U:\BIKASECR\KLEIJNEN\288.WPD

June 26, 1997 (10:40am)

6. EXPERIMENTAL DESIGN FOR SENSITIVITY ANALYSIS, OPTIMIZATION, AND VALIDATION OF SIMULATION MODELS

Jack P.C. Kleijnen, Department of Information Systems/Center for Economic Research (CentER), Tilburg University, 5000 LE Tilburg, Netherlands

E-mail: kleijnen@kub.nl
Fax: +3113-663377
Web: http://cwis.kub.nl/~few5/center/STAFF/kleijnen/cv.htm

Draft prepared for

Handbook of Simulation, Jerry Banks, Editor, to be published by Wiley, New York

Version: June 26, 1997 (10:40am)

Handbook of Simulation, edited by Jerry Banks, Wiley, 1997

1. Principles of Simulation, Jerry Banks, Georgia Tech, Atlanta, GA

2. Modeling and Simulation, Alan Pritsker, Pritsker & Associates, Indianapolis, IN

3. Input Data Analysis, Stephen Vincent, Consultant, Greendale, WI

4. Random Number Generation, Pierre L'Ecuyer, University of Montreal, Montreal Canada

5. Random Variate Generation, Russell C.H. Cheng, The University of Kent at Canterbury, Canterbury, ENGLAND

6. Experimental Design, Jack P.C. Kleijnen, Tilburg University, Tilburg, Netherlands

7. Output Data Analysis, Christos Alexopoulos, Georgia Tech, Atlanta, GA, and Andrew F. Seila, University of Georgia, Athens, GA

8. Comparing Systems via Simulation, David Goldsman, University, Atlanta, GA, and Barry L. Nelson, Georgia Tech, Northwestern, Evanston, IN

9. Optimization, Sigrun Andradottir, University of Wisconsin, Madison, WI

10. Verification, Validation and Testing, Osman Balci, VPI&SU, Blacksburg, VA

11. Simulation Software, Tom Schriber, University of Michigan, Ann Arbor, MI, and Dan Brunner, System Flow, Indianapolis, IN

12. Object Oriented Simulation, Steve Roberts, NC State University, Raleigh, NC

13. Parallel and Distributed Simulation, Richard Fujimoto, Georgia Tech, Atlanta, GA

14. Manufacturing and Material Handling Systems, Matt Rohrer, AutoSimulations, Inc., Bountiful, UT

15. Automobile Industry, Onur Ulgen and Ali Gunal, Production Modeling Corporation, Dearborn, MI

16. Transportation and Logistics Systems, S. Manivannan, Consolidated Freightways, Portland, OR

17. Health Care, Frank McGuire, SunHealth, Inc., Charlotte, NC

18. Service Systems, Ron Laughery, Micro Analysis & Design, Inc., Boulder, CO

19. Military Simulation, Keebom Kang, Naval Postgraduate School, Monterey, CA, and Jay Roland, Rolands and Associates Corporation, Monterey, CA

20. Computer and Communication Systems, Herb Schwetman and Al Hartmann, Mesquite Software, Austin, TX

21. Simulation and Scheduling, Ali S. Kiran, Kiran & Associates, San Diego, CA

22. Realtime Simulation, Wayne Davis, Univ. Of Ill., Champaign-Urbana, IL

23. Guidelines for Success, Ken Musselman, Pritsker Corporation, W. Lafayette, IN

24. Managing the Simulation Project, Van Norman, AutoSimulations, Inc., Bountiful, UT

25. Vendor Survey, Jerry Banks, Georgia Tech, Atlanta, GA

Experimental Design for Sensitivity Analysis, Optimization, and Validation of Simulation Models

Jack P.C. Kleijnen, Department of Information Systems and Auditing (BIKA)/Center for Economic Research (CentER), Tilburg University (KUB), 5000 LE Tilburg, Netherlands
e-mail: kleijnen@kub.nl; fax: +3113-663377;
Web: http://cwis.kub.nl/~few5/center/STAFF/kleijnen/cv.htm)

Abstract

This chapter gives a survey on the use of statistical designs for what-if analysis in simulation, including sensitivity analysis, optimization, and validation/verification. Sensitivity analysis is divided into two phases. The first phase is a pilot stage, which consists of screening or searching for the important factors among (say) hundreds of potentially important factors. A novel screening technique is presented, namely sequential bifurcation. The second phase uses regression analysis to approximate the input/output transformation that is implied by the simulation model; the resulting regression model is also known as a metamodel or a response surface. Regression analysis gives better results when the simulation experiment is well designed, using either classical statistical designs (such as fractional factorials) or optimal designs (such as pioneered by Fedorov, Kiefer, and Wolfowitz). To optimize the simulated system, the analysts may apply Response Surface Methodology (RSM); RSM combines regression analysis, statistical designs, and steepest-ascent hill-climbing. To validate a simulation model, again regression analysis and statistical designs may be applied. Several numerical examples and case-studies illustrate how statistical techniques can reduce the ad hoc character of simulation; that is, these statistical techniques can make simulation studies give more general results, in less time. Appendix 1 summarizes confidence intervals for expected values, proportions, and quantiles, in terminating and steady-state simulations. Appendix 2 gives details on four variance reduction techniques, namely common pseudorandom numbers, antithetic numbers, control variates or regression sampling, and importance sampling. Appendix 3 describes jackknifing, which may give robust confidence intervals.

Keywords

least squares, distribution-free, non-parametric, stopping rule, run-length, Von Neumann, median, seed, likelihood ratio

1. Introduction

This chapter gives a survey of 'Design Of Experiments' or 'DOE' (which includes designs such as 2^{k-p} designs) applied to simulation. The related term 'experimental design' may suggest that this subdiscipline is still experimental, but it is not: see early publications such as Plackett and Burman (1946). DOE is a subdiscipline within mathematical statistics. This chapter is a tutorial that discusses not only methodology, but also applications. These applications come from the author's own experience as a consultant, and from publications by others in the USA and Europe. The reader is assumed to have a basic knowledge of mathematical statistics and simulation.

The Introduction will address the questions: *what* is DOE, and *why* is DOE needed? These questions can be illustrated through the following two case studies.

The first case concerns an ecological study that uses a deterministic simulation model (consisting of a set of non-linear difference equations) with 281 parameters. The ecological experts are interested in the effects of these parameters on the response, namely, future carbon-dioxide or CO_2 concentration; CO_2 is the major cause of the greenhouse effect. The pilot phase of this study aims at *screening*: which factors among the many potentially important factors are really important? Details will be given in §2 (on screening).

The second case study concerns a Decision Support System (DSS) for production planning in a specific Dutch steel tube factory. The DSS and the factory are modeled through a stochastic, discrete-event simulation (stochastic or random simulations have a special variable, namely the pseudorandom number seed). The DSS is to be optimized. This DSS has fourteen input or decision variables; there are two response variables, namely, productive hours and lead time. Simulation of one combination of these fourteen inputs takes six hours on the computer available at that time, so searching for the optimal combination must be performed with care. Details will be given in §5 (on RSM).

Decision-making is an important use of simulation. In particular, factor screening and optimization concern that topic. Note that closely related to optimization is goal seeking: given a target value for the response variable, find the corresponding input values; the regression metamodel of this chapter can be used to find these desired input values. Much of this chapter is about the use of simulation to improve decision-making. This will be demonstrated by several case studies; more case studies -especially in manufacturing- are reviewed in Yu and Popplewell (1994).

It is convenient to introduce now some DOE *terminology*, defined in a simulation context. A *factor* is a parameter, an input variable, or a module of a simulation model (or simulation computer program). Examples of parameters and input variables have already been given in the preceding discussion of two case studies. Other examples are provided by classic queueing simulations: a parameter may be a customer arrival rate or a service rate; an input variable may be the number of parallel servers; a module may be the submodel for the priority rules (First-In-First-Out or FIFO, Shortest-Processing-Time or SPT, and so on).

By definition, factors are changed during an experiment: they are not kept constant during the whole experiment. Hence a factor takes at least two *levels* or 'values' during the experiment. The factor may be *qualitative*, as the priority rules exemplified. A detailed discussion of qualitative factors and various measurement scales is given in Kleijnen (1987, pp. 138-142).

The central problem in DOE is the astronomically great *number of combinations of factor levels*. For example, in the ecological case study at least 2^{281} ($> 10^{84}$) combinations may be distinguished; a queueing network also has many factors. DOE can be defined as selecting the combinations of factor levels that will be actually simulated in an experiment with the simulation model.

After this selection of factor combinations, the simulation program is executed or 'run' for these combinations. Next DOE analyzes the resulting input/output (I/O) data of the experiment, to derive conclusions about the importance of the factors. In simulation this is also known as *what-if analysis*: what happens if the analysts change parameters, input variables or modules of the simulation model? This question is closely related to sensitivity analysis, optimization, and validation/verification, as this chapter will show in detail.

Unfortunately, the vast literature on simulation does not provide a standard definition of *sensitivity analysis*. In this chapter, sensitivity analysis is interpreted as the systematic investigation of the reaction of the simulation responses to *extreme* values of the model's input or to *drastic* changes in the model's structure. For example, what happens to the customers' mean waiting time when their arrival rate doubles; what happens if the priority rule changes from FIFO to SPT? So this chapter does not focus on marginal changes or perturbations in the input values.

For this what-if analysis, DOE uses *regression analysis*, also known as Analysis Of Variance or ANOVA. This analysis is based on a *metamodel*, which is defined as a model of the underlying simulation model; see Kleijnen (1975b), Friedman (1996). In other words, a metamodel is an approximation of the simulation program's I/O transformation; it is also called a response surface. Typically, this regression metamodel belongs to one of the following three classes: (i) a first-order polynomial, which consists of main effects only, besides an overall or grand mean; (ii) a first-order polynomial augmented with interactions between pairs of factors (two-factor interactions); (iii) a second-order polynomial, which also includes purely quadratic effects; also see equation (1) in §3.2.

Most simulation models have *multiple outputs*, also called responses or criteria. For example, outputs may be both customer's queueing time and server's idle time, or both mean and 90% quantile of the waiting time. In practice, multiple outputs are handled through the application of the techniques of this chapter *per* output type. (Ideally, however, such simulations should be studied through multi-variate regression analysis, and the design should also account for the presence of multiple outputs; see Khuri (1996), Kleijnen (1987).) Simultaneous inference may be taken care of through Bonferroni's inequality; see appendix 1. Optimization in the presence of multiple responses will be discussed in §5. Optimization accounting for both the mean and the variance of the response is the focus of Taguchi's methods; see Ramberg et al. (1991). Note that the term 'multiple regression analysis' refers -not to the number of outputs- but to the presence of multiple inputs -or better- multiple independent variables; also see the definition of z below (4) in §3.2.

A metamodel treats the simulation model as a *black box*: the simulation model's inputs and outputs are observed, and the factor effects in the metamodel are estimated. This approach has the following *advantages* and *disadvantages*.

An advantage is that DOE can be applied to all simulation models, either deterministic or stochastic. Further, DOE gives better estimates of the factor effects than does the intuitive approach often followed in practice, namely the one-factor-at-a-time approach, which will be discussed in §4.2 (on resolution-3 designs).

A drawback is that DOE cannot take advantage of the specific structure of a given simulation model, so it takes more simulation runs than do *perturbation analysis* and modern *importance sampling*, also known as *likelihood ratio* or *score function*. These alternative methods usually require a single run (by definition, a simulation run is a single time path with fixed values for all its inputs and parameters). Such a run, however, may be much longer than a run in DOE. Moreover, these alternatives require more mathematical sophistication, and they must satisfy more mathematical assumptions. Importance sampling as a variance reduction technique (not a what-if technique) will be discussed in §3.3 and Appendix 2. There is much literature on these alternative methods: see Chapter 9 (by Andradottir) of this handbook, and also Ho and Cao (1991), Glynn and Iglehart (1989), Kleijnen and Rubinstein (1995), and Rubinstein and Shapiro (1993).

DOE may be used not only for sensitivity analysis and optimization, but also for *validation*. This chapter will address only part of the validation and verification problem (see §6). A detailed discussion of validation and verification can be found in Chapter 10 (by Balci) of this handbook.

In summary, this chapter discusses DOE as a method for answering what-if questions in simulation. It is not surprising that DOE is important in simulation: by definition, simulation means that a model is used, not for mathematical analysis or numerical methods, but for experimentation. But experimentation requires a good design and a good analysis!

DOE with its concomitant regression analysis is a standard topic in statistics. However, the standard statistical techniques must be adapted such that they account for the *peculiarities of simulation*:

(i) There are a great many factors in many practical simulation models. Indeed, the ecological case study (mentioned above) has 281 factors, whereas standard DOE assumes only up to (say) fifteen factors.

(ii) Stochastic simulation models use pseudorandom numbers, which means that the analysts have much more control over the noise in their experiments than the investigators have in standard statistical applications. For example, common and antithetic seeds may be used; see §3.5 and Appendix 2.

(iii) Randomization is of major concern in DOE outside simulation: assign the 'experimental units' (for example, patients) to the treatments (say, types of medication) in a random, non-systematic way so as to avoid bias (healthy patients receive medication of type 1 only). In simulation, however, this randomization problem disappears: pseudorandom number streams take over.

(iv) Outside simulation the application of blocking is an important technique to reduce systematic differences among experimental units; for example, tire wear differs among the four positions on the car: left front, ..., right rear. In simulation, however, complete control over the experiment eliminates the need for blocking. Yet, the blocking concept may be used to assign common and antithetic pseudorandom numbers, applying the Schruben-Margolin approach; see §3.5.

The *main conclusions* of this chapter will be:

(i) Screening may use a novel technique, called sequential bifurcation, that is simple, efficient, and effective.

(ii) Regression metamodeling generalizes the results of a simulation experiment with a small number of factors, since a regression metamodel estimates the I/O transformation specified by the underlying simulation model.

(iii) Statistical designs give good estimators of main (first-order) effects, interactions between factors, and quadratic effects; these designs require fewer simulation runs than intuitive designs do.

(iv) Optimization may use RSM, which combines regression analysis and statistical designs with steepest ascent; see (ii) and (iii).

(v) Validation may use regression analysis and statistical designs.

(vi) These statistical techniques have already been applied many times in practical simulation studies, in many domains; these techniques make simulation studies give more general results, in less time.

The remainder of this chapter is *organized* as follows.

§2 discusses the screening phase of a simulation study. After an introduction (§2.1), another subsection (§2.2) discusses a special screening technique, namely sequential bifurcation.

§3 discusses how to approximate the I/O transformation of simulation models by regression analysis. First it discusses graphical methods, namely scatter plots; see §3.1. Next §3.2 presents regression analysis, which formalizes the graphical approach, including Generalized Least Squares (GLS). Then §3.3 shows how to estimate the variances of individual simulation responses, including means, proportions, and quantiles, in either steady-state or transient state simulations. These estimates lead to Estimated GLS: see §3.4. Variance reduction techniques (VRTs), such as common random numbers, complicate the regression analysis; VRTs are discussed in §3.5. The estimated regression model may inadequately approximate the underlying simulation model: §3.6 gives several lack of fit tests. This section closes with a numerical example in §3.7 and a case study in §3.8.

§4 discusses statistical designs. After an introduction (§4.1), the focus is first on designs that assume only main effects (§4.2). Then follow designs that give unbiased estimators for the main effects, even if there are interactions between factors (§4.3). Further, this section discusses designs that allow estimation of interactions between pairs of factors (§4.4), interactions among any subset of factors (§4.5), and quadratic effects (§4.6). All these designs are based on certain assumptions; how to satisfy these assumptions is discussed in §4.7. 'Optimal' designs are discussed in §4.8. This section ends with a case-study in §4.9.

§5 covers optimization of simulated systems. RSM is discussed in §5.1. Two case studies are summarized in §5.2 and §5.3. §6 proceeds with the role of sensitivity analysis in validation, emphasizing the effects of data availability. §7 gives a summary and conclusions.

Three appendixes cover tactical issues, besides the strategic issues addressed by DOE. Appendix 1 summarizes confidence intervals for expected values, proportions, and quantiles, in terminating and steady-state simulations; also see Chapter 7 (by Alexopoulos and Seila). This appendix makes this chapter self-sufficient. Appendix 2 gives more details on VRTs, because VRTs are important when designing simulation experiments. This appendix covers four VRTs, namely common pseudorandom numbers, antithetic numbers, control variates or regression sampling, and importance sampling. Appendix 3 discusses jackknifing, which is a general method that may reduce bias of estimated simulation responses and may give robust confidence intervals.

References conclude the chapter. Nearly one hundred references are given. To reduce the number of references, only the most recent references for a topic are given, unless a specific older reference is of great historical value.

2. Screening

2.1 Introduction

In the pilot phase of a simulation study there are usually a great many potentially important factors; for example, the ecological case study has 281 parameters. It is the mission of science to come up with a *short list* of the most important factors: this is sometimes called the principle of parsimony or Occam's razor.

Obviously a full factorial design requires an astronomically great number of factor combinations: in the case study, at least 2^{281} ($> 10^{84}$). Even a design with only as many combinations as there are factors (see resolution-3 designs in §4.2) may require too much computer time. Therefore many practitioners often restrict their study to a few factors, usually no more than fifteen. Those factors are selected through intuition, prior knowledge, and the like. The factors that are ignored (kept constant), are -explicitly or implicitly- assumed to be unimportant. For example, in queueing networks the analysts may assume equal service rates for different servers. Of course, such an assumption severely restricts the generality of the conclusions from the simulation study!

The statistics literature does include screening designs: random designs, supersaturated designs, group screening designs, and so on; see Kleijnen (1987). Unfortunately, the statistics literature pays too little attention to screening designs. The reason for this neglect is that outside simulation it is virtually impossible to control hundreds of factors; (say) fifteen is hard enough.

In simulation, however, models may have hundreds of parameters, and yet their control is simple: specify which combinations of parameter values to simulate. Nevertheless, screening applications in simulation are still scarce, because most analysts are not familiar with these designs. Recently, screening designs have been improved and new variations have been developed; details are given in Bettonvil and Kleijnen (1997), Saltelli, Andres, and Homma (1995). The next subsection covers a promising screening technique, namely sequential bifurcation.

2.2 Sequential Bifurcation

Sequential bifurcation is a group-screening technique; that is, it uses *aggregation* (which is often applied in science when studying complicated systems). Hence, at the start of the simulation experiment, sequential bifurcation groups the individual factors into clusters. A specific group is said to be at its 'high' level (denoted as + or *on*), if each of its components or individual factors is at a level that gives a higher or equal response (not a lower response). Analogously, a group is at its low level (- or *off*), if each component gives a lower or equal response. To know with certainty that individual factor effects within a group do not cancel out, sequential bifurcation must assume that the analysts know whether a specific individual factor has a positive or negative effect on the simulation response; that is, the factor effects have *known signs*. In practice this assumption may not be too restrictive. For example, in specific queueing simulations it may be known that increasing the service rates while keeping all other factors constant (‘ceteris paribus’ assumption) decreases waiting time (but it is unknown how big this decrease is; therefore the analysts use a simulation model). In the ecological case study the experts could indeed specify in which direction a specific parameter affects the response (CO_2 concentration). Moreover, if a few individual factors have unknown signs, then these factors can be investigated separately, outside sequential bifurcation!

Sequentialization means -by definition- that factor combinations to be simulated are selected as the experimental results become available. So, as simulation runs are executed, insight into factor effects is accumulated and used to select the next run. It is well known that in general, sequentialization requires fewer observations; the price is a more cumbersome analysis and data handling. Sequential bifurcation eliminates groups of factors as the experiment proceeds, because the procedure concludes that these clusters contain no important factors

Also, as the experiment proceeds, the groups become smaller. More specifically, each group that seems to include one or more important factors, is split into two subgroups of the same size: *bifurcation*. At the end of bifurcation, individual factor effects are estimated.

As a numerical example, consider a simple academic example with 128 factors, of which only three factors are important, namely the factors #68, #113, and #120. Let the symbol $y_{(h)}$ denote the simulation output when the factors 1, ..., h are switched on and the remaining factors (h + 1, ..., k) are off. Consequently, the sequence $\{y_{(h)}\}$ is non-decreasing in h. The main effect of factor h is denoted by β_h (also see (1) in §3.2). Let the symbol $\beta_{h-h'}$ denote the sum of individual effects β_h through $\beta_{h'}$ with $h' > h$; for example, β_{-128} denotes the sum of β_1 through β_{128}; see Figure 1, line 1.

At the start (stage #0) of the procedure, sequential bifurcation always observes the two "extreme" factor combinations, namely $y_{(0)}$ (no factor high) and $y_{(k)}$ (all factors high). The presence of (three) important factors gives $y_{(0)} < y_{(128)}$. Hence sequential bifurcation infers that the sum of all individual main effects is important: $\beta_{1\text{-}128} > 0$. Sequential bifurcation works such that any important sum of effects leads to a new observation that splits that sum into two sub-sums: see the symbol ↓ in Figure 1. Because stage #0 gives $\beta_{1\text{-}128} > 0$, sequential bifurcation proceeds to the next stage.

Stage #1 gives $y_{(64)}$. The analysis first compares $y_{(64)}$ with $y_{(0)}$, and notices that these two outputs are equal (remember that only factors #68, #113, and #120 are important). Hence the procedure concludes that the first 64 individual factors are unimportant! So after only three simulation runs and based on the comparison of two runs, sequential bifurcation eliminates all factors in the first half of the total group of 128 factors. Next, sequential bifurcation compares $y_{(64)}$ with $y_{(128)}$, and notices that these two outputs are not equal. Hence the procedure concludes that the second subgroup of 64 factors is important; that is, there is at least one important factor in the second half of the group of 128 factors.

In stage #2 sequential bifurcation concentrates on the remaining factors (#65 through #128). That subgroup is again bifurcated, and so on. At the end, sequential bifurcation finds the three important factors (#68, #113, #120). In total, sequential bifurcation requires only 16 observations. The procedure also determines the individual main effects of the important factors: see the symbol ↑ in the last line of Figure 1.

INSERT FIGURE 1: Finding $k = 3$ important factors among $K = 128$ factors in Jacoby and Harrison's (1962) example

It can be proven that, if the analysts assume that there are interactions between factors, then the number of runs required by sequential bifurcation is double the number required in the main-effects only case (also see the foldover principle in §4.3). In general, it is wise to accept this doubling, in order to obtain estimators of main effects that are not biased by interactions between factors.

The ecological case study also uses the sequential bifurcation algorithm that was indicated in Figure 1. It took 154 simulation runs to identify and estimate the 15 most important factors among the original 281 factors. Some of these 15 factors surprised the ecological experts, so sequential bifurcation turns out to be a powerful statistical (black box) technique. Notice that on hindsight it turned out that there are no important interactions between factors, so only 154/2 = 77 runs would have sufficed.

Another case study is the building thermal deterministic simulation in De Wit (1997). In his simulation, sequential bifurcation gave the 16 most important inputs among the 82 factors after only 50 runs. He checked these results by applying a different screening technique, namely 'randomized one-factor-at-a time designs' of Morris (1991), which took 328 runs.

These two case studies concern deterministic simulation models. In *stochastic simulation* the signal/noise ratio can be controlled by selecting an appropriate run length. Then sequential bifurcation may be applied as in deterministic simulation. Bettonvil (1990, pp. 49-142) further investigates the use of sequential bifurcation in random simulation. More research is needed to find out whether in practice this method performs well in random simulations. Also see Cheng (1997).

3. Regression Metamodels

3.1 Introduction: Graphical Methods

Suppose the number of factors to be investigated is small, for example, fifteen (this small number may be reached after a screening phase; see §2). Suppose further that the simulation has been run for several combinations of factor levels. How should these I/O data be analyzed?

Practitioners often make a *scatter plot*, which has on the x-axis the values of one factor (for example, traffic rate) and on the y-axis the simulation response (say, average waiting time). This graph indicates the I/O transformation of the simulation model treated as a black box! The plot shows whether this factor has a positive or negative effect on the response, and whether that effect remains constant over the domain (experimental area) of the factor.

The practitioners may further analyze this scatter plot: they may fit a curve to these (x, y) data; for example, a straight line (say) $y = \beta_0 + \beta_1 x$. Of course, they may fit other curves (such as a quadratic curve: second degree polynomial), or they may use paper with one or both scales logarithmic.

To study interactions between factors, the practitioners may combine several of these scatter plots (each drawn per factor). For example, the scatter plot for different traffic rates was drawn, given a certain priority rule (say) FIFO. Plots for different priority rules can now be superimposed. Intuitively, the average waiting time curve for SPT lies below the curve for FIFO (if not, either this intuition or the simulation model is wrong; see the discussion on validation in §6). If the response curves are not parallel, then by definition there is interaction between priority rule and traffic rate.

However, superimposing many plots is cumbersome. Moreover, their interpretation is subjective: are the response curves really parallel and straight lines? These shortcomings are removed by regression analysis. Also see Kleijnen and Van Groenendaal (1992), and Kleijnen and Sargent (1997).

3.2 GLS, OLS, WLS

A regression metamodel may be used to approximate the I/O transformation of the simulation model that generates the data to which the regression analysis is applied. (A different use of regression analysis is to obtain control variates, which are a specific type of VRT; see (15) in §3.5.) Draper (1994) gives a bibliography on applied regression analysis, outside simulation.

Consider the *second degree polynomial*

$$Y_{i,j} = \beta_0 + \sum_{h=1}^{k} \beta_h x_{i,h} + \sum_{h=1}^{k} \sum_{h'=h}^{k} \beta_{h,h'} x_{i,h} x_{i,h'} + E_{i,j} \qquad (1)$$
$$(i = 1, \ldots, n;\ j = 1, \ldots, m_i)$$

where stochastic variables are shown as upper case letters; the specific symbols are

$Y_{i,j}$: simulation response of factor combination i, replication j
k: number of factors in the simulation experiment
β_0: overall mean response or regression intercept
β_h: main effect or first-order effect of factor h
x_{ih}: value of 'standardized' factor h in combination i (see (2) below)
$\beta_{h,h'}$: interaction between factors h and h' with $h < h'$
$\beta_{h,h}$: quadratic effect of factor h
$E_{i,j}$: fitting error of the regression model in combination i, replication j
n: number of simulated factor combinations
m_i: number of replications for combination i.

To interpret this equation, it is convenient to first ignore interactions and quadratic effects. Then the *relative importance* of a factor is obtained by sorting the absolute values of the main effects β_h, provided the factors are *standardized*. Let the original (non-standardized) factor h be denoted by w_h. In the simulation experiment w_h ranges between a lowest value l_h and an upper value u_h; that is, the simulation model is not valid outside that range (see the discussion on validation in §6) or in practice that factor can range over that domain only (for example, because of space limitations the number of servers can vary only between one and five). Measure the variation or spread of that factor by the half-range $a_h = (u_h - l_h)/2$, and its location by the mean $b_h = (u_h + l_h)/2$. Now the following standardization is appropriate:

$$x_{ih} = (w_{ih} - b_h)/a_h \ . \qquad (2)$$

Notice that importance and significance are related, but different concepts. Significance is a statistical concept. An important factor may be declared non-significant if the variance of the estimated effect is high (because the simulation response has high variance σ^2 and the total sample size $N = \sum_{i=1}^{n} m_i$ is small): this is called the type II or β error. Importance depends on the practical problem that is to be solved by simulation. An unimportant factor may be declared significant if the variance of the estimated factor effect is small (in simulation large sample sizes do occur).

It is convenient to use the following notation. Denote the *general linear regression model* by

$$Y_{i,\,j} = \sum_{g=1}^{q} \gamma_g z_{ig} + E_{i,\,j} \; . \tag{3}$$

Comparison of (1) and (3) gives the following identities: $\gamma_1 = \beta_0$, $\gamma_2 = \beta_1$, …, $\gamma_q = \beta_{k,\,k}$, and $z_{i,\,1} = 1$, $z_{i,\,2} = x_{i,\,1}$, …, $z_{i,\,q} = x_{i,\,k}^2$. It is also convenient to use matrix notation for the preceding equation, (3):

$$\boldsymbol{Y} = \boldsymbol{\gamma z} + \boldsymbol{E} \tag{4}$$

where bold symbols denote matrices including vectors (vectors are matrices with a single column; in linear algebra it is customary to denote matrices by upper case letters and vectors by lower case letters, but in this chapter upper case letters denote random variables); the specific symbols are

N: total number of simulation responses, $\sum_{i=1}^{n} m_i$;

Y: vector with *N* components; the first m_1 elements are the m_1 replicated simulation responses for input combination 1, …, the last m_n elements are the m_n replications for input combination *n*;

γ: vector of *q* regression parameters, $(\gamma_1, \gamma_2, \ldots, \gamma_g, \ldots, \gamma_q)'$;

z: *N* by *q* matrix of independent variables; the first m_1 rows are the same and denote factor combination 1, …, the last m_n rows are the same and denote factor combination *n*;

E: vector with *N* fitting errors, $(E_{1,\,1}, \ldots, E_{n,\,m_n})$.

An alternative notation is

$$\bar{\boldsymbol{Y}} = \boldsymbol{\gamma}\bar{\boldsymbol{z}} + \bar{\boldsymbol{E}} \tag{5}$$

where the bar denote the average per factor combination; each matrix (including vector) has now only *n* (instead of *N*) rows. For example, the first element of the vector $\bar{\boldsymbol{Y}}$ is $\bar{Y}_1 = \sum_{j=1}^{m_1} Y_{1,\,j}/m_1$.

Note that non-linear regression models and the concomitant DOE are discussed in Ermakov and Melas (1995, pp.167-187).

The *Generalized Least Squares* (GLS) estimator of the parameter vector ***γ***, denoted by $\boldsymbol{C}_{GLS} = (C_1, C_2, \ldots, C_g, \ldots, C_q)'$, is

$$\boldsymbol{C}_{GLS} = (\boldsymbol{z}'\boldsymbol{\sigma}_y^{-1}\boldsymbol{z})^{-1}\boldsymbol{z}'\boldsymbol{\sigma}_y^{-1}\boldsymbol{Y} \tag{6}$$

where $\boldsymbol{\sigma}_y$ denotes the *N* by *N* covariance matrix of ***Y***, which is assumed to be non-singular so that its inverse $\boldsymbol{\sigma}_y^{-1}$ exists (in §3.5 this assumption is revisited for the estimate $\hat{\boldsymbol{\sigma}}_y^{-1}$); it is further assumed that ***z*** is such that $\boldsymbol{z}'\boldsymbol{\sigma}_y^{-1}\boldsymbol{z}$ is regular (further discussion on ***z*** will follow in §4, on DOE); to simplify the notation, this chapter will sometimes denote $\boldsymbol{C}_{GLS}$ by ***C***. It is not too difficult to derive an alternative expression for the GLS estimator that uses the alternative notation in (5); see Kleijnen (1987, p. 195) and Kleijnen (1992).

GLS gives the *best linear unbiased estimator* (BLUE), where 'best' means minimum variance. The individual variances can be found on the main diagonal of the covariance matrix for ***C***:

$$\boldsymbol{\sigma}_c = (z'\boldsymbol{\sigma}_y^{-1}z)^{-1} \ . \qquad (7)$$

A (1 - α) confidence interval per parameter follows from the well-known Student statistic. The general formula for this statistic (which will be used repeatedly below) is

$$T_\nu = \frac{\bar{Y} - E(Y)}{S_{\bar{y}}} \qquad (8)$$

where ν denotes the number of degrees of freedom. Notice that a statistic is called studentized if the numerator is divided by its standard error or standard deviation. For the GLS estimator, Y in (8) is replaced by C_g (g = 1, ..., q) and $S_{\bar{y}}$ by the estimator of the standard deviation of C_g; this estimator will be discussed in the next subsection, §3.3. Software for GLS estimation is abundant; see Swain (1996).

A special, classic case of GLS is *Ordinary Least Squares* (OLS). OLS remains BLUE if the simulation responses have *white noise*; that is, they are independent and have constant variances; that is, $\boldsymbol{\sigma}_y = \sigma^2\mathbf{1}_{N\times N}$ where $\mathbf{1}_{N\times N}$ denotes the N by N identity matrix (usually denoted by I in linear algebra). Then the GLS estimator in (6) reduces to

$$\boldsymbol{C}_{OLS} = (z'z)^{-1}z'\boldsymbol{Y} \ . \qquad (9)$$

Even if the OLS assumptions do not hold, an alternative to GLS is the OLS point estimator in (9), but with the correct covariance matrix

$$\boldsymbol{\sigma}_{c_{ols}} = (z'z)^{-1}z\boldsymbol{\sigma}_y z'(z'z)^{-1} \ . \qquad (10)$$

If the OLS assumptions ($\boldsymbol{\sigma}_y = \sigma^2\mathbf{1}_{N\times N}$) do hold, then this equation (10) simplifies to $\boldsymbol{\sigma}_{c_{ols}} = (z'z)^{-1}\sigma^2$. The white noise assumption may be used in deterministic simulation; see the case study in §3.8.

In random simulation, however, it is realistic to assume that the response variances vary with the input combinations: var($Y_{i,j}$) = σ_i^2. (So $Y_{i,j}$ has a mean and a variance that both depend on the input.) In other words, $\boldsymbol{\sigma}_y$ (the covariance matrix for the simulation responses) becomes a diagonal matrix with the first m_1 elements equal to σ_1^2, ..., the last m_n elements equal to σ_n^2. Then a special case of GLS applies, namely *Weighted Least Squares* (WLS): substitute this diagonal matrix $\boldsymbol{\sigma}_y$ into the GLS formulas in (6) and (7). The interpretation of the resulting formula is that WLS uses the standard deviations σ_i as weights. In case of variance heterogeneity, the WLS estimator is BLUE; both OLS and WLS still give unbiased estimators.

3.3 Estimation of Response Variances

The preceding formulas feature $\boldsymbol{\sigma}_y$ (the covariance matrix of the simulation responses), which unfortunately is unknown (as are the means E(Y_i) = μ_i). The estimation of the standard deviations σ_i is a classic *tactical* problem in simulation. To solve this problem, the analysts should distinguish terminating and steady-state simulations (see Chapter 7, by Alexopoulos and Seila), and expected values, proportions, and quantiles. This subsection considers four cases: (i) mean response of terminating simulation, (ii) mean response of steady-state simulation, (iii) proportions, and (iv) quantiles.

(i) Mean response of terminating simulation

By definition, a terminating simulation has an event that stops the simulation run; an example is a queueing simulation of a bank that opens at 9 a.m. and closes at 4 p.m. In such a situation, the simulation runs for one specific combination (say) i may give independently and identically distributed (i.i.d.) responses Y_i, for example, average waiting time per day. Then the estimator of the variance of Y_i, given a sample size of m_i replications (for example, m_i days) with integer $m_i \geq 2$, is

$$S_i^2 = \sum_{j=1}^{m_i} (Y_{i,j} - \bar{Y}_i)^2/(m_i - 1) \tag{11}$$

where the average of the m_i replications is $\bar{Y}_i = \sum_{j=1}^{m_i} Y_{i,j}/m_i$; each replication uses a non-overlapping sequence of pseudorandom numbers.

Note that in the simulation literature on tactical issues it is classic to focus on (1- α) confidence intervals. If a Gaussian distribution is assumed ($N(\mu_i, \sigma_i^2)$), then the confidence interval should use the Student statistic in (8), now with $\nu = m_i - 1$ degrees of freedom. If, however, Y_i has an asymmetric distribution, then Johnson's modified Student statistic is a good alternative; this statistic includes an estimator for the skewness of the distribution of Y_i; see Appendix 1. One more alternative is a distribution-free or non-parametric confidence interval such as the sign test or the signed rank test. Conover (1971) gives an excellent discussion of distribution-free statistics. Kleijnen (1987) discusses the application in simulation. Other alternatives are jackknifing and bootstrapping; see the next subsection, §3.4. Appendix 1 gives more details on (1- α) confidence intervals for the response of an individual factor combination. (This appendix covers 'per comparison' intervals, not 'familywise' or 'experimentwise' intervals; see Bonferroni's inequality at the end of Appendix 1, and Chapter 8, by Goldsman and Nelson). In DOE, however, confidence intervals are desired, not for the individual simulation responses, but for the individual factor effects. For that purpose, the variances of the simulation responses are needed.

(ii) Mean response of steady-state simulation

Some practical problems require steady-state simulations; examples are strategic decisions on production-facility layout, assuming static environments and long-term reward systems. Suppose the simulationists execute a single long run (not several replicated runs) per factor combination. Assume that a simulation run yields a time series that (possibly, after elimination of the start-up phase) gives a stationary process in the wide sense. Then there are several methods for the estimation of the variance σ_i^2: batching (or subruns), renewal (or regenerative) analysis, spectral analysis, standardized time series; see Chapter 7, by Alexopoulos and Seila. Appendix 1 gives formulas for renewal analysis only.

(iii) Proportions

Let p_a denote the probability of the response exceeding a given value a; for example, the probability of waiting time exceeding five minutes in a simulation run. This leads to a binomially distributed variable, and the estimation of its mean p_a and variance p (1 - p_a)/m. Actually, the subscript i must be added to the parameters p_a and m; see (11).

When estimating an extremely small probability, importance sampling is needed: the probability distribution of some input variable is changed such that the probability of the event of interest increases; for example, the mean (say) $1/\lambda$ of the Poisson service process is increased so that the probability of buffer overflow increases; see Appendix 2 and also Heidelberger (1995) and Heidelberger, Shahabuddin, and Nicola (1996).

(iv) Quantiles

A response closely related to a proportion is a quantile: what is the value not exceeded by (say) 80% of the waiting times? Quantile estimation requires sorting the m observations y_j, which yields the order statistics $y_{(j)}$; that is, $y_{(1)}$ is the smallest observation, ..., $y_{(m)}$ is the largest observation. Appendix 1 gives formulas. Notice that the median is a good alternative for the mean when quantifying the location of a random variable; the regression metamodel may explain how the various factors affect the median simulation response.

3.4 Estimated GLS and WLS

The cases (i) through (iv) in the preceding subsection (§3.3) show that there are different types of responses, each requiring its own procedures for the estimation of the variances σ_i^2. But whenever the analysts use estimated response variances, WLS becomes *Estimated WLS* or EWLS. This gives a *non-linear* estimator for the factor effects $\boldsymbol{\gamma}$; see (6) with $\boldsymbol{\sigma}_y$ replaced by a diagonal matrix $\boldsymbol{S}_y$ with main-diagonal elements S_i^2 defined in (11). But the properties of non-linear estimators are not well-known: is the estimator still unbiased; does it have minimum variance; what is its variance?

Jackknifing is a general computer-intensive technique that the analysts should consider whenever they presume the estimator under consideration to be biased or whenever they do not know how to construct a valid confidence intervals. In the EWLS case, *Jackknifed EWLS* or JEWLS may be defined as follows. Suppose there are m replications of Y_i. This yields (say) $\boldsymbol{C}_{EWLS}$, the EWLS estimator of the regression parameter γ. Next, for each factor combination eliminate one replication, say, replication j (with j = 1, ..., m). Calculate the EWLS estimator from the remaining (m - 1) replications; this gives m estimators $\boldsymbol{C}_{EWLS,\,-j}$. The pseudovalue (say) P_j is defined as the following linear combination of the original and the j^{th} estimator:

$$P_j = m\boldsymbol{C}_{EWLS} - (m - 1)\boldsymbol{C}_{EWLS,\,-j} \; (j = 1, \ldots, m). \tag{12}$$

The jackknifed estimator is defined as the average pseudovalue:

$$\bar{P} = \sum_{j=1}^{m} P_j/m \; . \tag{13}$$

If the original estimator is biased, then the jackknifed estimator may have less bias, albeit it at the expense of a higher variance. Moreover, jackknifing gives the following robust confidence interval. Treat the m pseudovalues P_j as m i.i.d. variables: see the Student statistic in (8), now with Y replaced by P and $\nu = m - 1$.

Efron (1982) and Miller (1974) give classic reviews of jackknifing. Jackknifing of EWLS is further discussed in Kleijnen, Karremans, Oortwijn, and Van Groenendaal (1987). Jackknifed renewal analysis is discussed in Kleijnen and Van Groenendaal (1992, pp. 202-203); also see Appendix 3. Jackknifing is related to *bootstrapping*, which samples from the set of m observations; see Cheng (1995), Efron (1982), and Efron and Tibshirani (1993).

3.5 Variance Reduction Techniques

VRTs are supposed to decrease the variances of estimators such as the estimated mean simulation response, the estimated differences among mean simulation responses, and the estimated factor effects (which are linear combinations of simulation responses; see the GLS estimator in (6)). This subsection covers the following three VRTs that have relations with GLS: (i) common random numbers, (ii) antithetics, and (iii) control variates.

(i) Common random numbers

Practitioners often use common (pseudo)random numbers to simulate different factor combinations (see Appendix 2 for details). When responses are statistically dependent, GLS gives BLUE. But in practice the covariances between simulation responses of different factor combinations are unknown, so these covariances must be estimated. Assume m independent replications per factor combination, so Y_{ij} and $Y_{i'j}$ are dependent but Y_{ij} and $Y_{ij'}$ are not , when $j \neq j'$ and $j' = 1, \ldots, m$. Then the classic covariance estimator is

$$S_{i,\,i'} = \sum_{j=1}^{m} (Y_{i,\,j} - \bar{Y}_i)(Y_{i',\,j} - \bar{Y}_{i'})/(m - 1). \qquad (14)$$

Notice that this equation reduces to the classic variance estimator in (11) if $i = i'$ and S_{ii} is defined as S_i^2. Dykstra (1970) proves that the estimated covariance matrix S_y is singular if $m \leq n$. *Estimated GLS* or EGLS gives good results; see Kleijnen (1992). A detailed example is given in Chapter 8, by Goldsman and Nelson.

The simulation literature has ignored the estimation of covariances in *steady-state* simulations. If renewal analysis is used, then the renewal cycles get out of step; for example, when a factor combination uses a lower traffic rate, its cycles get shorter. However, if subruns are used, then the estimators are strictly analogous to the preceding equation, (14).

(ii) Antithetics random numbers

Closely related to common pseudorandom numbers are antithetic pseudorandom numbers: to realize negative correlation between pairs of replications, use the pseudorandom numbers r for one replication and the complements or 'antithetics' $1 - r$ for the other replication. Then m replications give $m/2$ independent pairs $(Y_1, \ldots, Y_{m/2})$ where $Y_i = (Y_{2i-1} + Y_{2i})/2$. Hence to estimate the variance, use (11) but replace m_i by $m/2$ and Y_j by Y_r with $r = 1, \ldots, m/2$.

Notice that these antithetic pseudorandom numbers do create negative correlation between two variables sampled from the same distribution if this sampling uses the *inverse transformation* technique: $X = \mathrm{f}(R)$ where f denotes the inverse (cumulative) distribution function (which is monotonically increasing). An example is provided by the exponential distribution: $X = -\ln(R)/\lambda_x$ with $1/\lambda_x = \mathrm{E}(X)$. A counter-example is provided by the sampling of two independent standard normal variables (say) V_1 and V_2 through the well-known Box-Muller transformation: $V_1 = \cos(2\pi R_1)\{-2\ln(R_2)\}^{1/2}$ and $V_2 = \sin(2\pi R_1)\{-2\ln(R_2)\}^{1/2}$. But then the basic idea can still be applied: $X_1 = \mathrm{E}(X) + \sigma_x V_1$ and $X_2 = 2\mathrm{E}(X) - X_1$ (V_2 can be used for the next sample of the pair, X_1 and X_2). Appendix 2 gives some details on antithetic variates.

Since common and antithetic pseudorandom numbers are so closely related, Kleijnen (1975c) investigated their combination. Later Schruben and Margolin (1978) examined this combination in the case of a first-order polynomial metamodel. Their rule is:

treat the selection of common and antithetic seeds as a separate factor -say, factor k- in a two-level design. Associate one level of that factor with common seeds; that is, use common seeds for the $n/2$ combinations that have factor k at its plus level. Associate the other level with antithetic seeds; that is, use the same antithetic seeds for the remaining $n/2$ combinations that have factor k at its minus level. Later, the Schruben-Margolin strategy was further investigated for second-order metamodels; see Donohue (1995).

(iii) Control variates

Regression models can be used not only as metamodels for what-if questions, but also as a VRT: control variates or regression sampling. Whereas antithetics makes the companion replication compensate 'overshoot' (that is, $y > \mathrm{E}(Y)$), control variates corrects the response of a given replication as follows.

A random variable (say) X can serve as a control variate if its mean $\mathrm{E}(X)$ is known and it is strongly correlated with the response Y: $|\rho_{x,\,y}| >> 0$ where $\rho_{x,\,y}$ denotes the linear correlation coefficient between X and Y. As an example, consider the simulation of an M/M/1 queueing system. On first reading the subscript i may be ignored in the following definitions and formulas; actually a control variate estimator may be computed for each factor combination i. Denote the average input #1 (say, average service time) per replication in combination i by $X_{i,\,1}$. Obviously this input and the output Y_i (say, either the average or the 90% quantile of waiting time in combination i) are positively correlated: $\rho(X_{i,\,1}, Y_i) \geq 0$. Hence in case of an overshoot, y should be corrected downwards:

$$\bar{Y}_{i,\,c} = \bar{Y}_i + C_{i,\,1,\,OLS}[E(\bar{X}_{i,\,1}) - \bar{X}_{i,\,1}] \qquad (15)$$

where $C_{\mathrm{i},\,1,\,OLS}$ denotes the OLS estimator when output Y_i is regressed on $X_{i,\,1}$, which denotes input #1 in combination i (this OLS estimator is computed from m_i replications of the pair $(Y_i,\ X_{i,\,1})$); the input averaged over m_i replicates is $\bar{X}_{i,\,1} = \sum_{j=1}^{m_i} X_{i,\,1,\,j}$. Therefore the technique of control variates is also called regression sampling. Notice that in this equation the input is a random variable (upper case X), whereas in the regression metamodel this input is fixed at its extreme values (lower case x; see (1)). Also see Appendix 2.

The single control variate estimator in the last equation, (15), can be extended to *multiple* control variates; for example, service time X_1 and arrival time (say) X_2. This requires multiple regression analysis. Actually, a better control variate may be traffic load, $\mathrm{E}(X_1)/\mathrm{E}(X_2)$. In general, the explanatory variables in this regression model may be selected such that the well-known multiple correlation coefficient R^2 is maximized. $R^2 = 1$ means perfect fit. However, because R^2 increases as the number of independent variables increases, the adjusted R^2 is preferred as criterion for the selection of control variates

A complication is that the estimator $C_{\mathrm{i},\,1,\,OLS}$ in (15) leads to a non-linear estimator (see the product $C_{\mathrm{i},\,1,\,OLS}\bar{X}_{i,\,1}$), which in general is biased. Moreover, the construction of a confidence interval for $\mathrm{E}(\bar{Y}_{i,\,c})$ becomes problematic. These problems can be solved, either assuming multivariate normality for $(Y_i,\ X_{i,\,1},\ X_{i,\,2},\ \ldots)$ or using the robust technique of jackknifing. Details of jackknifing the control variates are shown in Appendix 3.

The joint application of common, antithetic, and control variates is examined in Tew and Wilson (1994).

3.6 Lack of Fit of the regression metamodel

Once having specified a regression metamodel and having estimated its parameters -with or without applying VRTs- it becomes necessary to check possible lack of fit of the regression metamodel: is the estimated regression model an adequate approximation of the I/O transformation of the specific simulation model, given a specific experimental domain? In other words, the simulation model is supposed to be valid only within a certain area of its parameters and input variables; also see §6. Likewise, the metamodel is supposed to be valid only for those simulation I/O data that lead to its estimated parameter values. Consequently, the metamodel is more reliable when used for interpolation; it may be dangerous when used to extrapolate the simulated behavior far outside the domain simulated in the DOE.

Notice that in practice, analysts often try to interpret individual effects before they check whether the regression model as a whole makes sense. However, first the analysts should check if the estimated regression model is a *valid* approximation of the simulation model's I/O transformation. If the metamodel seems valid, its individual effects are to be examined.

To test the adequacy of the metamodel, this model might be used to predict the outcomes for *new* factor combinations of the simulation model. For example, in (1) replace $\boldsymbol{\beta}$ by its estimate, and substitute new combinations of $\boldsymbol{x}$ (remember there are n old combinations). Compare the predictions $\hat{y}$ with the simulation response y.

A refinement is *cross-validation*. The idea is as follows:
(i) eliminate one combination (say) i with $i = 1, \ldots, n$, instead of adding new combinations, which require more computer time;
(ii) re-estimate the regression model from the remaining n - 1 combinations;
(iii) repeat this elimination for all values of i.
Notice that cross-validation resembles jackknifing.

The formulas are as follows. The predictor for the simulation response given $\boldsymbol{C}$ (the estimator of the regression parameters or factor effects) and z_i (the vector of independent variables) is

$$\hat{Y}_i = z_i' \boldsymbol{C} \ . \tag{16}$$

The variance of this predictor is

$$var(\hat{Y}_i) = z_i' \sigma_C z_i \ . \tag{17}$$

The estimator of this variance follows from substitution of the estimator for $\boldsymbol{\sigma}_C$. Hence, for OLS the estimator of (17) follows immediately from (10), (11), and (14). For GLS the estimator of (17) might use (7), (11), and (14); this gives an asymptotically valid estimator.

After elimination of I/O combination i the new vector of estimated effects is based on the GLS formula in (6):

$$C_{-i} = (z'_{-i}\boldsymbol{\sigma}_{y_{-i}}^{-1}\mathbf{z}_{-i})^{-1}\mathbf{z}'_{-i}\sigma_{y_{-i}}^{-1}Y_{-i} \ . \tag{18}$$

Substitution into (16) gives the predictor $\hat{Y}_{-i} = y(\boldsymbol{C}_{-i}, z_{i})$. The Studentized cross-validation statistic (see (8)) is

$$\tilde{T}_{\nu} = \frac{Y_i - \hat{Y}_{-i}}{[\hat{var}(Y_i) + \hat{var}(\hat{Y}_{-i})]^{1/2}} \tag{19}$$

where $\hat{var}(Y_i) = S_i^2$ and $var(\hat{Y}_{-i})$ follows from (17), replacing $\boldsymbol{C}$ by $\boldsymbol{C}_{-i}$, etc. The degrees of freedom ν in (19) are unknown; Kleijnen (1992) uses $\nu = m - 1$. When this Studentized prediction error is significant, the analysts should revise the regression metamodel they specified originally. When judging this significance, they may apply Bonferroni's inequality, since there are multiple runs, namely m (see Appendix 1). In their revision, they may use transformations of the original inputs, such as logarithmic transformations and cross-products or interactions.

An alternative to cross-validation is *Rao's lack of fit test.* To understand this test, it is convenient to first consider the classic OLS case: normally distributed simulation responses Y_i with white noise. Then there are the following two estimators of the common response variance σ^2. The first estimator is based on replication: see the classic variance estimator S_i^2 defined in (11). Because the true variance is constant, these estimators are averaged or pooled: $\sum_{i=1}^{n} S_i^2/n$; if m_i is not constant, then a weighted average is used, with the degrees of freedom $m_i - 1$ as weights. Next consider the n estimated residuals, $\hat{E}_i = Y_i - \hat{Y}_i$ with $i = 1, \ldots, n$. These residuals give the second variance estimator, $\sum_{i=1}^{n} \hat{E}_i^2 m/(n - q)$. The latter estimator is unbiased if and only if (iff) the regression model is specified correctly; otherwise this estimator overestimates the true variance. Hence the two estimators are compared statistically through the well-known F-statistic, namely $F_{n - q, n(m - 1)}$.

Rao (1959) extends this test from OLS to GLS:

$$F_{n - q, m - n - q} = \frac{(m - n + q)}{(n - q)(m - 1)}(\bar{\boldsymbol{Y}} - \bar{\boldsymbol{z}}\boldsymbol{C})'\boldsymbol{S}_{\bar{y}}^{-1}(\bar{\boldsymbol{Y}} - \bar{\boldsymbol{z}}\boldsymbol{C}) \tag{20}$$

where the n estimated GLS residuals are collected in the vector $\bar{\boldsymbol{Y}} - \bar{\boldsymbol{z}}\boldsymbol{C}$, and where not only response variances are estimated but also response covariances, collected in the estimated covariance matrix $\boldsymbol{S}_{\bar{y}} = \boldsymbol{S}_y/m$; also see (14).

Kleijnen (1992) shows that Rao's test is better than cross-validation if the simulation responses are symmetrically distributed, for example, normally or uniformly distributed. Lognormally distributed responses, however, are better analyzed through cross-validation.

In *deterministic* simulation, studentizing the prediction errors as in (19) gives misleading conclusions. In such simulations the constant error σ^2 is estimated from the residuals. Hence, the worse the metamodel is, the bigger this estimate becomes. But then the denominator in (19) increases ($\hat{var}(Y_i) = 0$), so the probability of rejecting this false model decreases! Therefore relative prediction errors $\hat{y}_i/y_i$ are of more interest to practitio ners. Examples will be presented in the coal-transport and the FMS case-studies in §3.8 and §4.9.

It is also interesting to observe how the *estimated individual input effects* change, as combinations are deleted: see $c_{h,\,-i}$. Obviously, if the specified regression model is a good approximation, then the estimates remain stable. Examples will be given for the same coal-transport and FMS case-studies. Notice that the $q \times n$ matrix $\boldsymbol{C}_{-i} = (c_{h,\,-i})$ concerns the use of the simulation model and its concomitant metamodel for explanation, whereas the vector with the n elements $\hat{Y}_i/Y_i$ concerns the use for prediction purposes. Other diagnostic statistics are PRESS, DEFITS, DFBETAS, and Cook's D; see the general literature on regression analysis, and Kleijnen and Van Groenendaal (1992, p. 157).

3.7 Numerical Example: Multiple Server System

Kleijnen and Van Groenendaal (1992, pp. 150-151, 159-162) considers the well-known class of server systems with (say) s servers in parallel, and Markovian arrival and service processes: exponential interarrival times with rate λ and exponential service times with rate μ, which are i.i.d. These systems are denoted as M/M/s.

Suppose the response of interest is the steady-state mean queue length. Estimate this mean by the run average (say) $\bar{v}$. Start each run in the empty state. Stop each run after 2,000 customers. (Better solutions for these tactical problems are discussed in Appendix 1.)

Simulate six intuitively selected combinations of λ, μ, and s. Replicate each combination twenty times. This gives Table 1. (The next section will show that much better designs are possible.)

INSERT Table 1: Average queue length $\bar{v}$ of 2000 customers in M/M/s simulation started in empty state

Specify a regression metamodel for the M/M/s simulation model, with both the response and the inputs logarithmically transformed: see (1) with $Y = \ln(\bar{v})$, $x_1 = \ln(\lambda)$, $x_2 = \ln(\mu)$, and $x_3 = \ln(s)$. Use of the SAS package for the regression analysis of this problem gives Table 2.

INSERT Table 2: Regression analysis of M/M/s example, using standard SAS software

Table 2 shows $N = 6 \times 20 = 120$ and $q = 1 + 3 = 4$, so 116 degrees of freedom remain to estimate the common response variance under the classical OLS assumptions. CLS stands for Corrected Least Squares, and denotes the OLS point estimates with correct standard errors; see (10). The classical OLS computations give wrong standard errors for the estimated factor effects that are slightly smaller than the standard errors that use the unbiased estimated response variances.

The point estimates for OLS and EWLS do not differ much in this example: the standard errors are small, and both estimators have the same expectation. EWLS gives only slightly smaller standard errors. The explanation is that the logarithmic transformation reduces variance heterogeneity: estimated response variances range only between 0.17 and 0.26 (the next section, §4, will return to transformations).

All values for the multiple correlation coefficient R^2, adjusted or not, are very high. This numerical example does not formally test the goodness of fit. The individual input effects have the same absolute values, roughly speaking. Their signs are as expected intuitively. Kleijnen and Van Groenendaal (1992) also examine a simpler regression metamodel that uses a single input, namely the traffic rate $\lambda/(\mu s)$.

3.8 Case Study: Coal Transport

This subsection summarizes Kleijnen (1995d), which concerns the following real-life system. A certain British coal mine has three coalfaces, each linked to its own bunker. These bunkers have specific capacities. Each bunker receives coal from a single coalface, and discharges this coal onto a conveyor belt that serves all three bunkers. This belt transports the coal to the surface of the mine. Whenever a bunker is full, the corresponding coalface must stop; obviously this congestion decreases the efficiency.

Wolstenholme (1990, p. 115) considers three inputs, namely total belt capacity, maximum discharge rate per bunker, and bunker capacities (which are assumed to be equal). He further presents three control rules for managing the discharge rate of the bunkers. For example, under policy I the discharge rate of each bunker can only be either zero or maximal (no intermediate values). The maximum is used as long as there is coal in the bunker and room on the conveyor belt. For this chapter it suffices to understand that policies II and III are more sophisticated than policy I. Policy is a qualitative factor with three levels.

Vital questions are: what are the efficiency effects of changing inputs; are there interactions among inputs? So this case study is representative of many problems that arise in real life, especially in physical distribution and production planning.

Wolstenholme (1990) develops a *system dynamics* model, which is a particular type of simulation, namely deterministic non-linear difference equations with feedback relations. As software he uses STELLA, whereas Kleijnen (1995d) uses POWERSIM 1.1.

Kleijnen (1995d) runs eight combinations of the three quantitative inputs (w_1, w_2, w_3); the output is the efficiency y; see Table 3 (the selection of input combinations will be discussed in the next section, §4).

INSERT Table 3: Input/output per policy for Wolstenholme (1990)'s coal transport model

The problem is how to find a *pattern* in the I/O behavior of the simulation model. To solve this problem, Wolstenholme (1990, pp. 116-121) uses intuition and common sense, studying run after run. This section, however, uses regression metamodeling, as follows.

Because qualitative factors are slightly more difficult to represent in a regression model, the effects of the three quantitative factors are first examined per policy. Next, policy is incorporated as a factor.

Assume a regression metamodel with main effects and two-factor interactions: see (1) with $k = 3$ and no quadratic effects. Hence $q = 7$ effects need to be estimated. OLS is used, because the simulation model is deterministic.

To check the *validity* of this metamodel, R^2 and cross-validation are used. The first measure is computed by all standard statistical software, whereas the second measure is supported by modern software only. In cross-validation the regression model is estimated using only seven of the eight combinations in Table 3. First combination 1 is deleted, and the regression parameters are estimated from the remaining seven combinations. This new estimator (say) $\hat{\beta}_{-1}$ is used to predict the simulation response through $\hat{y}_1$; see (18) and (16) respectively. The actual simulation response is already known: y_1 = 55.78 for policy I; see Table 3. Hence prediction errors can be computed. In this case the eight relative prediction errors $\hat{y}_i/y_i$ vary between 0.77 and 1.19, for policy I.

As combinations are deleted, estimated individual input effects change. But the estimates remain stable if the specified regression model is a good approximation. Table 4 gives results for the metamodel with main effects only, still using the I/O data of Table 3. The three estimated main effects have the correct positive signs: increasing input capacities increase efficiency. Moreover, an intuitive analysis of the I/O data in Table 3 suggests that input #3 has more effect than input #1, which in turn exceeds the effect of input #2; the formal analysis agrees with this intuitive analysis.

INSERT Table 4: Estimates of main effects (β_h), upon deleting a combination, in policy I;

The I/O data in Table 3 are also analyzed through other regression metamodels. Searching for a 'good' regression model requires intuition, common sense, and knowledge of the underlying system that generated the I/O data, namely the System Dynamics model and the real system. This search receives more attention in econometrics than in DOE; also see the case-study in Van Groenendaal and Kleijnen (1996). To save space, these details are skipped. The final conclusion is that in this case study a regression model with the three main effects seems best: interactions turn out to be insignificant, whereas deleting the main effect of input #2 (which is the smallest estimated main effect) increases the relative prediction error.

Next *individual* estimated input effects are tested statistically, assuming white noise. Then classic OLS yields the standard errors $s(\hat{\beta}_h)$. To test if β_h is zero (unimportant main effect), OLS uses Student's t statistic. The critical value of this statistic is determined by the significance level α. A usual value is 0.10, but to reduce the probability of falsely eliminating important inputs, the value 0.20 is also used in this study. A higher α means a smaller critical value. This yields the blanks and the symbol * in Table 4.

Next consider *policy II*, for which no table is displayed. The best metamodel turns out to have main effects only for inputs #2 and #3: deleting the nonsignificant main effect of input #1, decreases the maximum relative prediction error; interactions are not significant.

For *policy III* a model with the three main effects gives a good approximation. Note that such simple metamodels do not always hold: in the case study on a Flexible Manufacturing System or FMS, only a regression model *with* interactions -in addition to main effects- gives valid predictions and sound explanations; see §4.9.

Finally consider regression metamodels with policy as a *qualitative factor*. So now there are three quantitative inputs, each simulated for only two values, and there is one qualitative factor with three 'levels', denoted by I, II, and III. Technically, regression analysis handles this qualitative factor through two binary (0, 1) variables; see Kleijnen (1987). Now the best regression model includes the main effects of all four factors. Policy III is the best policy; policy II is worse than policy I, even though policy I is the simplest policy. These regression results agree with an intuitive analysis of the I/O data in Table 3: calculate the efficiency per policy, averaged over all eight combinations of the three other factors> These averages are 72.50, 70.75, 78.75.

4. Design of Experiments

4.1 Introduction

The section on regression metamodels (§3) assumed that the matrix of independent variables **z** is such that the corresponding inverse matrices are not singular when computing GLS, WLS, and OLS point estimates and their estimated covariance matrices. An obvious condition seems to be that the number of observations is not smaller than the number of regression parameters. But does that mean $N \geq q$ or $n \geq q$ (with $N = \sum_{i=1}^{n} m_i$)?

Consider a simple example, namely a second-order polynomial with a single factor: $Y = \beta_0 + \beta_1 X_1 + \beta_{1,1} X_1^2$; see (1) with $k = 1$. Obviously, simulating only two values of X_1 -corresponding with $n = 2$- does not give a unique estimate of this regression model, whatever the number of replicates m is. Hence the condition is $n \geq q$, not $N \geq q$. Also see the alternative notation of the regression model in (5).

Which n combinations to simulate -provided $n \geq q$- can be determined such that the variances of the estimated factor effects are minimized. This is one of the main goals of the statistical theory on DOE; other goals follow in the subsection on optimal DOE, §4.8.

This section first covers *classical DOE*, which assumes white noise or $\boldsymbol{\sigma}_y = \sigma^2 \mathbf{1}_{N\times N}$ and a correctly specified regression model or $\mathrm{E}(E_{i,j}) = 0$ in (3). This gives well-known designs such as 2^{k-p} and central composite design; see §4.2 through §4.6. Next this section presents ways to design simulation experiments such that these assumptions do hold; see §4.7. *Optimal DOE* does account for heterogeneous variances and correlated simulation responses; see §4.8. An FMS case-study finishes this section; see §4.9.

This chapter does not cover all types of designs. For example, it does not discuss 'mixture' designs, which imply that the factor values are fractions that sum up to the value one: $\sum_{h=1}^{k} x_{i,h} = 1$; see Myers et al. (1989, p. 142). It does not cover Taguchi's DOE; see Donohue (1994). This section further assumes that the experimental area is a k-dimensional hypercube in the standardized factors. In practice, however, certain corners of this area may represent unfeasible combinations; see Nachtsheim (1987), Nachtsheim et al. (1996), and also Kleijnen (1987, p. 319).

Classical designs are tabulated in many publications. Two authoritative textbooks are Box and Draper (1987) and Box, Hunter, and Hunter (1978). Two textbooks focussed on DOE in simulation, are Kleijnen (1975a, 1987).

The analysts may also learn how to construct those designs; see the preceding references. The next subsections show which types of designs are available, but they only indicate how to construct them (see the 'generators' in §4.2).

Recently, software, including artificial intelligence and expert systems, has been developed to help the analysts specify these designs. Expert systems for DOE, however, are still in the prototype phase; see Nachtsheim et al. (1996). 'Non-intelligent' software for DOE outside the simulation domain is supported by several commercial software products, such as CADEMO, ECHIP, and RS/1; see ProGAMMA (1997), Nachtsheim (1987) and BBN (1989) respectively. The need for DOE software in simulation is articulated in a panel discussion at the 1994 Winter Simulation Conference; see Sanchez et al. (1994). DOE software for simulation is investigated in Ören (1993), Ozdemirel, Yurttas, and Koksal (1996), and Tao and Nelson (1997).

Simulation applications of classical designs are referenced in Donohue (1994), Kleijnen (1987, 1995c), and Kleijnen and Van Groenendaal (1992). As mentioned above, a FMS case-study will be presented in §4.9.

4.2 Main Effects Only: Resolution-3 Designs

According to Box and Hunter's (1961) definition, resolution-3 designs give unbiased estimators of the parameters of a first-order polynomial regression model. These parameters are the k main effects, plus the overall mean; see the first two terms in the right-hand side of (1). Sometimes these designs are called 'screening' designs; see Nachtsheim (1987, p. 133). This chapter, however, reserves the term 'screening' for designs with fewer runs than factors: $n < k$; see §2.

In practice, analysts often simulate the 'base' situation first, and next they change *one factor at a time*. This approach implies $n = 1 + k$. However, consider *orthogonal* designs, that is, designs that satisfy

$$\boldsymbol{d}'\boldsymbol{d} = n\mathbf{1}_{nxn} \tag{21}$$

with design matrix $\mathbf{d} = (d_{ij})$ and $i = 1,\dots, n$, $j = 1,\dots, k$, and $n > k$. Notice that the matrix of independent variables z becomes $(\mathbf{1}_{nx1}, \boldsymbol{d})$ where $\mathbf{1}_{nx1}$ corresponds with the dummy factor $x_{i0} = 1$, which has effect β_0. Box (1952) proves that an orthogonal design minimizes the variances of the estimated regression parameters. Moreover, the effect estimators become independent: use (21) in the covariance matrix in (10) with $\sigma_y = \sigma^2\mathbf{1}_{N\times N}$. Obviously, both orthogonal and one-factor-at-a-time designs give unbiased estimators.

How to obtain the desired orthogonal matrices? Plackett and Burman (1946) derive orthogonal designs for k up to 99, and n equal to $k + 1$ rounded upwards to a multiple of four. For example, k equal to 8, 9, 10 or 11 requires $n = 12$. This design is displayed in Table 5.

INSERT Table 5: A Plackett-Burman design with 12 combinations

Another example is k equal to 4, 5, 6 or 7, which requires $n = 8$. Writing $n = 2^{7-4}$ symbolizes that a fraction 2^4 is not simulated; that is, only 2^3 combinations are simulated. This design is displayed in Table 6, where the symbol **4** = **1.2** means $\sum_{i=1}^{n} d_{i,4} = \sum_{i=1}^{n} d_{i,1} d_{i,2}$, ..., the symbol **7** = **1.2.3** means $\sum_{i=1}^{n} d_{i,7} = \sum_{i=1}^{n} d_{i,1} d_{i,2} d_{i,3}$. These symbols are called the *generators* of the design.

INSERT Table 6: A 2^{7-4} design with generators **4** = **1.2**, ...

In general, 2^{k-p} designs with non-negative integer p smaller than k are a subclass of Plackett-Burman designs. These designs have p generators. These generators determine how effects are confounded with each other; that is, they fix the bias pattern among factor effects. For example, **4** = **1.2** implies that the estimator of the main effect of factor 4 is biased by the interaction between factors 1 and 2. Of course, if the analysts assume that only main effects are important, then this bias is unimportant. See Kleijnen (1987).

In the examples of Tables 5 and 6, *saturated* designs result when k is 11 or 7. Smaller k values -namely, 8, 9, 10 and 4, 5, 6 respectively- enable cross-validation, to check if the first-order regression metamodel is adequate (see §3.6).

So resolution-3 designs are useful when a first-order polynomial seems an adequate regression model *apriori*. This will be the case in the first stages of RSM used in optimization; see §5. Moreover these resolution-3 designs are useful as building blocks for the next type of design, namely resolution-4 designs. An example of this building block approach will be given in the FMS case study of §4.9.

4.3 Main Effects Against Two-factor Interactions: Resolution-4 Designs

According to Box and Hunter (1961), resolution-4 designs give unbiased estimators of all k main effects, even if there are interactions between pairs of factors. Box and Wilson (1951, p. 35) prove that this design property can be achieved through the *foldover* principle: to the original resolution-3 design with design matrix (say) $\boldsymbol{d}_3$ now add the 'mirror' or 'negative' image of $\boldsymbol{d}_3$, namely $-\boldsymbol{d}_3$. Obviously, the foldover principle implies doubling the number of simulated factor combinations; for example, $k = 7$ requires $n = 2 \text{ x } 8 = 16$. A subclass of resolution-4 designs are 2^{k-p} designs with the proper choice of p. An example is the 2^{8-4} design in Table 7.

INSERT Table 7: A 2^{8-4} foldover design with generators $\mathbf{4} = \mathbf{1.2.8}$, ...

Notice that this design implies that the estimator of the main effect of factor 4 is biased by the interaction among the factors 1,2, and 8, but not by any interactions between pairs of factors. (The estimators of two-factor interactions are biased by each other; for example, the estimated interaction between the factors 1 and 2 is biased by the interaction between the factors 4 and 8; see Kleijnen (1987).)

Another example of a resolution-4 design for a great many factors is given in Kleijnen, Van Ham, and Rotmans (1992, p. 416): $k = 62$ factors requires $p = 55$ generators, which are specified in that reference. Fürbringer and Roulet (1995) give a simulation application with $k = 24$ and $p = 16$.

A different subclass are the *non-orthogonal* designs derived by Webb (1968). These designs are specified only for k is 3, 5, 6 or 7 with n equal to $2k$,. Details are given in Kleijnen (1987, pp. 303-309) and in the other references.

Obviously, resolution-4 designs leave degrees of freedom over since $n > 1 + k$. Hence, these designs can give an indication of the importance of two-factor interactions. Actually, these designs give estimators of certain sums of two-factor interactions; for example, $\beta_{1,2} + \beta_{4,8} + \beta_{3,7} + \beta_{5,6}$; see Kleijnen (1987, pp. 304-305).

4.4 Individual Two-factor Interactions: Resolution-5 Designs

Resolution-5 designs give estimators of main effects and two-factor interactions that are not biased by each other; they may be biased by interactions among three or more factors. Obviously, there are $k(k - 1)/2$ such interactions: see $\beta_{h,h'}$ in (1).

One subclass is again 2^{k-p} designs with a proper choice of the p generators. An example is a 2^{8-2} design with the generators **7** = **1.2.3.4** and **8** = **1.2.5.6**. In this design no two-factor interaction estimator is biased by another two-factor interaction or main effect estimator; the estimator of the interaction between the factors 1 and 2, however, is biased by the interaction among the factors 3, 4, and 7, and among the factors 5, 6, and 8. See Kleijnen (1987).

Rechtschaffner (1967) gives *saturated* resolution-5 designs: $n = 1 + k + k(k - 1)/2$; see Kleijnen (1987, pp. 309-311). In general, resolution-5 designs require many factor combinations. Therefore, in practice, these designs are used only for small values of k (also see the next subsection).

4.5 High-order Interactions: Full Factorial Designs

If k is very small (say, $k = 3$), then all 2^k combinations can be simulated. Then all interactions -not only two-factor interactions- can be estimated. In practice, these full factorial designs are sometimes used.

Though high-order interactions can be defined mathematically, they are hard to interpret. Therefore a better metamodel may be specified, using *transformations*. For example, replace Y or x in a first-order polynomial by log Y or log x, so elasticity coefficients and decreasing marginal responses may be represented. A numerical example was presented in §3.7, concerning an M/M/s simulation model. A recent application is a simulation of Japanese production control systems known as kanban systems; see Aytug, Dogan, and Bezmez (1996). Transformations are further discussed in Cheng and Kleijnen (1997). Also see §4.7, the subsection on how to satisfy DOE assumptions.

4.6 Quadratic Effects: Central Composite Designs

Obviously, if quadratic effects are to be estimated, then at least k extra runs are needed; see $\beta_{h,\,h}$ in (1) with $h = 1, \ldots, k$. Moreover, each factor must be simulated for more than two values. Designs that are popular in both statistics and simulation are *central composite designs*. These designs combine resolution-5 designs (see §4.4) with 'one-factor-at-a-time star' designs; that is, each factor is simulated for two more values, while the other k - 1 factors are kept at their base values. In symbols: taking standardized values for all factor values. each factor is simulated for the values c and $-c$ with c not equal to one or zero, so these values are symmetrical relative to zero The selection of an appropriate value for c is surveyed in Myers et al. (1989). Besides the resolution-5 and the star design, simulate the central point $x_h = 0$ with $h = 1, 2, \ldots, k$. In other words, each factor is simulated for five values, namely the standardized values -1, +1, $-c$, $+c$, 0. These designs require relatively many runs: $n >> q$. An example is displayed in Table 8: $k = 2$ gives $n = 9$ whereas $q = 6$. These designs imply non-orthogonal columns for the $k + 1$ independent variables that correspond with the quadratic and the overall effects.

INSERT Table 8: Central composite designs for two factors

Other designs for second-order polynomials, including saturated and fractional 3^k designs, are summarized in Kleijnen (1987, pp. 314-316) and Myers et al. (1989, p. 140). A case study that uses the design of Table 8 will follow in the section on RSM (§5).

4.7 Satisfying the Classical DOE Assumptions

Classical DOE assumes white noise and a correctly specified regression model: $\sigma_y = \sigma^2 \mathbf{1}_{N \times N}$ and $\mathrm{E}(E_{i,\, j}) = 0$. In simulation, responses are *independent* if the pseudorandom number streams do not overlap and if the pseudorandom number generator is adequate (see Chapter 4, by L'Ecuyer). Practitioners, however, often use *common* pseudorandom numbers. In that case the resulting correlations should be estimated and incorporated through either EGLS (see §3.5, on VRTs) or OLS with corrected covariance matrix (see (10)). Orthogonal designs, however, no longer give independent estimators of the regression parameters: see (7) and (10). Whether other 'optimality' characteristics still hold, requires more research; also see §4.8 (on optimal DOE).

The classical DOE literature tries to realize *constant variances* through variance stabilizing transformations; see Box and Cox (1964). A major problem is that such a transformation may result in a regression metamodel that has lack of fit, or that is hard to interpret; also see Dagenais and Dufour (1996). A counter-example is the successful M/M/s study in §3.7.

In simulation, however, there is another way to realize constant variances, as simulation proceeds sequentially (apart from simulation on multi-processors). So simulate so many replications that the average response per factor combination has a constant variance (say) c_0:

$$\sigma^2_{\bar{y}_i} = \sigma^2_{y_i}/m_i = c_0 \ . \tag{22}$$

The variances in this equation can be estimated, either sequentially or in a pilot phase; Kleijnen and Van Groenendaal (1995) report good results for this heuristic. In steady-state simulations this equation needs slight adjustment.

Whether the regression metamodel is correctly specified, can be verified through *lack of fit* tests, which were discussed at length in §3.6. These tests assume that degrees of freedom are left over after estimation of the factor effects: $n > q$. Saturated designs violate this assumption. Many designs, however, are not saturated, as the preceding subsections demonstrated. If, however, a design is saturated, then one or more extra factor combinations may be simulated; also see Kleijnen and Sargent (1997).

If the regression model shows significant lack of fit, then a polynomial regression model may be augmented with higher-order terms: see the high-order interactions in §4.5. Fortunately, many classical designs can be simulated *stagewise*. For example, a resolution-3 design can be easily augmented to a resolution-4 design: see the foldover principle in §4.3. And a resolution-5 design can be easily augmented to a central composite design: see §4.6. An example will be provided by the case study on FMS in §4.9. Also see Kleijnen (1987, pp. 329-333) and Kleijnen and Sargent (1997). Lack of fit may also be reduced through transformations of the dependent or independent regression variables: see the M/M/s study in §3.7.

4.8 Optimal DOE

There are several optimality criteria in DOE: see St. John and Draper (1975); many more references are given in Kleijnen (1987, p.357, footnote 65). Here only the following three related criteria are discussed.
(i) A-optimality or trace of σ_C
An 'optimal' design may minimize the average variance of the estimated regression parameters, given the number of parameters *q* and the number of factor combinations *n*. In other words, the trace of the covariance matrix σ_C may be minimized.
(ii) D-optimality or determinant of σ_C
An error in the estimate for one parameter may affect the error for another parameter, so the off-diagonal elements of the covariance matrix σ_C may also be relevant. The determinant of σ_C gives a scalar criterion, which may be minimized.
(iii) G-optimality or maximum mean squared error (MSE)
The regression metamodel specified may be wrong. Therefore the design may minimize the maximum value of the MSE between the true regression model and the fitted model. Also see Box and Draper (1987) and Myers et al. (1989).

Kiefer and Wolfowitz (1959, p. 272) prove that under certain conditions a saturated design ($n = q$) gives 'optimal' results. They further started the investigation of *which values* of the independent variables to observe and *how many replicates* to take; also see Fedorov (1972). Recently, Cheng and Kleijnen (1997) extended Kiefer and Wolfowitz's results to 'nearly saturated' queueing simulations (traffic rates close to one), which have variance heterogeneity. Their main results are that the highest traffic rate actually simulated should be much lower than the highest traffic rate that is of interest; and the highest traffic rate simulated should be simulated for more customers than the lower traffic rates. The latter result implies that the simulation budget should not be allocated equally to the various traffic rates simulated.

Sacks, Welch, Mitchell, and Wynn (1989) assume that fitting errors are *positively correlated*: the closer two factor combinations are in *k*-dimensional space, the more the fitting errors are correlated. This assumption is realistic if the response function is smooth. Technically speaking, they assume a covariance stationary process, as an alternative to white noise. Also see Welch, Buck, Sacks, Wynn et al. (1992).

Optimal DOE gives designs that do not have the standard geometric patterns that classical designs have. These designs can not be looked up in a table; they are generated by means of a computer; see Nachtsheim (1987, pp. 146, 151). Additional research is needed to facilitate more frequent application of optimal designs in simulation practice.

An overview of optimal DOE theory, including many references to the older literature is Myers et al. (1989, pp. 140-141). More recent literature is given in Atkinson and Cook (1995) and Ermakov and Melas (1995).

4.9 Case Study: FMS

A machine-mix problem for a particular FMS is studied in Kleijnen and Standridge (1988), and summarized in Kleijnen and Van Groenendaal (1992, pp. 162-164). The analysts wish to determine the number of machines, per type of machine such that a given production volume is realized. There are four machine types: w_1 through w_4. Only type #4 is a flexible machine; the other types are dedicated to a particular operation. A more complicated simulation model would allow for random service and arrival times, and random machine breakdowns. Yet the deterministic simulation model actually used, demonstrates many main issues to be solved in DOE. Issues typical for random simulation are runlength determination and estimation of the response covariance matrix $\mathbf{\sigma}_y$. These tactical issues are not addressed in this case study. (Yet this study was initiated at a major supplier of discrete-event simulation software, namely Pritsker & Associates.)

Because these inputs must be integers, there are only 24 feasible combinations in the experimental domain. The actual values are given in Kleijnen and Standridge (1988), but they are not relevant here. Initially eight intuitively selected combinations are simulated; see Table 9.

INSERT Table 9: Intuitive design for the four factors in the FMS simulation in Kleijnen and Standridge (1988)

Then DOE is applied, initially assuming a first-order polynomial regression metamodel for these four factors. A possible resolution-3 design is a 2^{4-1} design with **3** = **1**.**2**.**4**; see §4.2. This design is not saturated, so lack of fit can be examined. Because the simulation model is deterministic, OLS is used to estimate the factor effects. Table 10 shows that the orthogonal design indeed gives more accurate estimates.

INSERT Table 10: Estimated variances of estimated effects in first-order polynomial regression metamodel for FMS simulation

The remainder of this subsection concentrates on the results of the more accurate 2^{4-1} design. Cross-validation gives Table 11, which displays the estimated factor effects that remain significantly different from zero when $\alpha = 0.30$.

INSERT Table 11: Cross-validation and estimated factor effects significantly different from zero when $\alpha = 0.30$; first-order polynomial metamodel for four factors in FMS simulation

Besides the stability of individual effects in cross-validation, the relative prediction errors $\hat{y}_i/y_i$ are of interest. These errors turn out to range between -38% and +33%. Altogether, these cross-validation results lead to rejection of this first-order polynomial regression metamodel.

Table 11 does suggest that the factors #1 and #3 are *not* important. Therefore a metamodel for the remaining factors # 2 and #4 is formulated, but now including their interaction. Using the old I/O data, which resulted from the 2^{4-1} design, gives Table 12.

INSERT Table 12: Cross-validation and stability of $\hat{\beta}_2$, $\hat{\beta}_4$, $\hat{\beta}_{2,4}$ and $\hat{\beta}_0$ in metamodel with interaction for FMS

This table shows that in this new metamodel all estimated effects remain significant in cross-validation! Further, the relative prediction errors $\hat{y}_i/y_i$ become much smaller: they now vary between -16% and +14%. So the conclusions of this case study are:

(i) The machines of types 2 and 4 are bottlenecks, not the types 1 and 3: $\hat{\beta}_1$ and $\hat{\beta}_3$ are not significant in Table 11.
(ii) There is a trade-off between the machine of types 2 and 4, as there is a significant negative interaction: see $\hat{\beta}_{2,4}$ in Table 12.
(iii) The regression metamodel helps to understand how the FMS works.

5. Optimization of Simulated Systems: RSM

Whereas the previous designs were meant to gain understanding of the simulation model through what-if analysis, the goal of simulation experimentation may also be to optimize the simulated system. There are many mathematical techniques for optimizing the decision variables of nonlinear implicit functions; simulation models are indeed examples of such functions. These function may include stochastic noise, as is the case in random simulation. Examples of such optimization techniques are sequential simplex search (see Nachtsheim 1987), genetic algorithms, simulated annealing, and tabu search (see Nash 1995). Software is given in Chapter 25 (Vendor survey; §25.3), for example, ProModel's SimRunner and MicroSaint's OptQuest. There is virtually no software for optimization in the presence of multiple responses; see Khuri (1996, p. 240, 242). Software for the optimization of system dynamics includes DYSMOD's pattern search; see Dangerfield and Roberts (1996). System dynamics is a special kind of simulation, which may inspire developers of discrete-event simulation software. This paper, however, concentrates on RSM.

Note that some authors outside the discrete-event simulation area speak of RSM, but mean what this chapter calls the what-if regression-metamodeling approach, not the sequential optimization approach; see, for example, Olivi (1980). Further, RSM assumes that the decision variables are quantitative. Systems that differ qualitatively, are discussed in Chapter 8 (by Goldsman and Nelson).

Classical RSM assumes a *single* type of response. However, the two case studies in §5.2 and §5.3 have two response types. In these case studies one response is to be maximized, whereas the other response must meet a side-condition. In the case study in Keijzer, Kleijnen, Mullenders, and Van Reeken (1981), however, three response types are considered, namely waiting time for each of three priority classes; a single overall criterion function is formulated by quantifying tradeoffs among the waiting times per job class.

The next subsection (§5.1) first gives general characteristics of RSM; next, it will give some details on RSM. The two last subsections (§5.2, §5.3) give case studies.

5.1 Response Surface Methodology

RSM has the following four general characteristics.
(i) RSM relies on first and second order polynomial regression metamodels or response surfaces; the responses are assumed to have white noise (see §3).
(ii) RSM uses classical designs (see §4).
(iii) RSM adds to regression models and DOE the mathematical (not statistical) technique of steepest ascent; that is, the estimated gradient determines in which direction the decision variables are changed.

(iv) RSM uses the mathematical technique of canonical analysis to analyze the shape of the optimal region: does that region have a unique maximum, a saddle point or a ridge (stationary points)?

Now consider some details. Suppose the goal of the simulation project is to maximize the response; minimization is strictly analogous.

RSM begins by selecting a *starting point*. Because RSM is a heuristic, success is not guaranteed. Therefore several starting points may be tried later on, if time permits.

RSM explores the *neighborhood* of the starting point. The response surface is approximated locally by a first-order polynomial in the decision variables, as the Taylor series expansion suggests. This gives the regression metamodel in (1), but with all cross-products eliminated. Hence k main effects β_h with $h = 1, \ldots, k$ are to be estimated,. For that purpose a resolution-3 design should be used; this gives $n \approx k + 1$; see §4.2.

If (say) the estimated effects are such that $\hat{\beta}_1 >> \hat{\beta}_2 > 0$, then obviously the increase of w_1 (decision variable 1) should be larger than that of w_2; the symbol w refers to the original, non-standardized variables; see (2). The *steepest ascent path* means $\Delta w_1/\Delta w_2 = \hat{\beta}_1/\hat{\beta}_2$; in other words, steepest ascent uses the local gradient.

Unfortunately, the steepest ascent technique does not quantify the *step size* along this path. Therefore the analysts may try a specific value for the step size. If that value yields a lower response, then the step size should be reduced. Otherwise, one more step is taken. Note that there are more sophisticated mathematical procedures for selecting step sizes; also see Safizadeh and Signorile (1994).

Notice that special designs have been developed to estimate the slope accurately; that is, these designs are alternatives to the classical resolution-3 designs mentioned above; see Myers et al. (1989, pp. 142-143).

Ultimately, the simulation response must decrease, since the first-order polynomial is only a local approximation to the real I/O transformation of the simulation model. In that case the procedure is repeated. So around the best point so far, the next $n \approx k + 1$ combinations of w_1 through w_k are simulated. Note that the same resolution-3 design may be used; only the locations and spreads of the original variables are adjusted: see (2). Next the factor effects in the new local first-order polynomial are estimated. And so RSM proceeds.

A first-order polynomial or hyperplane, however, cannot adequately represent a hill top. So in the neighborhood of the optimum, a first-order polynomial may show serious lack of fit. To detect such specification error, the analysts might use cross-validation. In RSM, however, simple diagnostic measures are more popular: $R^2 << 1$, $\hat{var}(\hat{\beta}_h) >> \hat{\beta}_h$. Also see §3.6 (on lack of fit).

So next a second-order polynomial is fitted: see §3 and §4. Finally, the *optimal* values of the decision variables w_h are found by straightforward differentiation of this polynomial. A more sophisticated evaluation uses canonical analysis; see Myers et al. (1989, p. 146).

Software for RSM is available. Much software referenced in the section on DOE (§4), also handles RSM. Nachtsheim (1987) discusses special RSM software, namely SIMPLEX-V and ULTRAMAX. This software provides designs such as central composite designs (see §4.6), and contour plots of the fitted surface, even for multiple responses; also see Myers et al. (1989, pp. 145-146, 148).

RSM is further discussed in Fu (1994), Ho, Shi, Dai, and Gong (1992), Khuri and Cornell (1996), and Sheriff and Boice (1994). Applications of RSM to simulated systems can be found in Hood and Welch (1993), Kleijnen (1987), and Kleijnen and Van Groenendaal (1992). Numerous applications outside the simulation field are surveyed in Myers et al. (1989, pp. 147-151). The next subsection summarizes a case study illustrating how RSM climbs a hill; a following subsection (§5.3) gives a different case study illustrating how a hill top can be further explored when there are two response types.

5.2 Case Study: Production Planning DSS

Kleijnen (1993) studies a DSS for production planning in a specific Dutch steel tube factory that has already been mentioned in §1. Both the DSS and the factory are simulated; the DSS is to be optimized. This DSS has fourteen decision variables; for example, w_1 denotes 'penalty for producing class-2 products on the next best machine'. There are two response variables, namely the total number of productive hours, and the 90% quantile of lead time. Simulation of one input combination takes six hours on the computer available at that time. Consequently, searching for the optimal combination must be performed with care.

The original team of operations researchers planned to fit a local first-order polynomial with 14 inputs. They designed to simulate the base case first. Next they planned to change one factor at a time, once making each factor 20% higher than its base value, and once 20% lower. Obviously, this design implies n, number of simulated factor combinations, equal to $1 + 2 \times 14 = 29$. Kleijnen (1993), however, uses a 2^{14-10} design, so $n = 16 > q = 15$. The specific $n = 16$ combinations are specified by writing all 2^4 combinations of the first four factors, and then using the generators **5** = **1.2**, **6** = **1.3**, **7** = **1.4**, **8** = **2.3**, **9** = **2.4**, **10** = **3.4**, **11** = **1.2.3**, **12** = **1..2.4**, **13** = **1.3.4**, and **14** = **2.3.4**.

The resulting OLS estimates are *not* tested for significance, as a small parameter value may become a big value in a next local area. Table 13 gives the local effects on productive hours and on lead time respectively.

INSERT Table 13: Estimated local effects on productive hours and lead time in simulated production planning DSS

This table demonstrates that some decision variables have a favorable local effect on both response types. For example, raising w_1 by one unit increases production by $\hat{\beta}_1 = 0.52$; at the same time this raise improves lead time by $\hat{\gamma}_1 = -0.054$. And lowering w_4 by one unit changes production by $\hat{\beta}_4 = -18.07$, while it decreases lead time by $\hat{\gamma}_4 = 150.583$. Because the decision variables have different scales and ranges, this table gives not only the unit effects but also the unit effects multiplied by the base values (say) $w_{0,1}$ through $w_{0,14}$.

The step size in RSM must be determined heuristically. In this case the step size is selected such that at least one of the decision variables is doubled. Table 14 shows that w_{12} changes from 0.33 to 0.5936; the other thirteen variables do not change much. These changes result from a step size of 0.0005; that is, the base values $w_{0,h}$ become $w_{0,h} + 0.0005\hat{\beta}_h$. Further results are given in Kleijnen (1993).

INSERT Table 14: Values of decision variables in base run and after a step of 0.005 on steepest ascent

5.3 Case Study: Coal Transport

This subsection returns to the coal transport model already presented in §3.8, but now the focus is on optimization instead of sensitivity analysis. Wolstenholme (1990, pp. 125-127) restricts the optimization to the best control rule, namely policy III. Obviously efficiency, now denoted by $y^{(1)}$, can not exceed 100%. Therefore the goal is to minimize total costs, denoted by (say) $y^{(2)}$, under the condition that the efficiency remains at its maximum of 100%. Wolstenholme assumes that one input is fixed: the 'maximum discharge rate' is fixed at an average of 1000 tons/hour. He wishes to optimize the two remaining inputs, w_1 or 'total belt capacity' and w_2 or 'capacity per bunker'. (The two symbols w_2 and w_3 have different meanings in §3.8 and this subsection.) The cost parameters are £1000/ton/hour for w_1 and £2000/ton for w_2. The total costs are a linear function of the decision variables: $y^{(2)} = 1000w_1 + 2000w_2$. Efficiency, however, is a complicated non-linear function specified by the system dynamics model; see §3.8.

Kleijnen (1995) develops the following heuristic, inspired by RSM. Classic RSM, however, maximizes a single criterion, ignoring restrictions such as the one on efficiency, namely $y^{(1)} = 1$.

Step 1. Find an initial combination $\mathbf{w} = (w_1, w_2)'$ that yields a simulated efficiency of 100%. Such a combination is already available: see the element in the last row and column of Table 3, discussed in §3.8.

Step 2. Reduce each input by (say) 10%. Simulate the system dynamics model with this input. Obtain the corresponding output.

Step 3. If the output of step 2 is still 100%, then return to step 2; else proceed to the next step.

Step 4. Find the most recent input combination that satisfies the efficiency restriction $y^{(1)} = 1$; see steps 1 and 2. Reduce the step size to (say) 5%. Simulate the model with this new input combination, and obtain the corresponding output.

Step 5. Further explore the most recent local area that includes a combination with $y = 1$. In other words, simulate the model for the four input combinations that are specified by the 2^2 design. In Figure 2 these four combinations form the rectangle with lower left corner (2693, 923) and upper right corner (2835, 972).

Since this heuristic does not result in further progress, the optimum seems to be close. Now a second-order polynomial is specified as an approximation to the production function $y^{(1)}(w_1, w_2)$. To estimate the six parameters in this polynomial, the 2^2 design is expanded to the central composite design of Table 8, discussed in §4.6. Figure 2 shows the central point (2764, 948). The standardized axial value c is set at 0.75; if $c > 1$ then $(0, c)$ and $(c, 0)$ give too high costs, and $(0, -c)$ and $(-c, 0)$ give too low efficiencies. Table 15 shows the standardized and original values of the two decision variables, and the resulting costs and efficiencies.

INSERT Table 15: Input combinations in the central composite design with corresponding costs and efficiencies for the coal transport system

From the I/O data in that table the second-order polynomial can be estimated. Because the simulation model is deterministic, OLS is used. This gives

$$\hat{y}^{(1)} = 601.138514 - 0.1239693w_1 - 0.7161188w_2 + \\ - 0.0000334w_1w_2 + 0.0000291w_1^2 + 0.0004282w_2^2$$

Though the quadratic and interaction coefficients look small, they should not be ignored, since they are multiplied by w_1^2, w_2^2, and w_1w_2, which are large.

Step 6. Combine this second-order polynomial approximation with the cost restriction $y^{(2)} = 1000w_1 + 2000w_2$; that is, replace $\hat{y}^{(1)}$ in the left-hand side of this polynomial by the value one. Mathematically, this results in an ellipsoid: Figure 2 shows only a small part of that ellipsoid. Economically, the ellipsoid gives the *efficiency frontier*: outside the ellipsoid, inputs are wasted; inside that ellipsoid the efficiency is too low.

Minimizing the total cost $y^{(2)} = 1000w_1 + 2000w_2$ under the efficiency restriction $y^{(1)} = 1$ can be done through a Lagrangean multiplier. This yields the estimated optimal input combination $(\hat{w}_1^*, \hat{w}_2^*) = (2841.35; 968.17)$. Graphically, this is the point in which the efficiency frontier is touched by an *iso-cost line* with the angle -1000/2000 = -1/2.

INSERT Figure 2. Central composite design, estimated efficiency frontier, and optimal iso-cost line; * means $y^{(1)} = 1$; O means $y^{(1)} < 1$

This optimum is based on an approximation: the estimated second-order polynomial has a multiple correlation coefficient R^2 of only 0.80. Therefore this solution is checked: simulate the system dynamics model with the estimated optimal input combination. This simulation indeed gives 100% efficiency. However, its cost is £ 4.78 mln, which exceeds the lowest cost in Table 15 that corresponds with a combination that also gives 100% efficiency. In that table the combination (2835.00; 923.40) gives a cost of £ 4.68 mln, which is 2% lower than the estimated optimal cost. Compared to Wolstenholme (1990, pp. 125-127), the final solution saves £ 0.72 mln or 15%, which is a substantial cost reduction.

6. Validation and Verification (V & V)

6.1 Overview

This chapter limits its discussion of V & V to the role of regression analysis and DOE; V & V are further discussed in Chapter 10 (by Balci); also see Kleijnen (1995a). Obviously, V & V is one of the first questions that must be answered in a simulation study. For didactic reasons, however, V & V is discussed at the end of this chapter.

True validation requires that *data on the real system* be available. In practice, the amount of such data varies greatly: data on failures of nuclear installations are rare, whereas electronically captured data on computer performance and on supermarket sales are abundant.

If data are available, then many statistical techniques can be applied. Assume that the simulation is fed with real-life input data: this is known as *trace driven simulation*. Then simulated and real responses (say) Y and X respectively might be compared through the Student statistic for paired observations, assuming (say) m normally and independently distributed (n.i.d.) observations on X and Y: see (8) with Y replaced by $Y - X$ and $\nu = m - 1$.

A better test, however, uses *regression analysis*, as follows. Regress $Y - X$ on $Y + X$; that is, in the regression model (1) replace Y by $Y - X$ and x by $Y + X$. Use a first-order polynomial; that is, delete cross-products in (1). Next test the null-hypothesis H_0: $\beta_0 = 0$ and $\beta_1 = 0$. This hypothesis implies equal means and equal variances of X and Y, as is easily derived assuming bivariate normality for (X, Y). This hypothesis is tested using the familiar F statistic. Whenever the simulation responses are non-normal, a normalizing transformation should be applied. Kleijnen, Bettonvil, and Van Groenendaal (1997) give details, including numerical examples for single-server queueing simulations. This reference also demonstrates that a valid simulation model is rejected too often when simply regressing Y on X, and testing for zero intercept and unit slope.

If *no data* are available, then DOE can be used, in the following way. The - simulationists and their clients do have *qualitative* knowledge about certain parts of the simulated and the corresponding real system; that is, they do know in which direction certain factors affect the response of the corresponding module in the simulation model; also see the discussion on known signs in sequential bifurcation in §2.2. If the regression metamodel discussed in §3 gives an estimated factor effect with the wrong sign, this is a strong indication of a wrong simulation model or a wrong computer program!

To obtain a valid simulation model, some inputs may need to be restricted to a certain domain of factor combinations. This domain corresponds with the *experimental frame* in Zeigler (1976), a seminal book on modeling and simulation.

The regression metamodel shows which factors are most important; that is, which factors have highly significant regression estimates in the metamodel. If possible, information on these factors should be collected, for validation purposes.

The importance of sensitivity analysis in V & V is also emphasized by Fossett, Harrison, Weintrob, and Gass (1991, p. 719). They investigate three military case studies, but do not present any details.

One application of DOE and regression analysis in V & V is the ecological study in Kleijnen, Van Ham, and Rotmans (1992, p. 415), which concerns the same greenhouse problem examined in Bettonvil and Kleijnen (1997). The regression metamodel helped to detect a serious error in the simulation model: one of the original modules should be split into two modules. This application further shows that some factors are more important than the ecological experts originally expected. This 'surprise' gives more insight into the simulation model. Another application will be given in the next subsection.

6.2 Case Study: Mine Hunting at Sea

Kleijnen (1995b) considers a simulation model for the study of the search for explosive mines on the sea bottom by means of sonar. This model was developed for the Dutch navy, by TNO-FEL, which stands for Applied Scientific Research-Physics and Electronics Laboratory; this is a major military research institute in the Netherlands. The model is called HUNTOP, which stands for mine HUNTing Operation. Other countries have similar simulation models for naval mine hunting; the corresponding literature is classified.

In this case study, V & V proceeds in two stages: in stage #1 individual modules are validated; in stage #2 the whole simulation model is treated as one black box, and is validated. The latter stage, however, is not discussed here, but in Kleijnen (1995b).

Some modules within the model give *intermediate* output that is hard to observe in practice, and hence hard to validate. Sensitivity analysis is applied to such modules, in order to check if certain factor effects have signs or directions that agree with experts' prior qualitative knowledge. For example, deeper water gives a wider sonar window; see β_2 in the sonar window module below.

Because of time constraints, only the following two modules are examined in the HUNTOP case study.

(i) Sonar window module

The sonar rays hit the bottom under the so-called grazing angle. This angle is determined deterministically by three factors, namely w_1 or Sound Velocity Profile (SVP) that maps sound velocity as a function of depth, w_2 or average water depth, and w_3 or tilt angle. SVP is treated as a qualitative factor.

The sonar window module has as response variables $y^{(1)}$, the minimum distance of the area on the sea bottom that is insonified by the sonar beam, and $y^{(2)}$, the maximum distance of that same area. Consider a second-degree polynomial in the two quantitative factors w_2 and w_3, namely one such polynomial for each SVP type. Note that a first-degree polynomial misses interactions and has constant marginal effects; a third-order polynomial is more difficult to interpret and needs many more simulation runs. So a second-order polynomial seems a good compromise

To estimate the $q = 6$ regression parameters of this polynomial, use the classical central composite design with $n = 9$ input combinations, already displayed in Table 8. The fitted polynomial turns out to give an acceptable approximation: the multiple correlation coefficient R^2 ranges between 0.96 and 0.98, for the four SVPs simulated.

Expert knowledge suggests that certain factor effects have specific signs, namely $\beta_2 > 0$, $\beta_3 < 0$, and $\beta_{2,3} < 0$. Fortunately, the corresponding estimates turn out to have the correct signs. So this module has the correct I/O transformation, and the validity of this module need not be questioned. The quadratic effects are not significantly different from zero. So on hindsight, simulation runs could have been saved, since a resolution-5 design instead of a central composite design would have sufficed.

For $y^{(2)}$, maximum distance, similar results hold. The exception is one SVP that results in an R^2 of only 0.68 and a non-significant $\hat{\beta}_2$.

(ii) Visibility module

An object is visible if it is within the sonar window, and it is not concealed by the bottom profile. HUNTOP represents the bottom profile through a simple geometric pattern, namely hills of fixed heights with constant upward slopes and constant downward slopes. A fixed profile is used within a single simulation run. Intuitively, the orientation of the hills relative to the ship's course and to the direction of the sonar beam is important: does the sonar look down a valley or is its view blocked by a hill? The response variable of this module is the time that the object is visible, expressed as a percentage of the time it would have been visible were the bottom flat; obviously, a flat bottom does not conceal an object. Six inputs are varied, namely water depth, tilt angle, hill height, upward hill slope, downward hill slope, and object's position on the hill slope (top, bottom, or in between).

A quadratic polynomial regression metamodel is also used for this module. To estimate the 28 regression parameters, a central composite design is used. This design has 77 input combinations. R^2 turns out to be 0.86. The upward hill slope has no significant effects at all: no main effect, no interactions with the other factors, no quadratic effect. These results agree with the experts' qualitative knowledge. So the validity of this module is not questioned either.

7. Conclusions

In the Introduction (§1) the following questions were raised:
1. *What if*: what happens if the analysts change parameters, input variables or modules of a simulation model? This question is closely related to sensitivity analysis and optimization.
2. *Validation*: is the simulation model an adequate representation of the corresponding system in the real world?
These questions were answered as follows, in the remainder of this chapter.

In the initial phase of a simulation it is often necessary to perform *screening*: which factors among the multitude of potential factors are really important? The goal of screening is to reduce the number of really important factors to be further explored in the next phase. The technique of sequential bifurcation is simple and efficient. This technique also seems to be effective.

Once the important factors are identified, further study requires fewer assumptions, namely no known signs are assumed. This study may use *regression analysis*. It generalizes the results of the simulation experiment, since it characterizes the I/O transformation of the simulation model.

Design Of Experiments (DOE) gives better estimators of the main effects, interactions, and quadratic effects in this regression metamodel. So DOE improves the effectiveness of simulation experimentation. DOE requires relatively few simulation runs, which means improved efficiency.

Once the factor effects are quantified through the corresponding regression estimates, they can be used in *V & V*, especially if there are no data on the I/O of the simulation model or its modules, and in *optimization* through RSM, which augments regression analysis and DOE with the steepest-ascent hill-climbing technique.

Applying the statistical techniques of this chapter in simulation studies is meant to make these studies give more general results, in less time These techniques have already been applied many times in practical simulation studies, in many domains, as the various case studies demonstrated. This chapter should stimulate more analysts to apply these techniques..

In the mean time, research on statistical techniques adapted to simulation, should continue. This chapter did mention several items that require further research.

Appendix 1: Confidence intervals for individual responses

The following formulas are taken from Kleijnen (1996).
(i) Mean of terminating simulation

A 1 - α one-sided confidence interval for E(Y) is

$$P[E(Y) > \bar{Y} - t_{\alpha;\, m-1} S_y/\sqrt{m}] = 1 - \alpha \qquad (24)$$

where $t_{\alpha, m-1}$ denotes the 1 - α quantile of the Student statistic T_{m-1}. This interval assumes that the simulation response Y is normally, independently distributed (n.i.d.). The Student statistic is known to be not very sensitive to non-normality; the average Y is asymptotically normally distributed (Central Limit Theorem).

Johnson (1978) modifies the Student statistic in case Y has a distribution with *asymmetry coefficient* μ_3 (also see Kleijnen 1987, pp. 22-23):

$$\tilde{t}_{m-1} = [(\bar{Y} - E(Y)) + \frac{S_3}{6S^2 m} + \frac{S_3}{3(S^2)^2}(\bar{Y} - E(Y))^2(\frac{S^2}{m})^{-1/2} \qquad (25)$$

where S_3 denotes the asymmetry estimator:

$$S_3 = \frac{m\sum_{j=1}^{m}(Y_{i,j} - \bar{Y}_i)^3}{(m-1)(m-2)} \qquad (26)$$

Kleijnen, Kloppenburg and Meeuwsen (1986) discuss this statistic in detail.

(ii) Mean of steady-state simulation

For steady-state simulations the analysts may apply *renewal analysis* (also see Chapter 7, by Alexopoulos and Seila). Denote the length of the renewal cycle by L, and the *total* cycle response (for example, total waiting time over the whole cycle) by W. Then the steady-state mean response is

$$E(Y) = E(W)/E(L). \qquad (27)$$

This analysis uses *ratio estimators*; see $\bar{W}/\bar{L}$ in the next equation. Crane and Lemoine (1977, pp. 39-46) derive the following asymptotic 1 - α confidence interval for the mean:

$$P[E(Y) > \bar{W}/\bar{L} - z_\alpha S/(\sqrt{m}/\bar{L})] = 1 - \alpha \qquad (28)$$

where z_α denotes the (1 - α) quantile of the standard normal variate N(0, 1), m is now the number of cycles (in terminating simulations m was the number of replications), and S^2 is a short-hand notation:

$$S^2 = [S_w^2 + (\bar{W}/\bar{L})^2 S_L^2 - 2(\bar{W}/\bar{L})S_{W,L}]^{1/2} \qquad (29)$$

where the (co)variances are estimated analogous to (14).

In a Markov system, *any* state can be selected as the renewal state. A practical problem, however, is that it may take a long time before the selected renewal state occurs again; for example, if the traffic rate is high, then it takes a long time fore all servers to be idle again. Also, if there are very many states (as may be the case in network systems), then it may take a long time before a specific state occurs again. In those cases *nearly renewal* states may be used; for example, define 'many servers busy' as the set of (say) two states, 'all servers busy' or 'all minus one servers busy'. This approximate renewal state implies that the cycles are not exactly i.i.d. However, for practical purposes they may be nearly i.i.d., which may be tested through the Von Neumann statistic for successive differences:

$$\sum_{j=2}^{m} (W_j - W_{j-1})^2/[(m - 1)S_w^2]. \tag{30}$$

This statistic is approximately normally distributed with mean 2 and variance $4(m - 2)/(m^2$- - 1), provided the W_j are n.i.d.. Since W is a sum, normality may apply. However, when testing the mutual independence of the cycle lengths L_j and L_{j-1}, a complication is that L probably has an asymmetric distribution. Then the rank version of the Von Neumann statistic may be applied; see Bartels (1982). To ensure that the Von Neumann test has a reasonable chance of detecting dependence, at least 100 cycles are needed: for $m < 100$ the test has low power; see Kleijnen (1987, p. 68).

The proposed approximate renewal analysis of simulation requires more research. Also see Gunther (1975), Gunther and Wolff (1980), and Pinedo and Wolff (1982). A different method for 'accelerating' the renewal process is proposed in Andradottir, Calvin, and Glynn (1995).

(iii) Proportions

A proportion (say) p_a is estimated by comparing the simulation outputs y_j ($j = 1, \ldots, m$) with the prespecified constant a, which leads to the binomially distributed variable (say) $B = \sum_{j=1}^{m} D_j$ with $D_j = 0$ if $Y_j < a$; else $D_j = 1$. This binomial variable has variance p_a (1 - p_a)m. Obviously, B/m is an estimator of p_a with variance $p_a(1 - p_a)/m$.

(iv) Quantiles

The following notation ignores the fact that pm, l, and u are not necessarily integers; actually these three real variables must be replaced by their integer parts. The p^{th} quantile z_p (with $0 < p < 1$) may be estimated by the order statistic $y_{(pm)}$. The (1 - α) confidence interval is

$$P(z_{(l)} < z_p < z_{(u)}) = 1 - \alpha \tag{31}$$

where the lower limit is the l^{th} order statistic with

$$l = pm - z_{\alpha/2}\sqrt{p(1 - p)m}\ , \tag{32}$$

and the upper limit is the u^{th} order statistic that follows from the preceding equation replacing the minus sign in front of $z_{\alpha/2}$ by a plus sign. Proportions and quantiles in terminating and steady-state simulations are further discussed in Kleijnen (1987, pp. 36-40) and Kleijnen and Van Groenendaal (1992, pp. 195-197).

(v) Multiple responses

In the presence of multiple responses, each individual (1 - α) confidence interval has a coverage probability of 1 - α, but the simultaneous or joint coverage probability is lower. If the intervals were independent and there were (say) two responses, then this probability would be $(1 - \alpha)^2$. Bonferroni's inequality implies that if the individual confidence intervals use α, then the probability that both intervals hold simultaneously is at least 1 - 2α. In general, this conservative procedure implies that the simultaneous type I error rate (say) α_E is divided by the number of confidence intervals, in order to guarantee a joint probability of α_E.

Appendix 2: Variance reduction techniques

VRTs are discussed in detail in Fishman (1989), Kleijnen (1974/75, pp. 105-285), Kleijnen and Van Groenendaal (1992, pp.197-201), Ermakov and Melas (1995), Tew and Wilson (1994), and in the references mentioned below.

(i) Common Pseudorandom Numbers

In the what-if approach there is more interest in the differences than in the absolute magnitudes of the simulation outputs. Intuitively it seems appropriate to examine simulated systems under equal conditions, that is, in the same environments. This implies the use of the same stream of pseudorandom numbers for two different factor combinations. Then the two simulation responses (say) Y_1 and ${}_2Y$ become statistically dependent. A general relationship is

$$\sigma^2_{y_1 - y_2} = \sigma^2_{y_1} + \sigma^2_{y_2} - 2\rho_{y_1, y_2}\sigma_{y_1}\sigma_{y_2} \ . \tag{33}$$

So if the use of the same pseudorandom numbers does result in positive correlation, then the variance of the difference decreases.

However, in complicated models it may be difficult to realize a strong positive correlation. Therefore separate sequences of pseudorandom numbers are used per 'process'; for example, in a queueing network a separate seed is used per server. One seed may be sampled through the computer's internal clock. However, sampling the other seeds in this way may cause overlap among the various streams, which makes times at different servers statistically dependent. For certain generators, there are tables with seeds 100,000 apart. For other generators such seeds may be generated in a separate computer run. Also see Kleijnen and Van Groenendaal (1992, pp. 29-30).

The advantage of a smaller variance comes at a price: the analysis of the simulation results becomes more complicated, since the responses are not independent anymore. Now the analysts should use either GLS or OLS with adjusted standard errors for the estimated regression parameters; this implies that the analysts should estimate the covariances between simulation responses. In practice this complication is often overlooked.

(ii) Antithetic pseudorandom numbers

The intuitive idea behind antithetic pseudorandom numbers (briefly 'antithetics') is as follows. When replication #1 samples many long service times, then the average waiting time y is higher than expected. So it is nice if replication #2 compensates this overshoot. Statistically this compensation means negative correlation between the responses of replications #1 and #2. The variance of their average (say) Y, taking into account that both replications have the same variance, follows from (33):

$$\sigma^2_{\bar{y}} = \sigma^2_y[1 + \rho_{y_1, y_2}]/2 \ . \tag{34}$$

So the variance of the average $\bar{Y}$ decreases as the correlation ρ_{y_1, y_2} becomes more negative.

To realize a strong negative correlation, use the pseudorandom numbers r for replication #1, and the 'antithetics' 1 - r for replication #2. Actually, the computer does not need to calculate the complements 1 - r, if it uses a multiplicative congruential generator. Then it suffices to replace the seed r_0 by its complement e - r_0 , where e stands for the generator's modulo; that is, $r_t = fr_{t-1}$ *mod e* where f denotes the generator's multiplier. Also see Kleijnen (1974, pp. 254-256).

(iii) Control variates or regression sampling
Consider the following linear correction:

$$Y_c = Y + \gamma_1[E(X_1) - X_1] \tag{35}$$

where Y_c is called the linear control variate estimator. Obviously, this new estimator remains unbiased. It is easy to derive that this control variate estimator has minimum variance, if the correction factor γ_1 equals

$$\gamma_1^* = \rho_{y,\, x_1}\sigma_y/\sigma_{x_1} \ . \tag{36}$$

In practice, however, the correlation $\rho_{y,\, x_1}$ is unknown, so it is estimated. Actually, replacing the three factors in the right-hand side of the preceding equation by their classic estimators results in the OLS estimator (say) $C_{1,\, OLS}$ of the regression parameter γ in the regression model

$$Y_{x_1} = \gamma_0 + \gamma_1 x_1 + U \tag{37}$$

where U denotes the fitting error of this regression model, analogous to E in (1). Obviously, these two regression parameters (γ_0, γ_1) are estimated from the m replications that give m i.i.d. pairs ($X_{1,\, h}$, Y_h) with $h = 1, \ldots, m$.

The OLS estimator of γ_1^* defined in (36) gives a new control variate estimator. Let $\bar{Y}$ denote the average over m replications of the responses Y_j, X_1 the average over m replications of X_1 (average service time per run), and $C_{1,\, OLS}$ the OLS estimator of γ in (37) or γ_1^* in (36) based on the m pairs (Y, X_1). Then the new control variate estimator is given in (15). This estimator is easy to interpret, noticing that the estimated regression line goes through the point of gravity, (X_1, Y).

(iv) Importance sampling
The preceding VRTs relied on the correlation between (i) the responses of systems with comparable simulated environments realized through common seeds, (ii) the responses of antithetic runs, or (iii) the output and inputs or control variates. In other words, the simulation model itself was not affected; the computer program might be slightly adapted to increase the positive and negative correlations (seeds were changed or inputs were monitored). Importance sampling, however, drastically changes the sampling process of the simulation model. This technique is more sophisticated, but it is necessary when simulating *rare events*; for example, buffer overflow may occurs with a probability of (say) one in a million replicated months so a million replicated months must be simulated, to expect to see a single breakdown of the system!

The basic idea of importance sampling can be explained simply in the case of static, non-dynamic simulation, also known as Monte Carlo sampling. Consider the following example integral

$$\xi = \int_{\nu}^{\infty} \frac{1}{x}\lambda e^{-\lambda x} dx \ \ with\ \lambda > 0,\ \nu > 0 \ . \tag{38}$$

This ξ can be estimated through Crude Monte Carlo as follows.
(i) Sample x from the negative exponential distribution with parameter λ; that is, $x \sim$ Ne(λ).

(ii) Substitute the sampled value x into the 'response' $Y = g(X)$ with

$$g(x) = \frac{1}{x} \; if\; x > v; \qquad 0 \; otherwise. \tag{39}$$

Obviously, g(X) is an unbiased estimator of ξ. Notice that the event '$g(x) > 0$' becomes a rare event as $v \uparrow \infty$.

Importance sampling does not sample x from the original distribution $f(x)$ (in the example, Ne(λ)), but from a different distribution (say) $h(x)$. The resulting x is substituted into the response function $g(x)$. However, $g(x)$ is corrected by the *likelihood ratio* $f(x)/h(x)$. This gives the corrected response

$$g^*(x) = g(x)\frac{f(x)}{h(x)}. \tag{40}$$

This estimator is an unbiased estimator of ξ:

$$E[g^*(X)] = \int_v^\infty g(x)\frac{f(x)}{h(x)}h(x)dx = \int_v^\infty g(x)f(x)dx = \xi. \tag{41}$$

It is quite easy to derive the optimal form of $h(x)$, which results in minimum variance.

For *dynamic systems* (such as queueing systems) a sequence of inputs must be sampled; for example, successive service times $X_{1,\,t}$ with $t = 1, 2, \ldots$. If these inputs are assumed to be i.i.d. and Ne(λ)), then their joint density function is given by

$$f(x_{1,\,1}, x_{1,\,2}, \ldots) = (\lambda e^{-\lambda x_{1,\,1}})(\lambda e^{-\lambda x_{1,\,2}})\ldots\ . \tag{42}$$

Suppose crude Monte Carlo and importance sampling use the same type of input distribution, namely, negative exponential but with different parameters λ and λ_0 respectively. Then the likelihood ratio becomes

$$\frac{f(x_{1,\,1}, x_{1,\,2}, \ldots)}{h(x_{1,\,1}, x_{1,\,2}, \ldots)} = \frac{(\lambda e^{-\lambda x_{1,\,1}})(\lambda e^{-\lambda x_{1,\,2}})\ \ldots}{(\lambda_0 e^{-\lambda_0 x_{1,\,1}})(\lambda_0 e^{-\lambda_0 x_{1,\,2}})\ \ldots}. \tag{43}$$

Obviously this expression can be reformulated to make the computations more efficient.

In the simulation of dynamic systems it is much harder to obtain the optimal new density. Yet in some applications, distributions were derived that did give drastic variance reductions; see Heidelberger (1995), Heidelberger et al. (1996), Rubinstein and Shapiro (1992), and the literature mentioned at the beginning of this appendix.

Appendix 3: Jackknifing control variates

Control variates are based on m i.i.d. pairs, say, $(Y_j, X_{1, j})$ with $j = 1, \ldots, m$; see Appendix 2. Now eliminate pair j, and calculate the following control variate estimator (see (35)):

$$\overline{Y}_{-j, c} = \overline{Y}_{-j} + C_{-j, OLS}[E(\overline{X}_{-j, 1}) - \overline{X}_{-j, 1}] \tag{44}$$

where $\overline{Y}_{-j}$ denotes the sample average of the responses after elimination of Y_j: further $X_{-j, 1}$ denotes the average service time after eliminating replication j, and $C_{-j, OLS}$ denotes the OLS estimator based on the remaining m - 1 pairs. Note that $E(X_{-j, 1}) = E(X_1) = 1/\lambda$. This $Y_{-j; c}$ gives the pseudovalue

$$P_j = m\overline{Y}_c - (m - 1)\overline{Y}_{-j, c} \tag{45}$$

where $\overline{Y}_c$ is the control variate estimator based on all m pairs.

References

Andradottir, S., J.M. Calvin, and P.W. Glynn (1995), Accelerated regeneration for Markov chain simulations. *Probability in the Engineering and Information Sciences*, 9, pp. 497-523.

Atkinson, A.C. and R.D. Cook (1995), D-optimum designs for heteroscedastic linear models. *Journal of the American Statistical Association, 90*, no. 429, pp. 204-212.

Aytug, H., C.A. Dogan, and G. Bezmez (1996), Determining the number of kanbans: a simulation metamodeling approach. *Simulation*, 67, no. 1, pp. 23-32.

BBN (1989), RS/1 software: data analysis and graphics software. BBN, Cambridge, Massachusetts.

Bartels, R. (1982), The rank version of von Neumann's ratio test for randomness, *Journal of the American Statistical Association, 77*, no. 377, pp. 40-46.

Bettonvil, B. (1990), *Detection of important factors by sequential bifurcation*. Tilburg University Press, Tilburg.

Bettonvil, B. and J.P.C. Kleijnen (1997) Searching for important factors in simulation models with many factors: sequential bifurcation. *European Journal of Operational Research*, 96, no. 1, pp. 180-194.

Box, G.E.P. (1952), Multi-factor designs of first order, *Biometrika*, 39, no. 1, pp. 49-57.

Box, G.E.P. and D.R. Cox (1964), An analysis of transformations. *Journal Royal Statistical Society, Series B*, 26, pp. 211-252.

Box, G.E.P. and N.R. Draper (1987), *Empirical model-building with response surfaces*, John Wiley & Sons, New York.

Box, G.E.P. and J.S. Hunter (1961), The $2^{k - p}$ fractional factorial designs, Part 1. *Technometrics*, 3, 1961, pp. 311-351.

Box, G.E.P., W.G. Hunter, and J.S. Hunter (1978) *Statistics for experimenters: an introduction to design, data analysis and model building*. John Wiley & Sons, Inc., New York.

Box, G.E.P. and K.B. Wilson (1951), On the experimental attainment of optimum conditions. *Journal Royal Statistical Society, Series B*, 13, no. 1, pp. 1-38.

Cheng, R.C.H. (1995), Bootstrap methods in computer simulation experiments. *Proceedings of the 1995 Winter Simulation Conference*, edited by C. Alexopoulos, K. Kang, W.R. Lilegdon, D. Goldsman, pp. 171-177.

Cheng, R.C.H. (1997), Searching for important factors: sequential bifurcation under uncertainty. Institute of Mathematics and Statistics, The University of Kent at Canterbury, Canterbury, Kent CT2 7NF, England.

Cheng R.C.H. and J.P.C. Kleijnen (1997), Improved designs of queueing simulation experiments with highly heteroscedastic responses. *Operations Research* (accepted).

Conover, W.J. (1971), *Practical Non-parametric Statistics*. Wiley, New York.

Crane, M.A. and A.J. Lemoine (1977), *An Introduction to the Regenerative Method for Simulation Analysis*, Springer Verlag, Berlin.

Dagenais, M.G. and J.M. Dufour (1996), Pitfalls of rescaling regression models with Box-Cox transformations. *Review of Economics and Statistics* (forthcoming).

Dangerfield, B. and C. Roberts (1996), An overview of strategy and tactics in system dynamics optimization, *Journal of the Operational Research Society,* 47, pp. 405-423.

De Wit M.S. (1997), Uncertainty analysis in building thermal modelling. *Journal of Statistical Computation and Simulation* (accepted).

Donohue J.M. (1994), Experimental designs for simulation. *Proceedings of the 1994 Winter Simulation Conference*, ...

Donohue, J.M. (1995), The use of variance reduction techniques in the estimation of simulation metamodels. *Proceedings of the 1995 Winter Simulation Conference*, edited by C. Alexopoulos, K. Kang, W.R. Lilegdon, and D. Goldsman, pp. 195-199.

Draper, N.R. (1994), Applied regression analysis; bibliography update 1992-93. *Communications in Statistics, Theory and Methods*, 23, no. 9, pp.2701-2731.

Dykstra, R.L. (1970), Establishing the positive definiteness of the sample covariance matrix. *The Annals of Mathematical Statistics*, 41, no. 6, pp. 2153-2154.

Efron, B. (1982), *The Jackknife, the Bootstrap, and Other Resampling Plans*. CBMS-NSF Series, SIAM, Philadelphia.

Efron, B. and R.J. Tibshirani (1993), *Introduction to the Bootstrap*. Chapman & Hall, London.

Ermakov, S.M. and V.B. Melas (1995), *Design and analysis of simulation experiments*. Kluwer, Dordrecht, Netherlands.

Fedorov, V.V. (1972), *Optimal experimental design*. Wiley, New York.

Fishman, G.S. (1989), Focussed issue on variance reduction methods in simulation: introduction, *Management Science, 35*: 1277.

Fossett, C.A., Harrison D., Weintrob H., and Gass S.I. (1991), An assessment procedure for simulation models: a case study, *Operations Research* 39, pp. 710-723.

Friedman, L.W. (1996), The simulation metamodel. Kluwer, Dordrecht, Netherlands.

Fu, M.C. (1994), Optimization via simulation: a review. *Annals of Operations Research*, volume 53, pp. 199-247.

Fürbringer, J.-M. and C.A. Roulet (1995), Comparison and combination of factorial and Monte-Carlo design in sensitivity analysis. *Building and Environment*, 30: 1996, 505-519.

Glynn, P.W. and D.L. Iglehart (1989), Importance sampling for stochastic simulation. *Management Science* 35: 1367-1392.

Gunther, F.L. (1975), The almost regenerative method for stochastic system simulations. Research report no. 75-21, Operations Research Center, University of California, Berkeley.

Gunther, F.L. and R.W. Wolff (1980), The almost regenerative method for stochastic system simulations. *Operations Research*, 28, no. 2, pp. 375-386.

Heidelberger, P. (1995), Fast simulation of rare events in queueing and reliability models. *ACM Transactions on Modeling and Computer Simulation*, 5, no. 1, pp. 43-85.

Heidelberger, P., P. Shahabuddin, and V. Nicola (1996). Bounded relative error in estimating transient measures of highly dependable non-Markovian systems. In *Reliability and Maintenance of Complex Systems*, NATO ASI Series, Springer-Verlag, Heidelberg, Germany.

Ho, Y. and X. Cao (1991) *Perturbation analysis of discrete event systems*. Dordrecht: Kluwer.

Ho, Y.C., L. Shi, L. Dai, and W. Gong (1992), Optimizing discrete event dynamic systems via the gradient surface method. *Discrete event dynamic systems: theory and applications*, vol. 2, pp. 99-120.

Hood, S.J. and P.D. Welch (1993) Response surface methodology and its application in simulation. In *Proceedings of the 1993 Winter Simulation Conference*

Johnson N.J. (1978), Modified t tests and confidence intervals for asymmetric populations. *Journal of the American Statistical Association*, 73, pp. 536-544.

Keijzer, F., J. Kleijnen, E. Mullenders, and A. van Reeken (1981), Optimization of priority class queues, with a computer center case study. *American Journal of Mathematical and Management Sciences*, 1, no. 4, pp. 341- 358. (Reprinted in: Dudewicz, E.J. and Z.A. Karian, *Modern design and analysis of discrete-event computer simulations*. IEEE Computer Society Press, Washington (D.C. 20036-1903), 1985, pp. 298-310.)

Khuri, A.I. (1996), Analysis of multiresponse experiments: a review. *Statistical design and analysis of industrial experiments*, edited by S. Ghosh, Marcel Dekker, New York, 1996, pp. 231-246.

Khuri, A.I. and J.A. Cornell (1996) *Response surfaces: designs and analyses, second edition*. Marcel Dekker, Inc., New York.

Kiefer, J. and J. Wolfowitz (1959), Optimum designs in regression problems. *Annals Mathematical Statistics.*, 30, pp. 271-294.

Kleijnen, J.P.C. (1974), *Statistical Techniques in Simulation; Part I*, Marcel Dekker, Inc., New York.

Kleijnen, J.P.C. (1975a), *Statistical Techniques in Simulation; Part II*, Marcel Dekker, Inc., New York.

Kleijnen, J.P.C. (1975b) A comment on Blanning's metamodel for sensitivity analysis: the regression metamodel in simulation. *Interfaces*, 5, no. 3, pp. 21-23.

Kleijnen, J.P.C. (1975c) Antithetic variates, common random numbers and optimum computer time allocation. *Management Science*, *Application Series*, 21, no. 10, pp. 1176-1185.

Kleijnen, J.P.C. (1987) *Statistical tools for simulation practitioners*. New York: Marcel Dekker.

Kleijnen, J.P.C. (1992) Regression metamodels for simulation with common random numbers: comparison of validation tests and confidence intervals. *Management Science*, 38, no. 8, pp. 1164-1185.

Kleijnen, J.P.C. (1993) Simulation and optimization in production planning: a case study, *Decision Support Systems* 9: 269-280.

Kleijnen, J.P.C. (1995a) Verification and validation of simulation models. *European Journal of Operational Research* 82: 145-162.

Kleijnen, J.P.C. (1995b), Case study: statistical validation of simulation models. *European Journal of Operational Research*, 87, no. 1, pp. 21-34.

Kleijnen, J.P.C. (1995c) Sensitivity analysis and optimization in simulation: design of experiments and case studies. *Proceedings of the 1995 Winter Simulation Conference*, edited by C. Alexopoulos, K. Kang, W.R. Lilegdon, and D. Goldsman, pp. 133-140.

Kleijnen, J.P.C. (1995d), Sensitivity analysis and optimization of system dynamics models: regression analysis and statistical design of experiments. *System Dynamics Review*, 11, no. 4, pp. 275-288.

Kleijnen, J.P.C. (1996) Simulation: runlength selection and variance reduction techniques. *Reliability and Maintenance of Complex Systems*, edited by S. Ozekici et al., Springer-Verlag, Berlin, 1996, pp. 411-428..

Kleijnen, J.P.C, B. Bettonvil, and W. Van Groenendaal (1997) Validation of trace-driven simulation models: a novel regression test. *Management Science* (accepted).

Kleijnen, J.P.C., P.C.A. Karremans, W.K. Oortwijn, and W.J.H. Van Groenendaal (1987), Jackknifing estimated weighted least squares: JEWLS, *Communications In Statistics, Theory and Methods, 16*, no. 3: 747-764.

Kleijnen, J.P.C., G.L.J. Kloppenburg and F.L. Meeuwsen (1986), Testing the mean of an asymmetric population: Johnson's modified t test revisited. *Communications in Statistics*, *Simulation and Computation*, 15, no. 3, pp. 715-732.

Kleijnen, J.P.C. and R.G. Sargent (1997), A methodology for the fitting and validation of metamodels. Tilburg University.

Kleijnen, J.P.C. and C.R. Standridge (1988), Experimental design and regression analysis in simulation: an FMS case study. *European Journal of Operational Research, 33*: 257-261.

Kleijnen J.P.C. and W. Van Groenendaal (1992) *Simulation: a statistical perspective.* Chichester (U.K.): Wiley.

Kleijnen, J.P.C. and W. Van Groenendaal (1995), Two-stage versus sequential sample-size determination in regression analysis of simulation experiments. *American Journal of Mathematical and Management Sciences*, 15, nos. 1&2, pp. 83-114.

Kleijnen, J.P.C, G. Van Ham, and J. Rotmans (1992) Techniques for sensitivity analysis of simulation models: a case study of the CO2 greenhouse effect. *Simulation*, 58, pp. 410-417.

Morris M.D. (1991), Factorial plans for preliminary computational experiments. *Technometrics* 33 (2), 161-174.

Nachtsheim, C.J. (1987), Tools for computer-aided design of experiments. *Journal of Quality Technology*, 19, no. 3, pp. 132-160.

Nachtsheim, C.J., P.E. Johnson, K.D. Kotnour, R.K. Meyer, and I.A. Zualkernan (1996), Expert systems for the design of experiments. *Statistical design and analysis of industrial experiments*, edited by S. Ghosh, Marcel Dekker, New York, pp. 109-131.

Nash, S.G. (1995) Software survey NLP. *OR/MS Today* 22: 60-71.

Olivi, L. (1980), Response surface methodology in risk analysis. *Synthesis and analysis methods for safety and reliability studies*, edited by G. Apostolakis, S. Garibba, and G. Volta, Plenum Publishing Corporation, New York.

Őren, T.I. (1993) Three simulation experimentation environments: SIMAD, SIMGEST, and E/SLAM. In *Proceedings of the 1993 European Simulation Symposium*. La Jolla: Society for Computer Simulation.

Ozdemirel, N.E., G.Y. Yurttas, and G. Koksal (1996), Computer aided planning and design of manufacturing simulation experiments. *Simulation*, accepted.

Pinedo, M. and R.W. Wolff (1982), A comparison between tandem queues with dependent and interdependent service times. *Operations Research*, 30, no. 3, pp. 464-479.

Plackett R.L. and J.P. Burman (1946), The design of optimum multifactorial experiments. *Biometrika*, 33, 305-325.

ProGAMMA (1997), CADEMO, Computer Aided Design of Experiments and Modeling. ProGAMMA, P.O. Box 841, 9700 AV Groningen, The Netherlands.

Ramberg, J.S., S.M. Sanchez, P.J. Sanchez, and L.J. Hollick (1991), Designing simulation experiments: Taguchi methods and response surface metamodels. *Proceedings of the 1991 Winter Simulation Conference*, pp. 167-176.

Rao, C.R. (1959), Some problems involving linear hypothesis in multivariate analysis. *Biometrika*, 46, pp. 49-58.

Rechtschaffner R.L. (1967), Saturated fractions of 2n and 3n factorial designs. *Technometrics*, 9, pp. 569-575.

Rubinstein, R.Y. and A. Shapiro (1993) *Discrete event systems: sensitivity analysis and stochastic optimization via the score function method*, New York: Wiley.

Sacks, J., W.J. Welch, T.J. Mitchell and H.P. Wynn (1989), Design and Analysis of Computer Experiments (Includes Comments and Rejoinder). *Statistical Science*, 4, no. 4, pp. 409-435.

Safizadeh, M.H. and R. Signorile (1994), Optimization of simulation via quasi-Newton methods. *ORSA Journal on Computing*, 6, no. 4, pp. 398-408.

Sanchez, P.J. F. Chance, K. Healy, J. Henriksen, W. D. Kelton, and S. Vincent (1994), Simulation statistical software: an introspective appraisal. *Proceedings of the 1994 Winter Simulation Conference*, edited by J.D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila, pp. 1311-1315.

Saltelli, A,, T.H. Andres, and T. Homma (1995), Sensitivity analysis of model output; performance of the iterated fractional factorial design method. *Computational Statistics & Data Analysis*, 20, pp.387-407.

Schruben, L.W. and B.H. Margolin (1978) Pseudorandom number assignment in statistically designed simulation and distribution sampling experiments, *Journal of the American Statistical Association*, 73, no. 363, pp. 504-525.

Sheriff, Y.S. and B.A. Boice (1994), Optimization by pattern search. *European Journal of Operational Research*, 78, no. 3, pp. 277-303.

St. John, R.C. and N.R. Draper (1975), D-optimality for regression designs: a review *Technometrics*, 17, no. 1, pp. 15-23.

Swain , J.J. (1996), Number crunching: 1996 statistics survey. *OR/MS Today*, 23, no. 1, pp. 42-55.

Tao, Y-H. and B.L. Nelson (1997), Computer-assisted simulation analysis. IEE Transactions, 29, no.3, pp. 221-231.

Tew, J.D. and J.R. Wilson (1994), Estimating simulation metamodels using combined correlation-based variance reduction techniques. *IIE Transactions*, 26, no. 3, pp. 2-16.

Van Groenendaal, W. and J.P.C. Kleijnen (1996), Regression metamodels and design of experiments. *Proceedings of the 1996 Winter Simulation Conference* (edited by J.M. Charnes, D.J. Morrice, D.T. Brunner, and J.J. Swain), pp. 1433-1439.

Webb, S. (1968) Non-orthogonal designs of even resolution. *Technometrics*, 10, pp. 291-299.

Welch, W. J., R. J. Buck, J. Sacks, H. P. Wynn c.s. (1992), Screening, Predicting, and Computer Experiments *Technometrics*, Vol. 34, No. 1, pp. 15-25.

Wolstenholme, E.F. (1990), *System Enquiry: a System Dynamics Approach*. Chichester (United Kingdom): John Wiley & Sons.

Yu, B. and K. Popplewell (1994), Metamodel in manufacturing: a review. *International Journal of Production Research*, 32, no. 4, pp. 787-796.

Zeigler, B. (1976) *Theory of modelling and simulation*. New York: Wiley Interscience.

Acknowledgment

Professor Jerry Banks (Georgia Tech, Atlanta), Drs Bert Bettonvil and Gül Gürkan (both at Tilburg University, Tilburg), Professor Barry Nelson (Northwestern, Evanston), Professor Moshe Pollatschek (Technion, Haifa), and one anonymous referee gave useful comments on preliminary versions of this chapter.

Table 1: Average queue length $\overline{v}$ of 2000 customers in M/M/s simulation started in empty state

						Replication							
Comb.	λ	μ	s	1	2	3	4	5	6	7	8	9	10
				11	12	13	14	15	16	17	18	19	20
1	1	0.9	2	0.56	0.55	0.47	0.52	0.42	0.55	0.44	0.46	0.31	0.52
				0.38	0.56	0.66	0.64	0.56	0.45	0.51	0.49	0.38	0.47
2	1	0.38	4	1.14	0.72	0.78	0.74	1.14	0.67	0.51	1.04	0.86	0.97
				0.82	0.71	0.63	0.81	0.60	0.67	0.49	0.47	0.66	0.62
3	2	1.6	2	1.18	0.78	0.80	1.60	0.87	0.81	0.93	0.71	0.62	0.87
				0.69	1.15	0.98	1.00	0.72	0.84	0.67	0.99	0.90	0.69
4	2	1	4	0.11	0.19	0.16	0.14	0.20	0.14	0.13	0.11	0.20	0.16
				0.17	0.18	0.18	0.23	0.12	0.14	0.16	0.13	0.18	0.23
5	4	2.9	2	1.74	1.10	1.24	1.23	1.31	1.27	1.07	1.64	1.70	1.57
				1.07	1.27	0.90	1.12	1.29	1.13	1.08	1.41	1.19	1.17
6	4	1.69	4	0.41	0.39	0.60	0.31	0.37	0.36	0.42	0.26	0.42	0.40
				0.43	0.57	0.27	0.36	0.38	0.35	0.29	0.29	0.32	0.41

Table 2: Regression analysis of M/M/s example, using standard SAS software

Linear regression: OLS

R-square	0.9078
ADJ R-SQ	0.9054

Parameter estimates

Variable	D.F.	Parameter estimate	Standard error	T for HO: parameter=0	Prob > |T|
INTERCEPT	1	2.82523873	0.10951557	25.798	0.0001
LOGLABDA	1	5.10760536	0.19460843	26.246	0.0001
LOGMU	1	-5.21545	0.19938765	-26.157	0.0001
LOGSERVERS	1	-5.90304	0.18832235	-31.345	0.0001

Covariance of estimates

COVB	INTERCEPT	LOGLABDA	LOGMU	LOGSERVERS
INTERCEPT	0.01199366	0.01576176	-0.0172908	-0.0189349
LOGLABDA	0.01576176	0.03787244	-0.0381753	-0.034371
LOGMU	-0.0172908	-0.0381753	0.03975544	0.035793
LOGSERVERS	-0.0189349	-0.034371	0.0357937	0.03546531

Table 2: (Continued)

Linear regression: CLS

R-square
0.9491

Parameter	Standard errors	T statistic
2.82524	0.11376	24.83509
5.10761	0.20000	25.53805
-5.21545	0.21361	-24.41576
-5.90304	0.19607	-30.10680

COVB

0.012942	0.0180258	-0.020285	-0.02091
0.018026	0.0399989	-0.042134	-0.03712
-0.020285	-0.042134	0.045630	0.040085
-0.020909	-0.03712	0.040085	0.038444

Linear regression: EWLS

R-square
0.9477

ADJ R-SQ
0.9463

Parameter	Standard errors	T statistic
2.80905	0.11300	24.85885
5.08647	0.19873	25.59488
-5.19890	0.21285	-24.42518
-5.87790	0.19492	-30.15545

COVB

0.0127697	0.0178555	-0.020153	-0.020676
0.0178555	0.0394948	-0.04173	-0.036675
-0.020153	-0.04173	0.0453059	0.0397319
-0.020676	-0.036675	0.0397319	0.0379927

Table 3: Input/output per policy for Wolstenholme's (1990) coal transport model

Run	w_1	w_2	w_3	y for	y for	y for
				policy I	policy II	policy III
1	2000	700	150	55.78	56.87	65.87
2	3500	700	150	66.34	57.65	66.34
3	2000	1000	150	56.48	62.09	66.42
4	3500	1000	150	72.63	65.48	72.63
5	2000	700	1200	62.62	74.61	85.16
6	3500	700	1200	87.94	74.76	87.94
7	2000	1000	1200	78.47	80.73	85.60
8	3500	1000	1200	100	93.18	100

Table 4: Estimates of main effects β_h, upon deleting a combination, in policy I;
a blank denotes a nonsignificant effect;
* denotes an estimated effect significant at $\alpha = 0.20$;
all other estimated effects are significant at $\alpha = 0.10$

Combination deleted	β_0	β_1	β_2	β_3	R^2	adj. R^2
1	70.900	10.823	5.995	11.358	0.9772	0.9543
2	72.858	9.520	4.038 *	9.400	0.9316	0.8632
3	72.906	8.821	4.736 *	9.351	0.9202	0.8404
4	73.466	10.129	5.300	8.791	0.9478	0.8955
5	74.053	7.675	2.843 *	11.245	0.9730	0.9461
6	72.320	8.983	4.575 *	9.613	0.9194	0.8387
7	72.271	9.456	4.101	9.463	0.9310	0.8620
8	71.486	8.149		8.679	0.9026	0.8052
None	72.535	9.195	4.363	9.725	0.9314	0.8799

Table 5: A Plackett-Burman design with 12 combinations
+ denotes +1; - denotes -1

1	2	3	4	5	6	7	8	9	10	11
+	-	+	-	-	-	+	+	+	-	+
+	+	-	+	-	-	-	+	+	+	-
-	+	+	-	+	-	-	-	+	+	+
+	+	+	-	+	+	-	-	_	+	+
+	-	+	+	-	-	-	-	-	-	+
+	+	-	+	+	+	+	+	-	-	-
-	+	+	+	-	+	+	-	+	-	-
-	-	+	+	+	-	+	+	-	+	_
-	-	-	+	+	+	-	+	+	-	+
+	-	-	-	+	+	+	-	+	+	-
-	+	-	-	-	+	+	+	-	+	+
-	-	-	-	-	-	-	-	-	-	-

Table 6: A 2^{7-4} design with generators **4** = **1.2**, **5** = **1.3**, **6** = **2.3**, **7** = **1.2.3**

1	**2**	**3**	**4 = 1.2**	**5 = 1.3**	**6 = 2.3**	**7 =1.2.3**
-	-	-	+	+	+	-
+	-	-	-	-	+	+
-	+	-	-	+	-	+
+	+	-	+	-	-	-
-	-	+	+	-	-	+
+	-	+	-	+	-	-
-	+	+	-	-	+	-
+	+	+	+	+	+	+

Table 7: A 2^{8-4} foldover design with generators **4 = 1.2.8**, **5 = 1.3.8**, **6 = 2.3.8**, **7 = 1.2.3**

Run	1	2	3	4	5	6	7	8
1	-	-	-	+	+	+	-	+
2	+	-	-	-	-	+	+	+
3	-	+	-	-	+	-	+	+
4	+	+	-	+	-	-	-	+
5	-	-	+	+	-	-	+	+
6	+	-	+	-	+	-	-	+
7	-	+	+	-	-	+	-	+
8	+	+	+	+	+	+	+	+
9	+	+	+	-	-	-	+	-
10	-	+	+	+	+	-	-	-
11	+	-	+	+	-	+	-	-
12	-	-	+	-	+	+	+	-
13	+	+	-	-	+	+	-	-
14	-	+	-	+	-	+	+	-
15	+	-	-	+	+	-	+	-
16	-	-	-	-	-	-	-	-

Table 8: Central composite designs for two factors

Run	1	2	3	4	5	6	7	8	9
1'	+	-	+	-	c	-c	0	0	0
2'	-	+	-	+	0	0	c	-c	0

Table 9: Intuitive design for the four factors in the FMS simulation in Kleijnen and Standridge (1988)

Run *i*	w_1	w_2	w_3	w_4
1	-	-	-	-
2	+	-	+	0
3	-	+	+	0
4	+	+	-	-
5	-	-	+	+
6	+	-	-	0
7	-	+	-	0
8	+	+	+	+

Table 10: Estimated variances of estimated effects in first-order polynomial regression metamodel for FMS simulation

Effect	Intuitive design	Formal design
$\hat{\beta}_1$	0.5	0.5
$\hat{\beta}_2$	0.5	0.5
$\hat{\beta}_3$	1.0	0.5
$\hat{\beta}_4$	0.5	0.13
$\hat{\beta}_0$	20.6	19.6

Table 11: Cross-validation and estimated factor effects significantly different from zero when $\alpha = 0.30$; first-order polynomial metamodel for four factors in FMS simulation

Run deleted	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_0$
1		557		577	
2		712		500	2032
3		640		700	
4		629		694	
5				658	1962
6				736	
7				536	
none				541	3288
				541	3288

Table 12: Cross-validation and stability of $\hat{\beta}_2$, $\hat{\beta}_4$, $\hat{\beta}_{2,\ 4}$ and $\hat{\beta}_0$ in metamodel with interaction for FMS

Run deleted	$\hat{\beta}_2$	$\hat{\beta}_4$	$\hat{\beta}_{2,4}$	$\hat{\beta}_0$
1	952	1364	-492	776
2	952	1300	-460	776
3	952	1324	-468	776
4	952	1340	-484	776
5	1152	1432	-576	576
6	752	1232	-376	976
7	952	1332	-476	776
8	952	1332	-476	776
none	952	1332	-476	776

Table 13: Estimated local effects on productive hours and lead time in simulated production planning DSS; unit effects $\hat{\beta}_h$ and $\hat{\gamma}_h$; base values $w_{0,h}$

	Effect on productive hours		Effect on lead time	
h	$\hat{\beta}_h$	$\hat{\beta}_h w_{0,h}$	$\hat{\gamma}_h$	$\hat{\gamma}_h w_{0,h}$
1	0.52	62.40	-0.054	-6.48
2	-39.30	-117.90	-1.504	-4.51
3	0.65	78.00	0.072	8.64
4	-18.07	-0.90	150.583	7.53
5	-128.96	-64.48	-16.519	-8.26
6	0.00	0.00	-0.102	-29.38
7	-0.22	-132.00	-0.006	-3.60
8	13.88	20.82	2.963	4.44
9	-1.53	-38.25	1.311	32.78
10	1.39	139.00	0.072	7.20
11	0.03	9.00	0.037	11.10
12	527.23	158.17	8.485	2.55
13	-9.27	-46.35	-6.351	-31.76
14	-0.46	-55.20	-0.145	-17.40

Table 14: Values of decision variables in base run and after a step of 0.005 on steepest ascent path

	Unit effect	Value of variable in	
h	$\hat{\beta}_h$	Base run	Steepest ascent
1	0.52	132	132.0003
2	-39.30	3.3	3.28035
3	0.65	132	132.0003
4	-18.07	0.075	0.065965
5	-128.96	0.55	0.48552
6	0.00	316.8	316.8
7	-0.22	660	659.9999
8	13.88	1.65	1.65694
9	-1.53	27.5	27.44992
10	1.39	110	110.0007
11	0.03	330	330
12	527.23	0.33	0.5936
13	-9.27	5.5	5.4954
14	-0.46	132	131.9998

Table 15: Input combinations in the central composite design with corresponding costs and efficiencies for the coal transport system

Input combination standard variables (x_1, x_2)	Input combination original variables (w_1, w_2)	Cost (£ mln) $y^{(2)}$	Efficiency (%) $y^{(1)}$
(1.00, 1.00)	(2835.00; 972.00)	4.7790	100.00
(-1.00, 1.00)	(2693.25; 972.00)	4.63725	99.59
(1.00, -1.00)	(2835.00; 923.40)	4.68180	100.00
(-1.00, -1.00)	(2693.25; 923.40)	4.54005	99.36
(0.00, 0.00)	(2764.13; 947.70)	4.65953	99.35
(0.00, 0.75)	(2764.13; 965.97)	4.69607	99.59
(0.75, 0.00)	(2817.28; 947.70)	4.71268	100.00
(0.00, -0.75)	(2764.13; 929.48)	4.62309	99.36
(-0.75, 0.00)	(2710.97; 947.70)	4.60637	98.83

Figure 1. Finding $k = 3$ important factors among $K = 128$ factors in Jacoby and Harrison's (1962) example

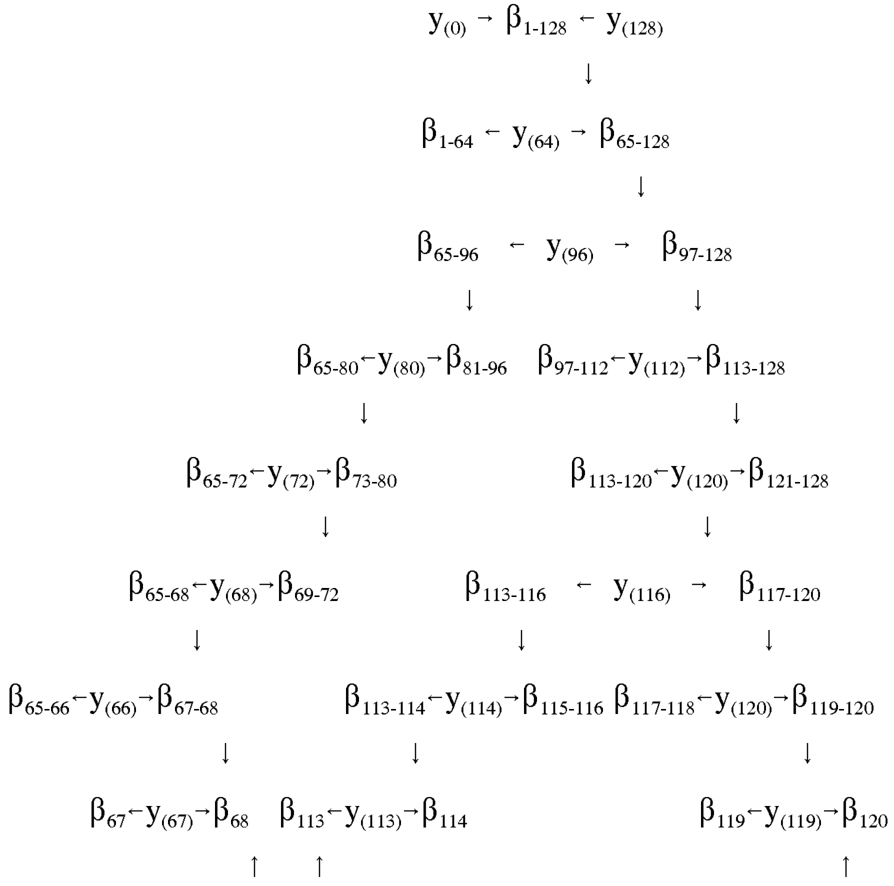
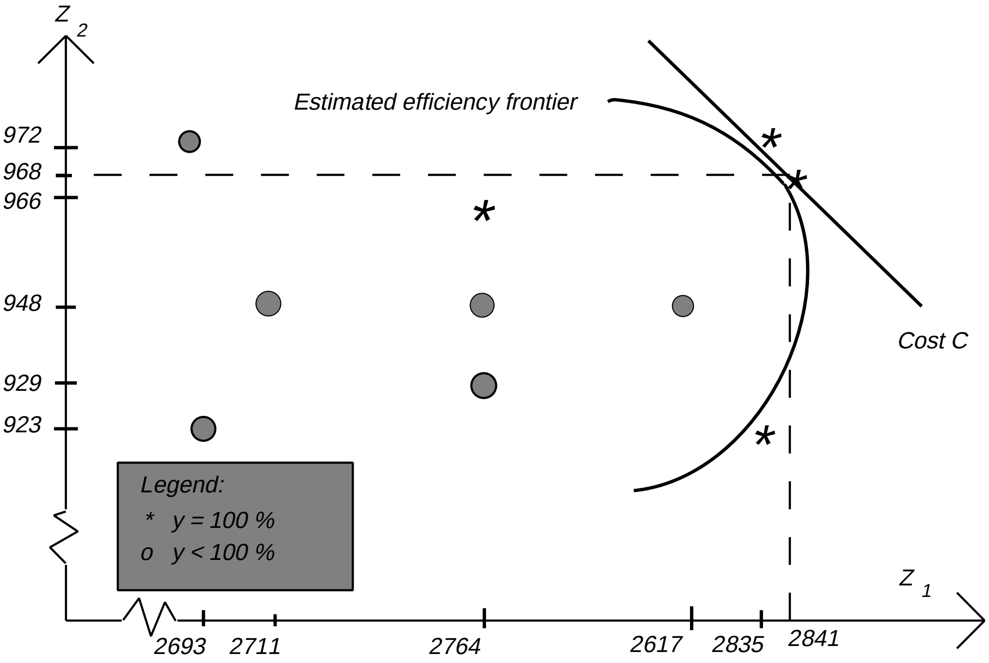

$y_{(0)} \rightarrow \beta_{1\text{-}128} \leftarrow y_{(128)}$
$\beta_{1\text{-}64} \leftarrow y_{(64)} \rightarrow \beta_{65\text{-}128}$
$\beta_{65\text{-}96} \leftarrow y_{(96)} \rightarrow \beta_{97\text{-}128}$
$\beta_{65\text{-}80} \leftarrow y_{(80)} \rightarrow \beta_{81\text{-}96}$
$\beta_{97\text{-}112} \leftarrow y_{(112)} \rightarrow \beta_{113\text{-}128}$
$\beta_{65\text{-}72} \leftarrow y_{(72)} \rightarrow \beta_{73\text{-}80}$
$\beta_{113\text{-}120} \leftarrow y_{(120)} \rightarrow \beta_{121\text{-}128}$
$\beta_{65\text{-}68} \leftarrow y_{(68)} \rightarrow \beta_{69\text{-}72}$
$\beta_{113\text{-}116} \leftarrow y_{(116)} \rightarrow \beta_{117\text{-}120}$
$\beta_{65\text{-}66} \leftarrow y_{(66)} \rightarrow \beta_{67\text{-}68}$
$\beta_{113\text{-}114} \leftarrow y_{(114)} \rightarrow \beta_{115\text{-}116}$
$\beta_{117\text{-}118} \leftarrow y_{(120)} \rightarrow \beta_{119\text{-}120}$
$\beta_{67} \leftarrow y_{(67)} \rightarrow \beta_{68}$
$\beta_{113} \leftarrow y_{(113)} \rightarrow \beta_{114}$
$\beta_{119} \leftarrow y_{(119)} \rightarrow \beta_{120}$

Figure 2. Central composite design, estimated efficiency frontier, and optimal iso-cost line;

means $y^{(1)} = 1$; O means $y^{(1)} < 1$


Z 2
Estimated efficiency frontier
Cost C
Legend:
* y = 100 %
o y < 100 %
Z 1