batch, and they have opposite signs for outputs l and l', then the unbiased estimators of the group main effects for output l and l' are*

$$\widehat{\beta_{(l);j'-j}} = \frac{[w_{(l);(j)} - w_{(l);-(j)}] - [w_{(l);(j'-1)} - w_{(l);-(j'-1)}]}{4} \tag{11}$$

and

$$\widehat{\beta_{(l');j'-j}} = -\frac{[w_{(l\to l');(j)} - w_{(l\to l');-(j)}] - [w_{(l\to l');(j'-1)} - w_{(l\to l');-(j'-1)}]}{4} \tag{12}$$

where $j' \leq j$.

Note the minus sign in (12) immediately after the equality sign. The proof of this theorem is given in Appendix 2.

Corollary 9 *For an individual input that makes one output increase and the other output decrease, the unbiased main-effect estimator is*

$$\widehat{\beta_{(l);j}} = \frac{[w_{(l);(j)} - w_{(l);-(j)}] - [w_{(l);(j-1)} - w_{(l);-(j-1)}]}{4} \tag{13}$$

$$\widehat{\beta_{(l');j}} = -\frac{[w_{(l\to l');(j)} - w_{(l\to l');-(j)}] - [w_{(l\to l');(j')} - w_{(l\to l');-(j')}]}{4} \tag{14}$$

Proof. Equations (13) and (14) follow from (11) and (12) when j' equals j. ■

2.4 MSB for Two Output Types

In this subsection, we concentrate on MSB in the simple case of only two types of outputs (so $n = 2$). Actually, most cases with multiple outputs have only two output types; see again Chan and Spedding (2001), Kumar and Nottestad (2006), Shang et al. (2004), and Yalcnkaya and Mirac Bayhan (2009).

Table 2 has already illustrated a situation in which the individual inputs 1 through k_1 in batch 1 increase both outputs; the inputs $k_1 + 1$ through K in batch 2 increase output 1 and decrease output 2. Table 3 details the MSB procedure for this situation; we shall detail the SPRT in Section 2.5.

In Appendix 3 we detail two special cases, each with a single batch of inputs so $k_1 = K$; namely, (i) each input makes both outputs increase, and (ii) each input makes one output increase and the other output decrease. In Appendix 4 we sketch MSB for the general case of $n > 2$ outputs

2.5 MSB: Replicates and SPRT

To *test* the importance of the estimated main effects (of groups of inputs or individual inputs), we follow Wan et al. (2010). They give a testing procedure

Table 3: MSB for Case 3

(1) Define the values of all K inputs such that changing each individual input from $L_{(1)}$ to $H_{(1)}$ makes w_1 increase; the $(1-k_1)$ inputs in batch 1 make w_2 increase, and the (k_1+1-K) inputs in batch 2 make w_2 decrease.

(2) Use SPRT with initial sample size $N_{0;1-k_1}$ to find the number of replicates m_{1-k_1} where k_1 is not necessarily a power of two:

$$\begin{pmatrix} w_{(1);(k_1);r} & w_{(1);-(k_1);r} & w_{(2);(k_1);r} & w_{(2);-(k_1)r} \\ w_{(1\to 2);(k_1);r} & w_{(1\to 2);-(k_1);r} & w_{(2\to 1);(k_1);r} & w_{(2\to 1);-(k_1);r} \end{pmatrix}.$$

Estimate $(\widehat{\beta_{(1);1-k_1}}, \widehat{\beta_{(2);1-k_1}})'$ and $(\widehat{\beta_{(1);k_1+1-K}}, \widehat{\beta_{(2);k_1+1-K}})'$.

For batch 1:

(a) If SPRT declares $(\overline{\widehat{\beta_{(1);1-k_1}}}, \overline{\widehat{\beta_{(2);1-k_1}}})'$ unimportant, then discard batch 1;

(b) else split batch 1 into two batches.

For batch 2:

(a) If SPRT declares $(\overline{\widehat{\beta_{(1);k_1+1-K}}}, \overline{\widehat{\beta_{(2);k_1+1-K}}})'$ unimportant, then discard batch 2;

(b) else split batch 2 into two batches.

...

Final: Use SPRT to identify the important individual inputs, and estimate their main effects.

that is meant to control the type-I (or α) and type-II (or β) error probabilities. However, we think that their procedure does not guarantee control of these probabilities over the whole procedure with its sequence of steps (also see De and Baron (2012) for an interesting discussion of so-called familywise error probabilities, in the context of clinical testing). We therefore consider Wan et al.'s SB and our MSB as heuristics (which are better than *apriori* assuming that the majority of potentially important individual inputs are unimportant, and experimenting with a small group of inputs that are subjectively assumed to be important).

Like Wan et al. we assume that the simulation outputs $w_{(l)}(\mathbf{x})$ for input combination $\mathbf{x}$ have a Gaussian marginal distribution with heterogeneous variances $\sigma^2_{(l)}(\mathbf{x})$; moreover, the four input combinations in Theorems 1 and 2 may use common random numbers (CRN).

Note that CRN are meant to reduce the variances of the estimated effects; see Law (2007)'s textbook on discrete-event simulation. To implement CRN, we make replicate (say) r use pseudo-random number (PRN) seed or initial value (say) v_r where $r = 1, ..., m_{j'-j}$ with $m_{j'-j}$ denoting the number of replicates when estimating $\beta_{(l);j'-j}$; we use a vector of seeds $\mathbf{v}_r$ if the simulation needs more than one seed (e.g., the simulation may select different seeds for different processes such as different service stations). These seeds should be selected such that they ensure that the PRN streams do not overlap; i.e., the seeds should generate replicates that give independent and identically distributed (IID) simulation outputs. Software such as Arena can easily satisfy these seed requirements; see Kelton, Sadowski, and Sturrock (2007). The required number of replicates tends to increase, as the group size decreases (see Figure

2 below). So as we proceed from stage to stage, we may use the same CRN for the first replicates that the new stage has in common with its immediately preceding stage, and use new PRN seeds for the additional replicates in the new stage. Consequently, replicate 1 uses the same seed $\mathbf{v}_1$ in all stages, etc. If we do not use CRN as much as possible, we still get correct results; actually, we do not use CRN in our Monte Carlo experiments, as explained in Section 3.

Instead of applying the classic Student t statistic, Wan et al. derive a test based on Wald (1947)'s *SPRT* (the latter test is also discussed by Kleijnen 1987, pp. 54-55 in a simulation context, and recently by De and Baron 2012). In general, SPRTs add one replicate at a time, and terminate as soon as a conclusion can be reached. Wan et al. apply their SPRT each time when they test a group effect (in the early stages) or an individual effect (in the final stage). Their SPRT adds one replicate at a time to the four groups being tested, and may use CRN (also see our Theorems 1 and 2). Let r denote the current number of replicates when estimating $\beta_{j'-j}$ (sum of main effects of inputs j' through j; we focus on a single output for the time being). The initial number of replicates—*initial sample size*—is N_0. Wan et al. select a value for N_0 that remains constant over all the stages of their SB; e.g., $N_0 = 25$ in their Monte Carlo study and their semiconductor case-study. We, however, conjecture that N_0 may be smaller in the early stages because those stages estimate the sum of the (positive) main effects of bigger groups (from four observations, except for the very first stage—called stage 0—which uses only two different observations) so the signal-noise ratio is larger (we shall detail our rule for selecting the initial sample size, in Section 3).

SPRT uses the estimated variance of the estimator of $\beta_{(l);j'-j}$ based on the initial sample $N_{0;j'-j}$:

$$S^2_{(l);j'-j} = \frac{\sum_{r=1}^{N_{0;j'-j}} (\widehat{\beta_{(l);j'-j;r}} - \overline{\widehat{\beta_{(l);j'-j}}})^2}{N_{0;j'-j} - 1} \text{ with } \overline{\widehat{\beta_{(l);j'-j}}} = \frac{\sum_{r=1}^{N_{0;j'-j}} \widehat{\beta_{(l);j'-j;r}}}{N_{0;j'-j}}.$$

An illustration of the SPRT procedure for two outputs is Figure 1, in which the two triangles (formed by the solid lines and the dotted lines) are the continuation regions for the $n = 2$ output types. The symbols $\bullet$ and $\blacktriangle$ represent the observed value of the test statistics for the two outputs as a function of the number of replicates (this plot shows decreasing values for output 1, and increasing values for output 2). The SPRT checks whether the statistic $\sum_{r=1}^{m_{j'-j}} [\widehat{\beta_{(l);j'-j;r}} - r_{(l);0;j'-j}]$ with drift parameter $r_{(l);0;j'-j}$ (see below) crosses one of their termination boundaries (TB). TB1 denotes the boundary of the region in which the effect is declared to be unimportant; TB2 denotes the boundary of the region in which the effect is declared to be important. If for one of the outputs its TB2 is crossed, the group is declared to be important; only if both outputs cross TB1, the group is declared unimportant. Compared with SB for a single output, MSB has a higher probability of declaring a group to be important. The maximum number of observations for estimating $\beta_{(l);j'-j}$ is one more than $M_{(l);j'-j}$; in general, $M_{(l);j'-j} \neq M_{(l');j'-j}$ where $l \neq l'$. The final number of replicates when estimating $\beta_{(l);j'-j}$ is $m_{j'-j}$. The triangular region is defined by

- the intercepts $\pm a_{(l);j'-j} = \pm a_{(l);0;j'-j} S^2_{j'-j}$, and
- the slopes of the sides of the triangular region $\pm\lambda_{(l)} = \pm(\Delta_{(l);1} - \Delta_{(l);0})/4$.

The constants $a_{(l);0;j'-j}$ and $r_{(l);0;j'-j}$ are the solutions of rather complicated equations given by Wan et al.; see their equations 5 and 6 and the Matlab code in Appendix C in their Online Supplement available at

http://joc.pubs.informs.org/ecompanion.html.

However, we correct an error in this code; i.e., we add their SumInt function, which is missing in their Online Supplement. Wan et al. state that the goal of their SB is to classify those inputs with $\beta_j \leq \Delta_0$ as "unimportant" and those inputs with $\beta_j > \Delta_0$ as important. For those inputs with $\beta_j \leq \Delta_0$, they want to control the Type-I error probability of declaring them important not to exceed α; for $\beta_j \geq \Delta_1$, they want the statistical power of the test to be not smaller than γ. For $\Delta_0 < \beta_j < \Delta_1$, they want "reasonable" power. Notice that the slopes of the triangle increase as Δ_1 increases; i.e., we need fewer replicates when we estimate bigger effects. The number of observations for the four input combinations when estimating $\beta_{j'-j}$ are equal if CRN are used, before beginning the test (as detailed by Wan et al., near the end of their Section 3.2).

Because our MSB considers $n \geq 2$ output types, we use *Bonferroni's inequality* and replace the type-I error probability α in this SPRT by α/n and the type-II error probability $1-\gamma$ by $(1-\gamma)/n$. This change in the SPRT implies a bigger triangular area in Figure 1 in which we continue sampling before accepting either H_0 stating that the group factor is unimportant for all output types (i.e., the group-factor is not important for any output type) or H_1 stating that the group-factor is important for one or more output types.

Obviously, the $m_{j'-j}$ replicates enable the computation of $\overline{\widehat{\beta_{(l);j'-j}}}$, which is the average of $\widehat{\beta_{(l);j'-j;r}}$ (the estimated effect for output l of the group of inputs j' through j in replicate r with $r = 1, \ldots, m_{j'-j}$).

2.6 MSB Validation

By definition, "screening" means that K (number of inputs) is too big to enable the estimation of all the individual effects of a second-degree polynomial; this number of effects is (say) $q(K) = 1+K+K+K(K-1)/2$ (likewise, the estimation of all the parameters of a Kriging metamodel would be problematic). The case study in Section 4 is a relatively small screening example with $K = 26$ so $q(26)$ = 378. Unlike Wan et al. (2010, pp. 489-491) we do not to use a central composite design (CCD) based on a resolution-V (R-V) design for all K inputs (our approach resembles the approach in Bettonvil and Kleijnen 1996, p. 187-189). The final result of MSB and SB are estimates of only (say) $K1$ ($<< K$) first-order effects of the inputs declared to be "important". This result is based

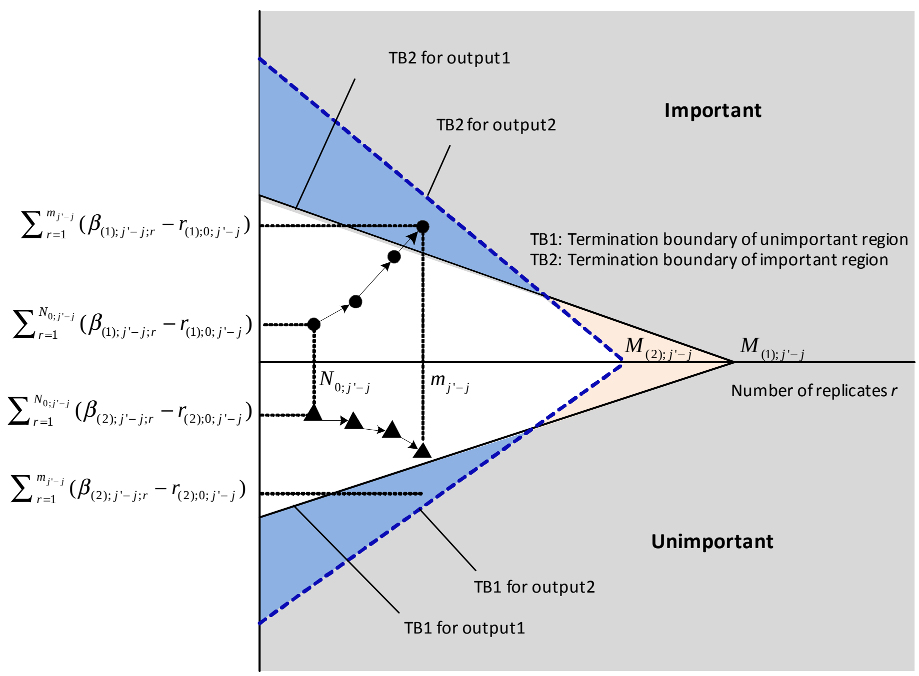

Figure 1: SPRT test for $n = 2$ output types

on the three major *assumptions* specified in Section 2.2; namely, (i) a second-order polynomial is a valid metamodel; (ii) the signs of all first-order effects in this polynomial are known; (iii) the heredity property holds. To validate these assumptions, we first estimate the $q(K1) = (1 + K1 + K1 + K1(K1 - 1)/2)$ individual effects of the second-degree polynomial with $K1$ inputs (e.g., $q(5) = 21 << q(26) = 378$). This estimation is not a screening problem, but a "classic" design of experiments (DOE) problem. This classic design uses a CCD including a R-V design (such a design, however, is not saturated at all; i.e., this design implies a number of input combinations n_{CCD} that is much higher than $q(K1)$; various alternative designs are discussed in Kleijnen 2008, p. 51). We shall use such a classic CCD (with $n_{CCD} = 43 >> q(5) = 21$) for the case study in Section 4.

When running the simulation with the $K1$ inputs declared important by MSB, we also need values for all the $(K - K1)$ unimportant inputs. We decide to keep the unimportant inputs constant. The unimportant quantitative inputs we keep at their coded value 0; the unimportant qualitative inputs we keep at (say) +1. We also need to select the number of replicates for the CCD (say) m_{CCD}.

Moreover, we should verify whether the $(K - K1)$ inputs declared to be *unimportant* by the screening method are indeed unimportant. We decide to select (say) n_{val} combinations of the K inputs (unimportant or important). Our selection of n_{val} depends on the computer time required per replicate and the computer budget. We select these n_{val} combinations such that we obtain a space-filling design for the quantitative inputs (important or unimportant); i.e., we use a Latin hypercube sample (LHS); for more details on LHS we refer to Kleijnen (2008, pp. 126-130). For a qualitative input j we sample without replacement its -1 and 1 values with equal probabilities of 0.5 (so $\Pr(x_j = -1) = \Pr(x_j = 1) = 0.5$) such that $n_{val}/2$ values are -1 (and the other $n_{val}/2$ values are 1). We randomly combine the n_{val} combinations of the quantitative inputs with the n_{val} values of the qualitative inputs.

We then simulate these n_{val} input combinations, using m_{val} replicates; to select m_{val}, we should examine m_j (the SPRT's final number of replicates needed to test the significance of individual inputs; see again Figure 1 in Section 2.5 and also Figure 2 in the next section). We might use CRN in these n_{val} input combinations.

Next, we *test* the validity of the estimated second-degree polynomial with the parameters $\widehat{\boldsymbol{\beta}}_{(l)}$ for the $K1$ important inputs, which implies that the remaining $K - K1$ unimportant inputs have zero first-order and second-order effects. We therefore predict output of type l for the n_{val} input combinations, and compare these regression predictions $\widehat{y}_{(l);i}$ $(i = 1, \ldots, n_{val})$ with the corresponding average simulated output values $\overline{w}_{(l);i} = \sum_{r=1}^{m_{val}} w_{(l);i;r}/m_{val}$. Because MSB declares an input j to be important if $\beta_{(l);j} \geq \Delta_{(l);1}$, the regression predictor $\widehat{y}$ uses more important inputs (and a bigger CCD) as Δ_1 decreases so its fit tends to increase. We therefore accept the regression predictor as valid if $\left|\overline{w}_{(l);i} - \widehat{y}_{(l);i}\right| \leq \Delta_{(l);1}$; this comparison is scale dependent. A Studentized statistic is scale-independent because it accounts for the noise $\widehat{var}(\overline{w}_{(l);i}) =$

$\sum_{r=1}^{m_{val}}(w_{(l);i;r} - \overline{w}_{(l);i})^2/\{m_{val}(m_{val}-1)\}$ and $\widehat{var}(\widehat{y}_{(l);i})$. Before presenting this statistic in (16), we need to discuss $\widehat{var}(\widehat{y}_{(l);i})$.

Obviously, $\widehat{y}_{(l);i} = \mathbf{x}_i'\overline{\overline{\boldsymbol{\beta}}}_{(l)}$ where $\mathbf{x}_i$ denotes the vector of important inputs determined by the CCD and $\overline{\overline{\beta}}_{(l);j} = \sum_{r=1}^{m_{CCD}} \widehat{\beta}_{(l);j;r}/m_{CCD}$ (where $\widehat{\beta}_{(l);j;r}$ denotes effect j for output l computed from replicate r with $l = 1, \ldots, n$, $j = 1, \ldots, q(K1)$, $r = 1, \ldots, m_{CCD}$). Hence

$$\widehat{var}(\widehat{y}_{(l);i}) = \mathbf{x}_i'\widehat{\mathbf{cov}}(\overline{\overline{\boldsymbol{\beta}}}_{(l)})\mathbf{x}_i. \quad (15)$$

To compute $\widehat{\mathbf{cov}}(\overline{\overline{\boldsymbol{\beta}}}_{(l)})$, we do not use the classic formula which assumes that the simulation outputs of type l have a constant variance $\sigma^2_{(l)}$ and that the n_{CCD} combinations do not apply CRN; instead we use the m_{CCD} replicates to estimate the (co)variances between $\widehat{\beta}_{(l);j}$ and $\widehat{\beta}$:

$$\widehat{cov}(\widehat{\beta}_{(l);j}, \widehat{\beta}_{(l);j'}) = \frac{\sum_{r=1}^{m_{CCD}} (\widehat{\beta}_{(l);j;r} - \overline{\overline{\beta}}_{(l);j})(\widehat{\beta}_{(l);j';r} - \overline{\overline{\beta}}_{(l);j'})}{m_{CCD} - 1} \; (j, j' = 1, \ldots, q(K1)).$$

These estimators give the $q(K1) \times q(K1)$ matrix $\widehat{\mathbf{cov}}(\overline{\overline{\boldsymbol{\beta}}}_{(l)}) = (\widehat{cov}(\widehat{\beta}_{(l);j}, \widehat{\beta}_{(l);j'})/m_{CCD})$, which we use in (15).

To validate the regression (meta)model, we use the following (Studentized) statistic with ν degrees of freedom (also see Kleijnen 2008, p. 58):

$$t_{(l);i;\nu} = \frac{\max(\left|\overline{w}_{(l);i} - \widehat{y}_{(l);i}\right| - \Delta_{(l);1}, 0)}{\sqrt{\widehat{var}(\overline{w}_{(l);i}) + \widehat{var}(\widehat{y}_{(l);i})}} \; (i = 1, \ldots, n_{val}). \quad (16)$$

Because the two variables $\overline{w}$ and $\widehat{y}$ have different variances, the correct degrees of freedom ν of this Student t-statistic is not so easy to determine (so-called Behrens-Fisher problem). We select $\nu = \min(m_{CCD}-1, m_{val}-1)$. Because there are n_{val} t-statistics for output l ($l = 1, \ldots, n$), we use Bonferroni's inequality; i.e., we replace the classic α value by $\alpha/(n_{val} \times n)$. If none of these t-statistics exceeds the critical value (say) $t_{m-1}(\alpha/(n_{val} \times n))$ (or $\max_{l;i} t_{(l);i;\nu} \leq t_{m-1}(\alpha/(n_{val} \times n))$), then we accept the metamodel. Next, we may use the n_{val} observations $\overline{w}_i$ ($i = 1, \ldots n_{val}$) to re-estimate the regression parameters $\boldsymbol{\beta}_{(l)}$; we expect that the resulting new estimate does not deviate much from the old estimate.

If this polynomial is accepted as valid, we test the remaining two assumptions; namely, known signs of all first-order effects in this polynomial, and heredity. If this validation procedure suggests that the MSB assumptions do not hold, then we need to look for a different screening method; see again Kleijnen (2008) and Kleijnen (2009). Details follow in Section 4. Note that the second-order polynomial for the important inputs can be used after the screening phase, to optimize these inputs through response surface methodology (RSM).

3 Monte Carlo Experiments

To evaluate the performance of our MSB heuristic, we first use a Monte Carlo laboratory (we shall use a case-study in Section 4). The reason is that MSB is based on three specific assumptions given in Section 2.2. We should therefore start our evaluation with situations that do satisfy these assumptions; a "laboratory" can fully satisfy the assumptions. (Wikipedia states: "A laboratory (lab) is a facility that provides controlled conditions in which scientific research experiments, and measurement may be performed".)

Note that case -studies (real-world applications) enable us to study the "robustness" of the MSB method; i.e., how well does the method perform if not all its assumptions are completely satisfied? Before we perform such robustness studies, we should examine the performance if all assumptions do hold. Moreover, realistic applications may be computationally expensive; i.e., a single simulation run may take hours or days; in the Monte Carlo lab, however, a "simulation" run (an observation) takes only (micro)seconds (depending on the computer hardware and software).

MSB is a "black-box" method; i.e., it selects a combination of the K inputs $\mathbf{x} = (x_1, \ldots x_K)'$, and observes the resulting multi-variate simulation output $\mathbf{w}$ $= (w_1, \ldots, w_n)'$ (next, MSB uses all available I/O data to estimate the group effects, etc.). Our Monte Carlo lab, however, is a "white box"; i.e., we select specific values for the coefficients of the second-order polynomial in (1) and the variances of the replicates (which equal the variances of the residuals); moreover we make the replicates normally independently and identically distributed (NIID).

3.1 Wan et al.'s Experiment, and a New Pilot-sample Rule

Wan et al. (2010) also use a Monte Carlo lab to evaluate their SB, but they consider a single output type and $K = 10$, whereas we also consider two output types and $K = 100$. Because the selection of parameter values in a Monte Carlo experiment is virtually unlimited, we follow Wan et al. as closely as we consider acceptable for our MSB; this leads us to the following Monte Carlo experiments.

Like Wan et al. (p. 488) we call inputs "unimportant" if they have main effects not exceeding $\Delta_0 = 2$ and "important" if they are at least $\Delta_1 = 4$; the residuals ϵ_l are normally distributed with mean zero and a standard deviation equal to 1 + $|E(w_l)|$. Note that these rather big standard deviations require many replicates (as will be illustrated by Table 4 and Figure 2). Furthermore, like Wan et al. we select the type I error probability $\alpha = 0.05$ and power $\gamma = 0.90$; the two-factor interactions are randomly generated from a normal distribution with mean zero and variance four; we assume that Wan et al. implicitly select all purely quadratic effects to be zero. In Monte Carlo experiments with additive noise ϵ_l, CRN would generate a linear correlation coefficient with value 1; therefore we do not use CRN in these experiments. Unlike Wan et al. we allow $n = 2$ outputs (instead of a single output); we assume that half of the important inputs affect both outputs, one quarter of the other important inputs affect only output 1, and the other quarter affect only output 2. To create more than one batch of inputs, we assume that the signs of the latter quarter is negative. Like Wan et al. we select the performance measure to be the estimated probability of declaring an individual input to be important, $\widehat{\Pr}(DI)$. We also use 1000 macroreplicates (which by definition use non-overlapping PRN streams and give IID results), and we resample the values for the two-factor interactions.

Wan et al. experiment with $K = 10$ inputs (which is not a typical value in screening). They consider three cases, but we examine neither their case with all main effects exactly zero nor their case with all main effects equal to Δ_0 = 2; we do examine their case with five main effects between Δ_0 and $\Delta_1 = 4$ and five main effects exceeding Δ_1 but not exceeding the value 6: their values are 2, 2.44, 2.88, 3.32, 3.76, 4.2, 4.64, 5.08, 5.52, and 6 (also see Table 4). We illustrate Wan et al.'s SPRT for a Monte Carlo example with a single output (our SPRT for multiple outputs equals the SPRT for a single output with the type I and type II error probabilities changed using Bonferroni's inequality; moreover, an input is important if it has a significant main effects for at least one output). Our Table 4 presents results for a fixed initial sample size $N_0 = 5$ or $N_0 = 25$ ($N_0 = 25$ is selected by Wan et al.), and (see the last four columns) N_0 that is either 5 or 25 in the first stage and either 25% or 50% of the final number of replicates in the immediately preceding stage. The results show that the selection of N_0 does not seriously affect $\widehat{\Pr}(DI)$.(estimated probability of declaring a main effect important). The last line of this table shows that a fixed N_0 (column 3 or 4) requires more replicates than our variable N_0 (the number of replicates are added over all stages, like Wan et al. do). Some details of this sample-size selection are shown in Figure 2, discussed next.

Figure 2 details initial sample sizes that are not fixed——except in the very first stage where all $K = 10$ inputs are either -1 or 1 and $N_0 = 5$— but are 25% of the final number of replicates in the immediately preceding stage. We show results for macro-replicate 1. The initial number in the first stage is $N_{0;1-10} =$ 5 and this stage ends with $m_{1-10} = 39$. In the next stage, SB splits the total group of 10 factors into $2^3 =$ 8 and the remaining 2 factors (to increases the efficiency, SB and MSB select the number of inputs for the first new subgroup to be a power of two, wand the remaining inputs form the second subgroup). The initial number of replicates is 25% of 39 so this number is 10 after rounding to the next integer. This stage ends with 214 replicates. Note that the final number of replicates tends to increase as the group size decreases (from stage to stage the signal/noise ratio decreases). The conclusion of this subsection is that the new pilot-sample rule increases the efficiency of SB and MSB.

Table 4: $\widehat{Pr}(DI)$ and number of replicates in SB for constant N_0 and variable $N_{0;j'-j}$

Input no. j	Effect β_j	$N_0 = 5$	$N_0 = 25$	$(5, 25\%)$	$(5, 50\%)$	$(25, 25\%)$	$(25, 50\%)$
1	2.00	0.01	0.00	0.00	0.00	0.00	0.00
2	2.44	0.15	0.10	0.17	0.13	0.14	0.12
3	2.88	0.37	0.34	0.38	0.40	0.36	0.38
4	3.32	0.65	0.62	0.69	0.69	0.68	0.57
5	3.76	0.83	0.84	0.84	0.91	0.80	0.76
6	4.20	0.96	0.95	0.96	0.95	0.97	0.96
7	4.64	0.98	0.98	0.97	0.98	0.98	0.99
8	5.08	1.00	1.00	1.00	1.00	1.00	1.00
9	5.52	1.00	1.00	1.00	1.00	1.00	1.00
10	6.00	1.00	1.00	1.00	1.00	1.00	1.00
No. of replicates		21,798	19,793	15,203	14,875	16,860	15,048

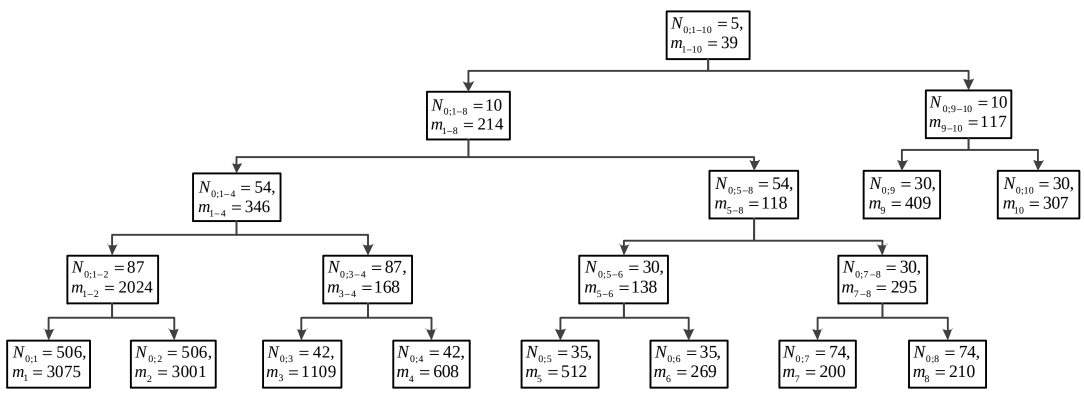

Figure 2: Initial (N_0) and final (m) number of replicates in macro-replicate 1, when N_0 is 5 in the first stage and 25% of m in the preceding stage

3.2 Experiment with $K = 100$ Inputs and Four Problem Characteristics

We also study the screening problem with $K = 100$ inputs, $n = 2$ outputs, and $q = 2$ batches. We compare SB—applied per output—and MSB. We select the thresholds $\Delta_0 = 2$ and $\Delta_1 = 4$. As initial number of replicates in the first stage we select $N_{0;1-100} = 5$, and as the initial number of replicates in the next stages we select 25% of the final number of replicates in the immediately preceding stage (for $K = 10$ see again Figure 2). The type-I error rate α is 0.05 and the power γ is 0.9. Because there are two outputs, we use Bonferroni's inequality and replace α by $\alpha/2 = 0.025$ and the type-II error rate $1 - \gamma$ by $(1 - \gamma)/2 = 0.05$. We use our lab to evaluate the performance effects of the following four problem characteristics (also see Table 5; the last two columns are discussed below):

1. *Sparsity* of effects; i.e., we select either 4 or 8 of the 100 main effects to be "important".

2. *Signal-noise ratio*; the higher the noise is, the more replicates should be obtained. We set the standard deviation of $\epsilon_{(l)}$ to either 5 or 10.

3. *Variability* of effects; i.e., we make either all important main effects to have the same value $|\beta_{(l);j}| = 5$ or we make them all different; namely, -5, -2, 2, 5 (so $|\beta_{(l);j}| = 2, 5$) when there are four important inputs (see characteristic 1), and $|\beta_{(l);j}| = 2, 3, 4, 5$ when there are eight important inputs.

4. *Clustering* of effects; the more clustered the individual important main effects are, the more efficient SB and MSB are. When there are four important inputs and they are clustered, then the important inputs are 1, 2, 99, and 100, and the non-clustered inputs are 1, 10, 91, and 100; when there are eight important inputs, then the clustered inputs are 1, 2, 3, 4, 97, 98, 99, 100, and the non-clustered inputs are 1, 10, 20, 30, 71, 81, 91, and 100.

Because we experiment with two levels per characteristic, there are 16 combinations. These combinations are shown in Table 5 (first thee columns). In all 16 combinations there are $q = 2$ batches: the $\pm$ signs in this table mean that all important main effects are positive for output 1 and some important main effects are negative for output 2 (e.g., in combination 1 the four important inputs 1, 2, 99, and 100 have positive main effects for output 1, but the inputs 99 and 100 have negative main effects for output 2).

We use 1,000 macroreplicates, and report the average *number of replicates* per stage, which quantifies the *efficiency*; see the last two columns of this table. Both MSB and SB require increasing number of replicates for higher number of important inputs, variability of effects, noise of simulation outputs, and clustering of important inputs. For example, combination 14—with $\sigma = 10$, eight important inputs with different values and even spread—requires the maximum number of replicates (namely, 1,723 for MSB), whereas combination 3 —with $\sigma = 5$, only four important inputs with the same value (namely, 5) and much clustering—requires the minimum number of replicates (namely, 119 for MSB). MSB requires only approximately half the number of replicates needed by SB; i.e., some input combinations applied for one output in MSB are again used when screening for the other output. Therefore, we conclude that MSB requires fewer input combinations and number of replicates than SB does.

Besides the efficiency, we also study the *efficacy* quantified through $\widehat{\Pr}(DI)$. We present $\widehat{\Pr}(DI)$ only for the combinations 2, 7, 9 and 14 (because we obtain similar results for the remaining 12 combinations); see Figure 3 where the x-axis gives $|\beta_{(l);j}|$ and the y-axis gives $\widehat{\Pr}(DI)$ (e.g., $|\beta_{(l);j}| = 0$, 2, and 5 in combination 2, and $|\beta_{(l);j}| = 0$, 2, 3, 4, and 5 in combination 9). Because

Table 5: MSB versus SB in Monte Carlo experiment

Combination	Scenario		Replicates	
	σ	Other three characteristics	MSB	SB
1	5	Inputs $(1, 2, 99, 100) = (2, 5, \pm 2, \pm 5)$	135	242
2	5	Inputs $(1, 10, 91, 100) = (2, 5, \pm 2, \pm 5)$	313	600
3	5	Inputs $(1, 2, 99, 100) = (5, 5, \pm 5, \pm 5)$	119	218
4	5	Inputs $(1, 10, 91, 100) = (5, 5, \pm 5, \pm 5)$	233	463
5	10	Inputs $(1, 2, 99, 100) = (2, 5, \pm 2, \pm 5)$	350	656
6	10	Inputs $(1, 10, 91, 100) = (2, 5, \pm 2, \pm 5)$	933	1,607
7	10	Inputs $(1, 2, 99, 100) = (5, 5, \pm 5, \pm 5)$	250	470
8	10	Inputs $(1, 10, 91, 100) = (5, 5, \pm 5, \pm 5)$	641	1,112
9	5	Inputs $(1, 2, 3, 4, 97, 98, 99, 100)$ $=(2, 3, 4, 5, \pm 2, \pm 3, \pm 4, \pm 5)$	178	354
10	5	Inputs $(1, 10, 20, 30, 71, 81, 91, 100)$ $=(2, 3, 4, 5, \pm 2, \pm 3, \pm 4, \pm 5)$	536	1,058
11	5	Inputs $(1, 2, 3, 4, 97, 98, 99, 100)$ $=(5, 5, 5, 5, \pm 5, \pm 5, \pm 5, \pm 5)$	145	290
12	5	Inputs $(1, 10, 20, 30, 71, 81, 91, 100)$ $=(5, 5, 5, 5, \pm 5, \pm 5, \pm 5, \pm 5)$	410	818
13	10	Inputs $(1, 2, 3, 4, 97, 98, 99, 100)$ $= (2, 3, 4, 5, \pm 2, \pm 3, \pm 4, \pm 5)$	464	922
14	10	Inputs $(1, 10, 20, 30, 71, 81, 91, 100)$ $= (2, 3, 4, 5, \pm 2, \pm 3, \pm 4, \pm 5)$	1,713	3,233
15	10	Inputs $(1, 2, 3, 4, 97, 98, 99, 100)$ $= (5, 5, 5, 5, \pm 5, \pm 5, \pm 5, \pm 5)$	319	620
16	10	Inputs $(1, 10, 20, 30, 71, 81, 91, 100)$ $= (5, 5, 5, 5, \pm 5, \pm 5, \pm 5, \pm 5)$	1,126	2,248

Note: "+" means positive effect on output 1, so $\beta_{(1)} > 0$
"−" means negative effect on output 2, so $\beta_{(2)} < 0$

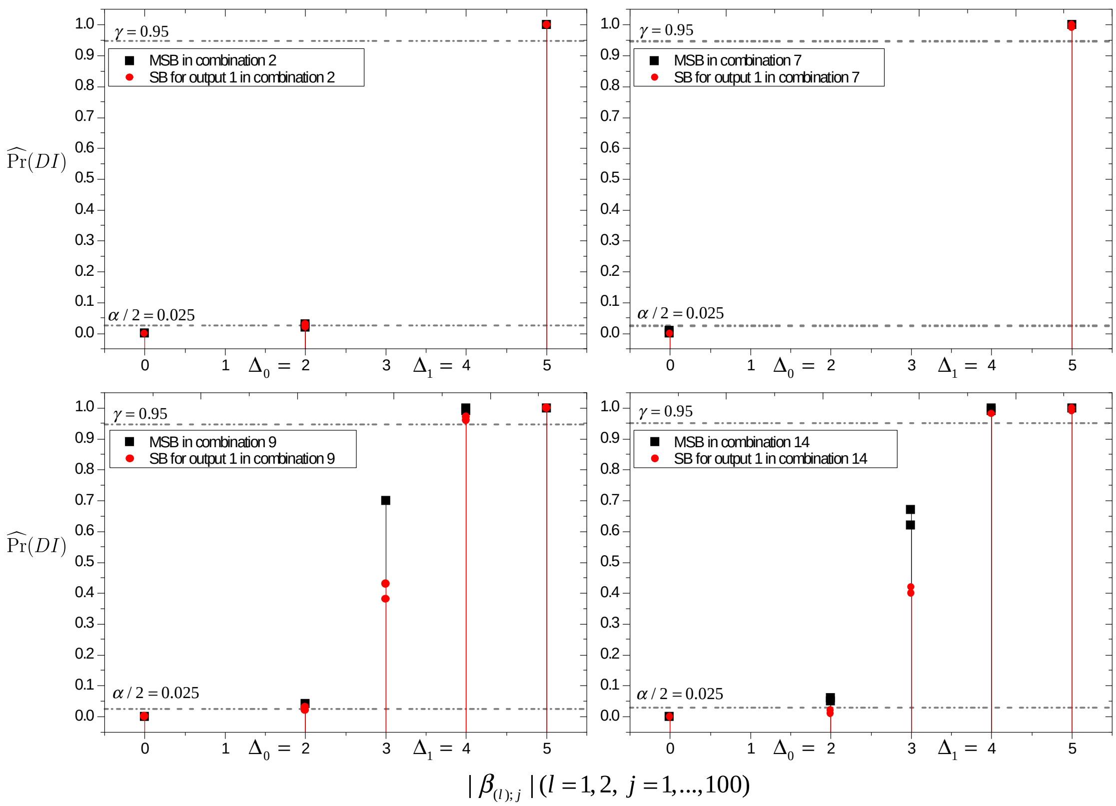

Figure 3: $\widehat{\Pr}(DI)$ of MSB and SB in combinations 2, 7, 9, and 14

$|\beta_{(1);j}| = |\beta_{(2);j}|$, we present results only for output 1. This figure shows that $|\beta_{(l);j}|$ has an important positive effect on $\widehat{\Pr}(DI)$. In all combinations, $\widehat{\Pr}(DI)$ lies in the interval $[0.025, 0.95]$ when $2 \leq |\beta_{(l);j}| \leq \Delta_1 = 4$, $\widehat{\Pr}(DI) = 0$ when $|\beta_{(l);j}| = 0$, and $\widehat{\Pr}(DI) = 1$ when $|\beta_{(l);j}| = 5$. So, both MSB and SB give the desired screening results with appropriate type-I error rates and power. However, MSB's $\widehat{\Pr}(DI)$ exceeds SB's $\widehat{\Pr}(DI)$ when $\Delta_0 \leq |\beta_{(l);j}| \leq \Delta_1$; e.g., in combination 9 (south-west corner of the figure), MSB's $\widehat{\Pr}(DI) = 0.7$ and SB's $\widehat{\Pr}(DI)$ is only 0.38 and 0.43 when $|\beta_{(l);j}| = 3$; also see combination 14 (south-eastern corner). Our explanation is that an input that is unimportant for one output has a chance to be important for the other output, so the probability of declaring this input to be important increases. The conclusion of this subsection is that MSB is more efficient and effective than SB.

4 Case Study

In this section we discuss a case study concerning a Chinese third-party logistics (TPL) company that wants to improve its "Just-In-Time" (JIT) system. This system uses the "TPL's milk-run pickup and cross-docking distribution" (TPL-MRCD); i.e., parts are first collected from routine milk-run suppliers as long as the TPL receives a pick order, and transferred to a regional cross-docking distribution center (CDDC), which consolidates parts and loads them onto large trucks for truck load (TL) delivery. Figure 4 shows a flow chart; IBT and OBT stand for inbound truck and outbound truck respectively. This figure depicts the flow of parts, truck scheduling, and door assignment. The TPL-MRCD is a pull system. Once the inventory level reduces to the reorder point, the assembly plants place a purchase order with the CDDC, which aggregates orders and places them with parts suppliers. In general, completed purchase orders from a given milk-run region will not create an entire truckload, but form a less-than-truckload (LTL). Therefore, parts collected from a milk-run pickup are first transferred to the CDDC, where parts are sorted per assembly plant, and then consolidated and loaded onto trucks for TL transportation. There are two transport types; namely, less-than-truckload (LTL) transport within the industrial zones, and TL transport, which often covers distances of more than 1,000 kilometers between CDDC and factory warehouse.

Given China's rapid economic growth, Feng et al. (2010) estimate that the Chinese car market will grow 10% to 15% over the next decade. To satisfy this growing demand, the joint venture served by the TPL expects to open another assembly plant. When this new plant becomes operational, 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 (30 days). Long 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 (say) 30 days. Therefore, our goal is to identify and improve critical logistic factors (identify important inputs), and to make the two performance measures (output) CT and NT satisfy desired values.

To solve this problem, we use Arena (namely, version 13.0 of Rockwell Automation); Arena supports the process-modeling paradigm. Because the production cycle of the assembly plants is a month, we treat our simulation model as a terminating simulation which runs for 30 simulated days after a warm-up period of 5 days.

The TPL-MRCD operates 16 hours per day; namely, from 8 AM until midnight. Its simulation model has 26 inputs that may affect the two outputs (CT and NT). (Actually, we could distinguish more than 26 inputs, but some of these extra inputs—such as number of suppliers—would require us to change the structure of the simulation model. Moreover, these extra inputs cannot be controlled by the TPL. Finally, the 26 inputs suffice to illustrate MSB.). The values of these 26 inputs are coded as -1 and 1; for CT these values are shown in Table 6, where SPT stands for "shortest processing time", and FIFO for

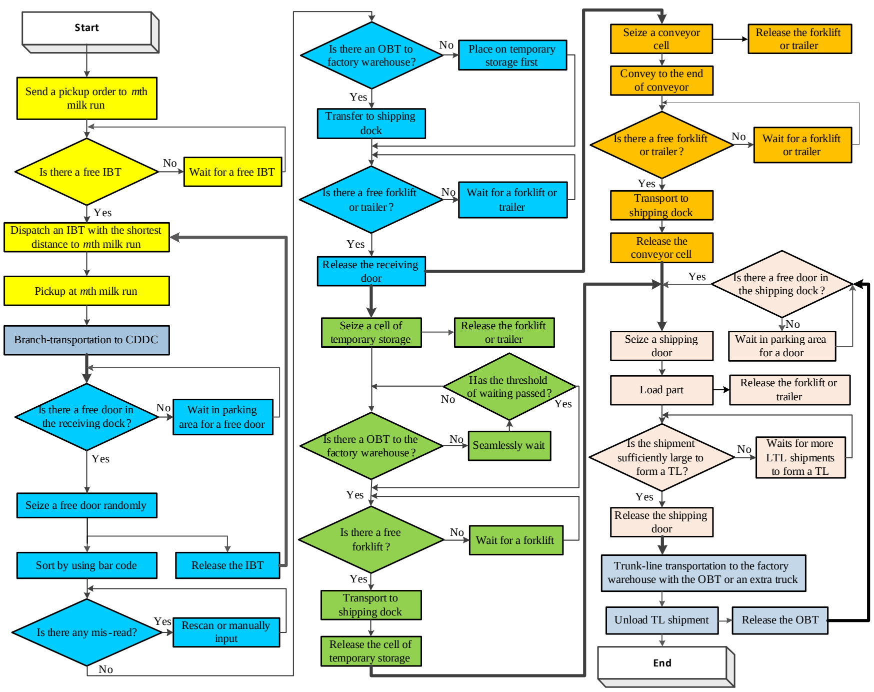

Figure 4: TPL-MRCD logic flow chart

"first-in, first-out" queue discipline. So, all inputs are quantitative, except for the queue discipline. This table shows that the inputs 1 through 5 have the same signs for the two outputs (CT and NT), because higher replenishment frequencies (the smaller value of inputs 1 through 5) lead to more parts in the TPL-MRCD system so NT increases; at the same time it may make logistic resources reach higher utilization, which increases queue length and delay so CT increases. The remaining 21 inputs have opposite signs for CT and NT; e.g., more receiving dock doors (input 14) decrease the parts waiting time in the CDDC so CT decreases, and increases the number of parts received by the assembly plants so NT increases. So in this case study we consider two batches; namely, batch 1 with inputs 1 though 5 and batch 2 with inputs 6 through 26.

Furthermore, we decide to set $\Delta_{(CT);0} = 2.5$ and $\Delta_{(NT);0} = 2{,}000$ as the minimum acceptable CT and NT values. We set $\Delta_{(CT);1} = 5$ and $\Delta_{(NT);1} = 3{,}000$ as the performance improvement that we do not want to miss. Inspired by Figure 2 for our Monte Carlo experiment, we select the initial number of replicates in the first stage $N_{0;1-26} = 5$, and the initial number of replicates in the next stages as 25% of the final number of replicates in the immediately preceding stage, but not smaller than 5. Because there are two outputs in this case study, we replace the type-I error rate $\alpha = 0.05$ and type-II error rate $\beta = 0.1$ by $\alpha/2 = 0.025$ and $\beta/2 = 0.05$. Figure 5 shows MSB results per stage, where shaded blocks denote important inputs. Altogether, MSB requires 233 replicates (namely, $m_{1-5} + m_{6-26} + \ldots + m_{21}$) to identify the five important inputs labeled 4, 5, 14, 17, and 20; the inputs 4 and 5 are in batch 1 (see the first left bifurcation) and inputs 14, 17, and 20 are in batch 2 (see the first right bifurcation). Figures 6 and 7 show that SB requires 238 and 117 replicates for CT and NT respectively, so altogether SB requires 355 (whereas MSB requires only 233) replicates. SB and MSB identify the same inputs as being important; SB identifies the inputs 4, 5, 14, 17, and 20 for CT and input 17 for NT. Note that MSB and SB do not use the same input combinations in every stage.

Finally, we *validate* the MSB assumptions. We use a CCD for these five important inputs (Wan et al. 2010 also use a CCD but for all 20 in their semiconductor case-study). This CCD includes a 2^5 full factorial design plus 10 $(= 2 \times 5)$ axial points $\pm\sqrt{5}$ and one central point (altogether 43 combinations). This CCD enables the estimation of the 21 $(= 1 + 5 + 5 + (5 \times 4)/2)$ individual effects in the second-degree polynomial. We obtain $m_{CCD} = 10$ replicates for each combination in the CCD, after considering the numbers in the last stages of Figure 5. The unimportant quantitative inputs we fix at their coded value 0; the one unimportant qualitative input we set to +1 (which denotes FIFO, the default queueing rule of the current TPL). We use CRN (the default in Arena).

Analysis of variance (ANOVA) of these CCD I/O data shows that the second-order polynomials for CT and NT have $R^2_{CT} = 0.9608$, $R^2_{CT}(adj) = 0.9519$, $R^2_{NT} = 0.9641$, $R^2_{NT}(adj) = 0.9588$, whereas the polynomials with main effects only have $R^2_{CT} = 0.7022$, $R^2_{CT}(adj) = 0.6683$, $R^2_{NT} = 0.6988$, and $R^2_{NT}(adj) = 0.6733$. So, the two second-order polynomials are significantly better, and we can use them to predict the outputs (i.e., Assumption 1 holds for the important inputs; also see equation 1).

Given that the two second-order polynomials for the five important inputs are valid, we now present their individual estimated coefficients:

$$\begin{aligned}\widehat{y}_{CT} &= 31 - 6.60x_4 - 11.23x_5 - 1.97x_{14} - 10.87x_{17} - 4.36x_{20} \\ &\quad -0.036x_4x_5 - 0.63x_4x_{14} - 0.039x_4x_{17} - 0.022x_4x_{20} + 0.65x_5x_{14} \\ &\quad -0.21x_5x_{17} + 1.15x_5x_{20} - 1.28x_{14}x_{17} + 1.49x_{14}x_{20} + 2.01x_{17}x_{20} \\ &\quad +1.39x_4^2 + 3.66x_5^2 + 5.92x_{14}^2 + 9.05x_{17}^2 + 0.29x_{20}^2 \qquad (17)\end{aligned}$$

and

$$\begin{aligned}\widehat{y}_{NT} &= 50040.33 - 985.71x_4 - 818.47x_5 + 1289.66x_{14} + 4493.64x_{17} + 1474.49x_{20} \\ &\quad -258.31x_4x_5 - 294.13x_4x_{14} - 606.87x_4x_{17} - 29.87x_4x_{20} - 716.75x_5x_{14} \\ &\quad +974.13x_5x_{17} - 371.37x_5x_{20} + 1004.94x_{14}x_{17} - 711.31x_{14}x_{20} - 693.81x_{17}x_{20} \\ &\quad +1167.85x_4^2 + 756.16x_5^2 - 2766.67x_{14}^2 - 3849.62x_{17}^2 - 27.17x_{20}^2. \qquad (18)\end{aligned}$$

These two equations show that the signs of the estimated main effects of the important inputs—displayed in (17) and (18)—confirm the signs assumed in Table 6; namely, inputs 4 and 5 have minus signs for both CT and NT, whereas inputs 14, 17, and 20 have opposite signs for these outputs. So we conclude that Assumption 2 (known signs of all main effects) holds for the important inputs. This assumption and assumption 3 (heredity) are now further examined.

To test that all first-order and second-order effects of all *unimportant* inputs are zero, we select $n_{val} = 10$ combinations through LHS in which we uniformly sample values between -1 and 1 for all 25 quantitative inputs, and we sample the two values -1 and 1 for the qualitative input 23 (details are given in Table 14 in the appendix). We decide to obtain $m_{val} = 20$ replicates for each combination (we also show results for only 10 replicates per combination in Appendix 5, Table 15). These 10 combinations with their 20 replicates give the simulated $\overline{w}$ and the predicted $\widehat{y}$, and their estimated variances $\widehat{var}(\overline{w})$ and $\widehat{var}(\widehat{y})$; see Table 7. To test these prediction errors, we select $\alpha = 0.20$ (such a relatively high value is typical when applying Bonferroni's inequality) so $t_{10-1}(0.20/(10 \times 2))$ $= t_9(0.01) = 2.821$ where degree of freedom $v = \min(10-1, 20-1) = 9$. This table shows that $\max_{l;i} t_{(l);i} = t_{(CT);6} = 2.48$ (for $m_{val} = 10$ we find $\max_{l;i} t_{(l);i}$ $= t_{(NT);10} = 2.10$), so we accept the two metamodels. We conclude that all three MSB assumptions hold, for this case study.

5 Conclusion

We present a novel method for factor screening in (random) discrete-event simulation with multiple response types; we call this method "multiple sequential bifurcation" (MSB). Our MSB assumes (i) a second-order polynomial is a valid metamodel of the I/O function of the simulation model; (ii) the first-order effects have known signs; (iii) the "heredity" property applies. Our case-study shows that realistic simulation models may indeed satisfy these three assumptions.

Table 6: Inputs of the TPL-MRCD system

ID	Description	$L_{(CT)}$	$H_{(CT)}$	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 (minutes)	2	4	+	−
9	Scanning time of unit parts in CDDC (seconds)	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 transportation	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	+	−

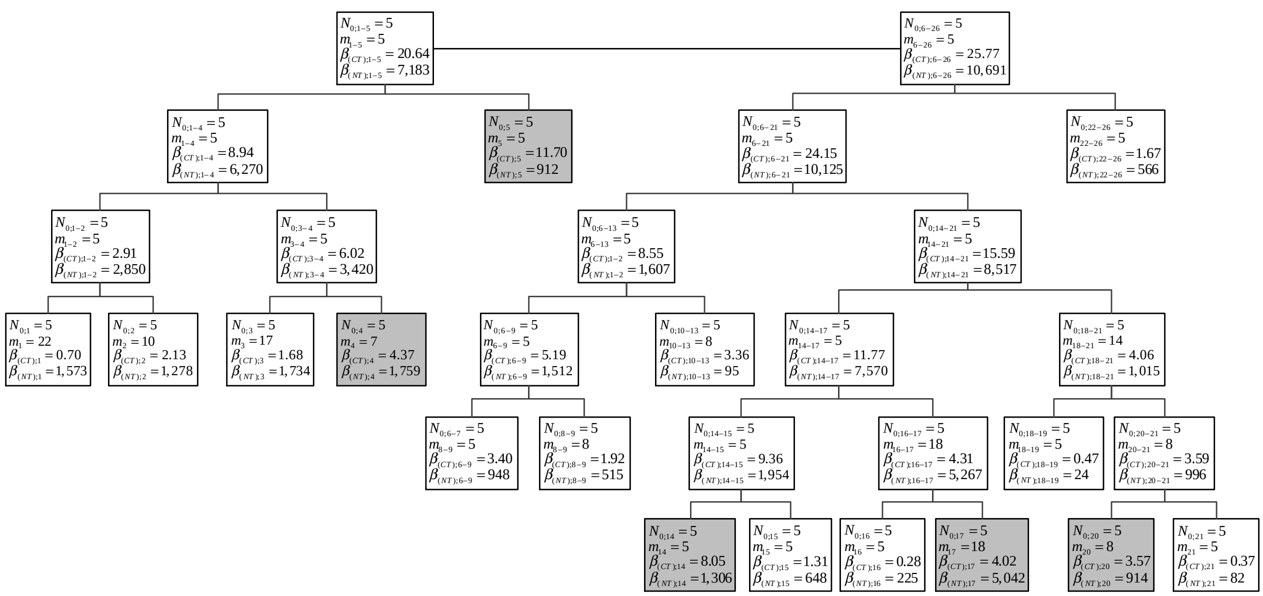

Figure 5: MSB for TPL-MRCD with $\Delta_{(CT);0} = 2.5$, $\Delta_{(NT);0} = 2,000$, $\Delta_{(CT);0} = 5$, $\Delta_{(NT);1} = 3,000$

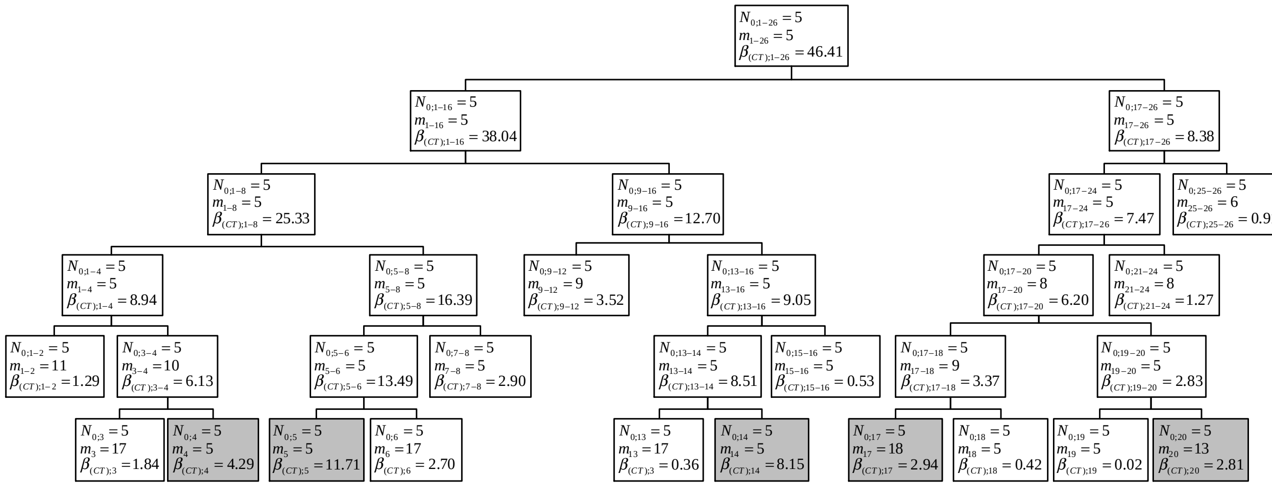

Figure 6: SB for CT with $\Delta_{(CT);0} = 2.5, \Delta_{(CT);0} = 5$

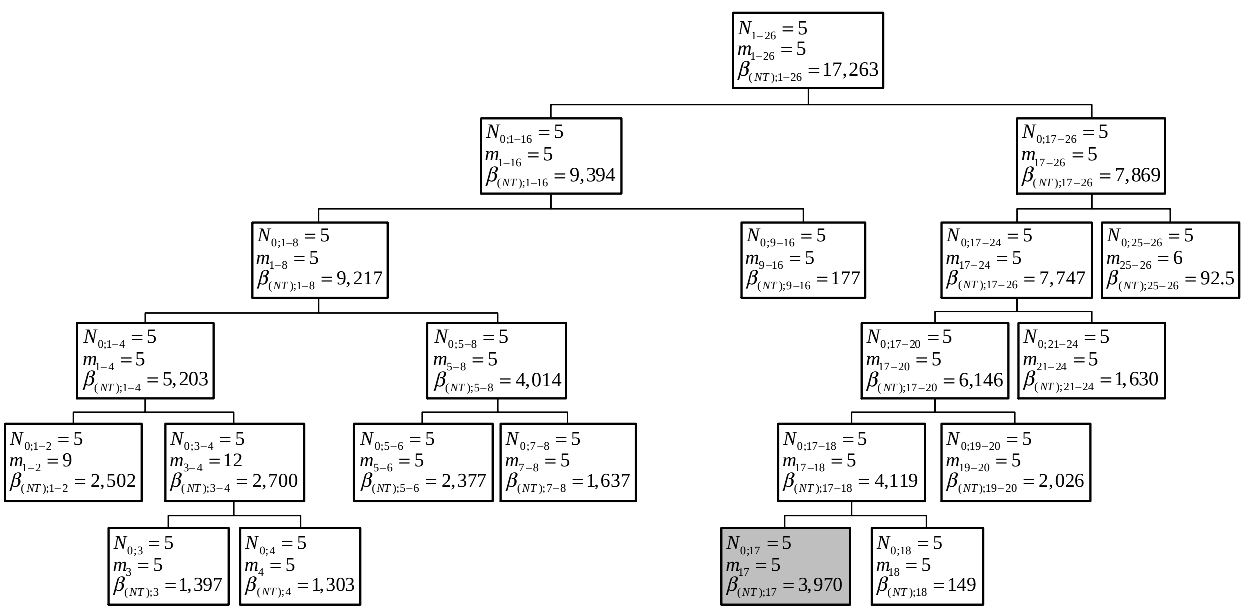

Figure 7: SB for NT with $\Delta_{(NT);0} = 2,000$, $\Delta_{(NT);1} = 3,000$

Table 7: Validation of MSB with 20 replicates

i	1	2	3	4	5	6	7	8	9	10
$\overline{w}_{CT;i}$	28.09	31.28	25.06	45.22	42.02	67.57	24.30	38.63	27.25	58.65
$\widehat{var}(\overline{w}_{CT;i})$	1.38	0.004	0.38	0.80	0.02	1.70	0.26	1.92	1.04	1.30
$\widehat{y}_{CT;i}$	34.03	30.01	26.53	49.40	40.78	61.26	26.26	34.37	24.96	54.58
$\widehat{var}(\widehat{y}_{CT;i})$	0.07	0.10	0.05	0.04	0.05	0.08	0.04	0.08	0.05	0.06
$t_{CT;i}$	2.43	0	0	1.29	0	2.48	0	0.89	0	0.86
$\overline{w}_{NT;i}$	49,387	53,664	53,122	45,513	51,563	38,952	51,003	44,424	48,402	51,562
$var(\overline{w}_{NT:i})$	64,407	209,913	36,151	9,250	52,819	23,850	97,016	30,896	21,505	3,876
$\widehat{y}_{NT:i}$	46,738	52,531	51,475	43,665	50,991	40,323	51,397	45,007	51,669	48,317
$\widehat{var}(\widehat{y}_{NT;i})$	9,530	14,028	6,613	5,427	6,131	10,392	5,008	11,161	6,963	7,554
$t_{NT;i}$	0	0	0	0	0	0	0	0	1.59	2.30

MSB extends sequential bifurcation (SB), originally published by Bettonvil and Kleijnen (1997). Later on, Wan et al. (2010) extended this SB to random simulation, using the SPRT to determine the number of replicates in each stage such that the type-I and type-II error rates are controlled. We now derive a more efficient rule for determining the initial number of replicates needed to start the SPRT. Moreover, we extend this SPRT to simulation with multiple responses.

More specifically, MSB uses batches of inputs such there is no cancellation of main effects for any response type. Moreover, MSB includes a novel procedure to validate the three assumptions of SB and MSB. Finally, MSB takes advantage of the fact that running a simulation model gives observations on all response types; i.e., when screening for one response, all the other response types are "for free".

Our Monte Carlo experiments ensure that all three assumptions of SB or MSB are satisfied. The first experiments show that MSB requires fewer replicates than Wan et al.'s SB; i.e., MSB is more efficient. Our next experiment considers two outputs, a hundred inputs in two batches, and four problem characteristics; namely, the effects' sparsity, signal-noise, variability, and clustering. This experiment shows that compared with SB our MSB is more efficient (fewer replicates), and more effective (better control of the two error rates).

Our Chinese third-party logistics case-study has two outputs and 26 inputs in two batches. MSB finds the same important inputs as SB, but requires fewer replicates. The validation procedure accepts all three assumptions used in MSB and SB. So MSB is a robust factor screening method.

Future research may focus on the SPRT following De and Baron (2012), who give many recent references—albeit not in a simulation context. In this paper we focus on two outputs; future research should examine more than two outputs. Additional case studies should confirm the robustness of MSB.

References

Bettonvil, B., J. P. C. Kleijnen. 1997. Searching for important factors in simulation models with many factors: Sequential bifurcation. *Eur. J. Oper. Res.* 96(1): 180-194.

Chan, K. K., T. A. Spedding. 2001. On-line optimization of quality in a manufacturing system. *Int. J. Prod. Res.* 39(6): 1127-1145.

Cheng, R. C. H., 1997. Searching for important factors: sequential bifurcation under uncertainty. D. H. Withers, L. N. Barry, S. Andradottir, K. J. Healy eds. *Proc. 1997 Winter Simulation Conf.*, Institute of Electrical and Electronics Engineers, Piscataway, NJ, 275-280.

Dabbas, R. M., H.-N. Chen, J. W. Fowler, et al., 2001. A combined dispatching criteria approach to scheduling semiconductor manufacturing systems. *Comput. Ind. Eng.* 39(3-4): 307-324.

De, S.K. and M. Baron (2012), Step-up and step-down methods for testing multiple hypotheses in sequential experiments. *J. Stat. Plan. Infer.* , in press

Ekren, B. Y., S. S. Heragu, A. Krishnamurthy, et al., 2010. Simulation based experimental design to identify factors affecting performance of AVS/RS. *Comput. Ind. Eng.* 58(1): 175-185.

Feng, F., X. Zhang, N. Zhang, Y. Fu. 2010. *Annual report on automotive industry in China*. Social Sciences Academic Press (China), Inc., Beijing.

Kelton, W. D., R. P. Sadowski, D. T. Sturrock. 2007. *Simulation with ARENA*. McGraw-Hill, New York.

Kleijnen, J. P. C., 1987. *Statistical Tools for Simulation Practitioners.* Marcel Dekker, New York.

Kleijnen, J. P. C., 2003. Supply chain simulation: a survey. *International Journal of Simulation and Process Modeling.* 1031-20.

Kleijnen, J. P. C., 2008. *Design and Analysis of Simulation Experiments.* Springer US, New York.

Kleijnen, J. P. C., 2009. Factor screening in simulation experiments: Review of sequential bifurcation. C. Alexopoulos, D. Goldsman, J.R. Wilson, eds. *Advancing the Frontiers of Simulation.* Springer, New York, 153-167.

Kleijnen, J. P. C., B. Bettonvil, F. Persson. 2006. Screening for the important factors in large discrete-event simulation models: Sequential bifurcation and its applications. A. Dean, S. Lewis, eds. *Screening: Methods for Experimentation in Industry, Drug Discovery, and Genetic.* Springer, New York, 287-307.

Kleijnen, J. P. C., S. M. Sanchez, T. W. Lucas, T. M. Cioppa. 2005. State-of-the-art review: A user's guide to the brave new world of designing simulation experiments. *INFORMS J. Comput.* 17(3): 263-289.

Kleijnen, J. P. C., M. T. Smits. 2003. Performance metrics in supply chain management. *J. Oper. Res. Soc.* 54(5): 507-514.

Kumar, S., D. A. Nottestad. 2006. Capacity design: an application using discrete-event simulation and designed experiments. *IIE Trans.* 38(9): 729 - 736.

Law, A. M., 2007. *Simulation modeling and analysis.* McGraw-Hill, Boston.

Miller, G.A., 1956. The magical number seven plus or minus two: some limits on our capacity for processing information. *The Psychological Review*, 63:81-97.

Montgomery, D. C., 2007. *Design and analysis of experiments.* John Wiley & Sons, New York.

Shang, J., S. Li, P. Tadikamalla. 2004. Operational design of a supply chain system using the Taguchi method, response surface methodology, simulation, and optimization. *Int. J. Prod. Res.* 42(18): 3823-3849.

Shang, J., P. Tadikamalla. 1998. Multicriteria design and control of a cellular manufacturing system through simulation and optimization. *Int. J. Prod. Res.* 36(6): 1515-1528.

Shen, H., H. Wan. 2009. Controlled sequential factorial design for simulation factor screening. *Eur. J. Oper. Res.* 198(2): 511-519.

Shen, H., H. Wan, S. M. Sanchez. 2010. A hybrid method for simulation factor screening. *Nav. Res. Log.* 57(1): 45-57.

Wald, A., 1947. *Sequential Analysis.* Wiley, New York.

Wan, H., B. E. Ankenman, B. L. Nelson. 2006. Controlled sequential bifurcation: A new factor-screening method for discrete-event simulation. *Oper. Res.* 54(4): 743-755.

Wan, H., B. E. Ankenman, B. L. Nelson. 2010. Improving the efficiency and efficacy of controlled Sequential Bifurcation for simulation factor screening. *INFORMS J. Comput.* 22(3): 482-492.

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

Yalcnkaya, O., G. Mirac Bayhan. 2009. Modelling and optimization of average travel time for a metro line by simulation and response surface methodology. *Eur. J. Oper. Res.* 196(1): 225-233.

Acknowledgment

We thank Jennifer Shang (Pittsburgh University) for helping us with the case study on the supply chain for automobile parts in China. We dedicate this paper to the memory of Bert Bettonvil, who defended his doctoral dissertation on SB in 1990, and passed away in 2011. This work is partly supported by the National Natural Sciences Foundation of China under Grants NSFC71072034.

Appendix 1: Proof of Theorem 1

We give a proof for the situation that both output l and output l' increase when changing inputs j' through j from $L_{(l)}$ to $H_{(l)}$. For output l, (7) can be proven using (3) directly (also see Bettonvil and Kleijnen 1997). For output l' we assume $a_{(l');i}$ is the coded values of input i for $w_{l'}$ where $a_{(l');i} = -1$ or $a_{(l');i}$ $= 1$ when changing the two levels of the individual input i from $L_{(l)}$ to $H_{(l)}$. Thus, the signs of the effects of the inputs j' through j for both w_l and $w_{l'}$ are $+$. We can then derive the following equations for the other output, l':

$$\begin{aligned} E(w_{(l\to l');(j)}) &= \beta_{(l');0} + a_1\beta_{(l');1} + \cdots + a_{j'-1}\beta_{(l');j'-1} + (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\ &\quad -a_{j+1}\beta_{(l');j+1} - \cdots - a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\ &\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K} \end{aligned} \quad (19)$$

$$\begin{aligned} E(w_{(l\to l');-(j)}) &= \beta_{(l');0} - a_1\beta_{(l');1} - \cdots - a_{j'-1}\beta_{(l');j'-1} - (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\ &\quad +a_{j+1}\beta_{(l');j+1} + \cdots + a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\ &\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K} \end{aligned} \quad (20)$$

$$
\begin{aligned}
E(w_{(l\to l');(j'-1)}) &= \beta_{(l');0} + a_1\beta_{(l');1} + \cdots + a_{j'-1}\beta_{(l');j'-1} - (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\
&\quad - a_{j+1}\beta_{(l');j+1} - \cdots - a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\
&\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K}
\end{aligned} \tag{21}
$$

$$
\begin{aligned}
E(w_{(l\to l');-(j'-1)}) &= \beta_{(l');0} - a_1\beta_{(l');1} - \cdots - a_{j'-1}\beta_{(l');j'-1} + (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\
&\quad + a_{j+1}\beta_{(l');j+1} + \cdots + a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\
&\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K}
\end{aligned} \tag{22}
$$

The unbiased group estimator $\widehat{\beta_{(l');j'-j}}$ defined in (8) follows from (19) through (22).

Appendix 2: Proof of Theorem 2

We give a proof for the situation that changing inputs j' through j from $L_{(l)}$ to $H_{(l)}$ makes w_l increase and $w_{l'}$ decrease. For w_l, our proof is the same as in (7). For $w_{l'}$ we still use the same $a_{(l');i}$ defined in the Proof of Theorem 1. We can then derive the following equations:

$$
\begin{aligned}
E(w_{(l\to l');(j)}) &= \beta_{(l');0} + a_1\beta_{(l');1} + \cdots + a_{j'-1}\beta_{(l');j'-1} - (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\
&\quad - a_{j+1}\beta_{(l');j+1} - \cdots - a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\
&\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K}
\end{aligned} \tag{23}
$$

$$
\begin{aligned}
E(w_{(l\to l');-(j)}) &= \beta_{(l');0} - a_1\beta_{(l');1} - \cdots - a_{j'-1}\beta_{(l');j'-1} - (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\
&\quad + a_{j+1}\beta_{(l');j+1} + \cdots + a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\
&\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K}
\end{aligned} \tag{24}
$$

$$
\begin{aligned}
E(w_{(l\to l');(j'-1)}) &= \beta_{(l');0} + a_1\beta_{(l');1} + \cdots + a_{j'-1}\beta_{(l');j'-1} + (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\
&\quad - a_{j+1}\beta_{(l');j+1} - \cdots - a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\
&\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K}
\end{aligned} \tag{25}
$$

$$
\begin{aligned}
E(w_{(l\to l');-(j'-1)}) &= \beta_{(l');0} - a_1\beta_{(l');1} - \cdots - a_{j'-1}\beta_{(l');j'-1} - (\beta_{(l');j'} + \cdots + \beta_{(l');j}) \\
&\quad + a_{j+1}\beta_{(l');j+1} + \cdots + a_K\beta_{(l');K} + a_1a_2\beta_{(l');1;2} + \cdots - a_1a_K\beta_{(l');1;K} \\
&\quad + \cdots + a_{K-1}a_K\beta_{(l');(K-1);K} + a_1^2\beta_{(l');1;1} + \cdots + a_K^2\beta_{(l');K;K}
\end{aligned} \tag{26}
$$

The unbiased group estimator $\widehat{\beta_{(l');j'-j}}$ defined in (12) follows from (23) through (26) where all quadratic effects cancel out.

Appendix 3: Two outputs and one batch of inputs

Case (i): Each input makes both outputs increase

Table 8: Case 1 of MSB with two outputs

Input	Low level for w_1	High level for w_1	w_1	w_2
1	$L_{(1);1}$	$H_{(1);1}$	+	+
2	$L_{(1);2}$	$H_{(1);2}$	+	+
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
K	$L_{(1);K}$	$H_{(1);K}$	+	+

Table 8 shows that there is only one batch and $L_{(1)} = L_{(2)}$ and $H_{(1)} = H_{(2)}$. Consequently, in this case MSB closely resembles SB (for a single output type). However, MSB classifies a (sub)group of inputs as important whenever that group has an important effect on at least one of the outputs; i.e., a group is declared unimportant only if its estimated effects for both w_1 and w_2 are unimportant.

Table 9 shows the complete MSB procedure for this case; Applying Theorem 1, we can compute the (sub)group effects for both outputs simultaneously.

Note that Step (2) estimates two big group effects $\beta_{(1);1-K}$ and $\beta_{(2);1-K}$, whereas in the main text we estimate batch effects directly from four input combinations instead of two combinations; i.e., $\beta_{(1);1-k_1}$, $\beta_{(2);1-k_1}$, $\beta_{(1);k_1+1-K}$ and $\beta_{(2);k_1+1-K}$.

Case (ii): Each input makes one output increase and the other output decrease

Like in Case (i), there is only one batch but now $L_{(1)} = H_{(2)}$ and $H_{(1)} = L_{(2)}$; see Table 10.

MSB for this case proceeds analogously to MSB for Case (i); see Table 11.

Appendix 4: Multi-response MSB

Table 12 illustrates the formation of batches of inputs in the case of $n > 2$ outputs. Obviously, this formation becomes complicated if the outputs display completely different signs. As a simple rule, we propose to split inputs into as few batches as possible: the fewer batches there are, the fewer combinations are required to estimate (sub)group effects.

Table 13 illustrates the detailed procedure for this case.

.

Appendix 5: Validation of MSB Assumptions

Table 14 shows the ten inputs combinations selected by LHS for the validation of the MSB assumptions. Table 15 shows the validation results in case of only ten replicates.

.

Table 9: MSB for case 1

(1) Define the values of all K inputs such that changing each individual input from $L_{(1)}$ to $H_{(1)}$ makes both outputs increase.
(2) Use SPRT with initial sample size $N_{0;1-K}$ to find the number of replicates m_{1-K}: $\begin{pmatrix} w_{(1);(K);r} & w_{(1);-(K);r} \\ w_{(1\to 2);(K);r} & w_{(1\to 2);-(K);r} \end{pmatrix}$ $(r = 1, \ldots, m_{1-K})$,and estimate $(\overline{\widehat{\beta_{(1);1-K}}}, \overline{\widehat{\beta_{(2);1-K}}})'$ through (7) and (8).Note: $w_{(1);-(K);r} = w_{(1);(0);r}$
(a) If SPRT declares $(\overline{\widehat{\beta_{(1);1-K}}}, \overline{\widehat{\beta_{(2);1-K}}})'$ unimportant, then stop MSB;
(b) else split the group into two subgroups $(1 - k_1, k_1 + 1 - K)$ with k_1 a power of two.
(3) Use SPRT with initial sample size $N_{0;1-k_1}$ to find the number of replicates m_{1-k_1}: $\begin{pmatrix} w_{(1);(k_1);r} & w_{(1);-(k_1);r} \\ w_{(1\to 2);(k_1);r} & w_{(1\to 2);-(k_1);r} \end{pmatrix}$, and estimate $(\overline{\widehat{\beta_{(1);1-k_1}}}, \overline{\widehat{\beta_{(2);1-k_1}}})'$ and $(\overline{\widehat{\beta_{(1);k_1+1-K}}}, \overline{\widehat{\beta_{(2);k_1+1-K}}})'$ through (7) and (8).
For the first subgroup:
(a) If SPRT declares $(\overline{\widehat{\beta_{(1);1-k_1}}}, \overline{\widehat{\beta_{(2);1-k_1}}})'$ unimportant, then discard the first subgroup;
(b) else split the first subgroup into two subgroups, similar to Step (2) (b).
For the second subgroup:
(a) If SPRT declares $(\overline{\widehat{\beta_{(1);k_1+1-K}}}, \overline{\widehat{\beta_{(2);k_1+1-K}}})'$ unimportant, then discard the second subgroup;
(b) else split the second subgroup into two subgroups, similar to (2) (b).
...
Final: Use SPRT with initial sample size $N_{0;j}$ to identify the important individual inputs j, and estimate their main effects.

Table 10: Case 2 of MSB with two outputs

Input	Low level for w_1	High level for w_1	w_1	w_2
1	$L_{(1);1}$	$H_{(1);1}$	+	−
2	$L_{(1);2}$	$H_{(1);2}$	+	−
⋮	⋮	⋮	⋮	⋮
K	$L_{(1);K}$	$H_{(1);K}$	+	−

Table 11: MSB for Case 2

(1) Define the values of all K inputs such that changing each individual input from $H_{(1)}$ to $L_{(1)}$ makes w_1 increase and w_2 decrease.

(2) Use SPRT to select the number of replicates m_{1-K}: $\begin{pmatrix} w_{(1);(K);r} & w_{(1);-(K);r} \\ w_{(1\to 2);(K);r} & w_{(1\to 2);-(K);r} \end{pmatrix}$ $(r = 1, \ldots, m_{1-K})$, and estimate $(\widehat{\beta_{(1);1-K}}, \widehat{\beta_{(2);1-K}})'$ through (11) and (12).

- **(a)** If SPRT declares $(\widehat{\beta_{(1);1-K}}, \widehat{\beta_{(2);1-K}})'$ unimportant, then stop MSB;
- **(b)** else split the group into two subgroups $(1 - k_1, k_1 + 1 - K)$ with k_1 a power of two.

(3) Use SPRT to select the number of replicates m_{1-k_1}: $\begin{pmatrix} w_{(1);(k_1);r} & w_{(1);-(k_1);r} \\ w_{(1\to 2);(k_1);r} & w_{(1\to 2);-(k_1);r} \end{pmatrix}$; estimate $\widehat{\beta_{(1);1-k_1}}, \widehat{\beta_{(2);1-k_1}})'$ and $(\widehat{\beta_{(1);k_1+1-K}}, \widehat{\beta_{(2);k_1+1-K}})'$ through (11) and (12).

For the first subgroup:

- **(a)** If SPRT declares $(\widehat{\beta_{(1);1-k_1}}, \widehat{\beta_{(2);1-k_1}})'$ unimportant, then discard the first subgroup;
- **(b)** else split the first subgroup into two subgroups, similar to Step (2) (b).

For the second subgroup:

- **(a)** If SPRT declares $(\widehat{\beta_{(1);k_1+1-K}}, \widehat{\beta_{(2);k_1+1-K}})'$ unimportant, then discard the second subgroup;
- **(b)** else split the second subgroup into two subgroups, similar to (2) (b).

...

Final: Use SPRT to identify the important individual inputs, and estimate their main effects.

Table 12: Batches for multiple outputs

Batch	Input	Low level for w_1	High level for w_1	w_1	w_2	$\cdots$	w_n
	1	$L_{(1);1}$	$H_{(1);1}$	+	+	+	−
1	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	k_1	$L_{(1);k_1}$	$H_{(1);k_1}$	+	+	+	−
	k_1+1	$L_{(1);k_1+1}$	$H_{(1);k_1+1}$	+	+	−	−
2	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	k_2	$L_{(1);k_2}$	$H_{(1);k_2}$	+	+	−	−
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	$k_{p-1}+1$	$L_{(1);k_{p-1}+1}$	$H_{(1);k_{p-1}+1}$	+	⋮	⋮	⋮
p	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	k_p	$L_{(1);k_p}$	$H_{(1);k_p}$	+	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	$k_{q-1}+1$	$L_{(1);k_{q-1}+1}$	$H_{(1);k_{q-1}+1}$	+	−	⋮	+
q	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	K	$L_{(1);k_q}$	$H_{(1);k_q}$	+	−	…	+

Table 13: MSB procedure with multiple outputs

(1) Define the values of all K inputs for output l such that changing the individual input i from $L_{(l);i}$ to $H_{(l);i}$ makes w_l increase, with $i = 1, \cdots, K$, $l = 1, \cdots, n$.
(2) Use SPRT to select the number of replicates: $(w_{(1);(K);r}, w_{(1);-(K);r}, w_{(2);(K);r}, w_{(2);-(K);r}, \cdots, w_{(n);(K);r}, w_{(n);-(K);r})'$; estimate $\left(\widehat{\beta_{(1);1-K}}, \widehat{\beta_{(2);1-K}}, \cdots, \widehat{\beta_{(n);1-K}}\right)'$.
(a) If SPRT declares $\left(\widehat{\beta_{(1);1-K}}, \widehat{\beta_{(2);1-K}}, \cdots, \widehat{\beta_{(n);1-K}}\right)'$ unimportant, then stop MSB;
(b) else split the group into q batches $(1-k_1, k_1+1-k_2, \cdots, k_{q-1}+1-K)$, such that each individual output within a batch has the same sign.
(3) Use SPRT to select the number of replicates: $\begin{pmatrix} w_{(1);(k_p);r} & w_{(1);-(k_p);r} \\ w_{(1\to l');(k_p);r} & w_{(1\to l');-(k_p)r} \end{pmatrix}'$; estimate $\left(\widehat{\beta_{(l);1-k_1}}, \cdots, \widehat{\beta_{(l);k_{p-1}+1-K}}\right)$ $l = 1, 2, \cdots, n$, $l' = 2, 3, \cdots, n$, $p = 1, 2, \cdots, q$.
(a) If SPRT declares $\left(\widehat{\beta_{(1);k_p+1-k_p}}, \cdots, \widehat{\beta_{(n);k_{p-1}-k_p}}\right)'$ of batch $(k_{p-1}+1-k_p)$ unimportant, then discard this batch $(p = 1, \cdots, q)$;
(b) else split the batch into two smaller batches $(k_{p-1}+1-k_{p'}, k_{p'}+1-k_p)$ with $k_{p'} - k_{p-1}$ a power of two if possible.
...
Final: Use SPRT to identify the important individual inputs, and estimate their main effects.

Table 14: LHS for Validation of MSB asumptions

Input	Validation combination 1	2	3	4	5	6	7	8	9	10
1	1.68	1.4	1.2	1.89	2.12	1.76	1.98	1.59	1.22	1.49
2	1.49	1.48	1.79	1.58	1.37	1.13	1.19	1.72	1.29	1.08
3	1.47	1.27	1.10	1.03	1.43	1.17	1.24	1.35	1.38	1.11
4	1.77	1.21	1.56	1.69	2.21	1.87	2.02	2.26	2.41	1.37
5	1.4	1.55	1.44	1.2	0.85	1.03	1.17	1.32	1.12	0.89
6	14.05	10.75	14.9	13.14	11.15	12.27	13.91	11.6	10.06	12.93
7	2.21	2.6	2.86	2.4	2.78	2.63	2.03	2.47	2.17	2.99
8	2.79	3.05	2.34	3.7	2.01	3.27	2.51	3.87	2.87	3.55
9	23.2	24.5	27.57	28.29	21.44	20.74	25.92	26.41	29.33	22.41
10	3.49	3.91	2.94	2.31	3.05	2.0	3.75	2.44	2.77	3.39
11	2.15	1.74	2.08	2.49	2.23	1.83	1.95	1.53	2.34	1.65
12	2.15	1.74	2.08	2.49	2.23	1.86	1.95	1.53	2.34	1.65
13	0.01	0.02	0.01	0.01	0.02	0.01	0.02	0.02	0.02	0.01
14	11	15	23	26	16	27	20	30	21	12
15	21	25	27	24	29	11	16	19	13	15
16	11	16	14	19	14	13	17	11	15	19
17	27	30	38	23	33	21	37	25	35	30
18	53	59	53	56	57	54	59	51	51	57
19	20.21	26.46	28.26	21.26	25.56	24.15	27.51	23.05	29.2	22.87
20	94.41	99.68	81.16	79.86	86.33	75.04	90.79	95.32	88.76	84.28
21	87.04	97.6	95.61	87.62	90.82	94.53	84.28	77.35	79.89	81.8
22	23.13	23.73	20.04	22.66	21.16	21.68	23.41	22.13	21.33	20.63
23	SPT	SPT	FIFO	SPT	FIFO	FIFO	FIFO	SPT	SPT	FIFO
24	30	9.33	7.31	6.99	9.68	8.72	6.32	5.29	5.77	7.84
25	20	16	14	17	11	14	13	17	11	19
26	12.42	22.92	17.88	13.69	14.9	19.77	22.25	18.04	21.38	16.13

Table 15: Validation of MSB with 10 replicates

i	1	2	3	4	5	6	7	8	9	10
$\overline{w}_{CT;i}$	30.00	31.18	25.34	46.87	41.87	66.43	24.15	37.95	27.11	60.33
$\widehat{var}(\overline{w}_{CT;i})$	0.29	0.005	1.21	0.52	0.01	1.98	0.24	4.86	2.25	2.65
$\widehat{y}_{CT;i}$	34.03	30.01	26.53	49.40	40.78	61.26	26.26	34.37	24.96	54.58
$\widehat{var}(\widehat{y}_{CT;i})$	0.07	0.10	0.05	0.04	0.05	0.08	0.04	0.08	0.05	0.06
$t_{CT;i}$	0.39	0	0	0	0	1.51	0	0.26	0	1.67
$\overline{w}_{NT;i}$	46,462	52,947	53,218	45,495	51,368	38,628	50,926	44,189	48,486	51,577
$var(\overline{w}_{NT:i})$	203,217	412,539	106,129	18,524	82,418	43,443	116,051	65,129	47,579	7,631
$\widehat{y}_{NT:i}$	46,738	52,531	51,475	43,665	50,991	40,323	51,397	45,007	51,669	48,317
$\widehat{var}(\widehat{y}_{NT;i})$	9,530	14,028	6,613	5,427	6,131	10,392	5,008	11,161	6,963	7,554
$t_{NT;i}$	0	0	0	0	0	0	0	0	0.78	2.10