COMPUTER EXPERIMENTS: DESIGN, MODELING AND INTEGRATION

A Thesis
Presented to
The Academic Faculty

by

Zhiguang Qian

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

School of Industrial and Systems Engineering
Georgia Institute of Technology
August 2006

COMPUTER EXPERIMENTS: DESIGN, MODELING AND INTEGRATION

Approved by:

Dr. C. F. Jeff Wu, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Yasuo Amemiya, Co-Advisor
Department of Mathematical Sciences
IBM T. J. Watson Research Center

Dr. Alexander Shapiro
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Jye-Chyi Lu
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Roshan Joseph Vengazhiyil
School of Industrial and Systems Engineering
Georgia Institute of Technology

Date Approved: 28 April 2006

To my parents and my wife Ruoyan

ACKNOWLEDGEMENTS

Foremost I would like to express my deep gratitude to my advisor, Professor C. F. Jeff Wu. His inspiration, guidance, encouragement and insight helped me through these valuable years at Georgia Tech and the University of Michigan. I am grateful to Dr. Yasuo Amemiya, my co-advisor, for his guidance, active support and encouragement in many phases of my study and research work.

I am thankful to Dr. Alexander Shapiro, Dr. Jye-chyi Lu and Dr. Roshan Joseph Vengazhiyil for serving on my dissertation committee and for their valuable comments and suggestions. I would like to acknowledge my debt to Dr. Alexander Shapiro, Dr. Wei Jiang and Dr. Kwok Tsui for having extended their support to my research. My thanks also go to IBM Research for generously supporting me for three summers.

All of my fellow students at Georgia Tech and the University of Michigan, especially Tirthankar Dasgupta, Roshan Joseph Vengazhiyil, Rohit Kulkarni, Abhyuday Mandal, Jing Wang, Ni Wang, Bowei Xi, Tongling Zhang, made my graduate study a wonderful and enjoyable one. It is my privilege to have befriended them.

Finally, I would like to give my heartful appreciation and gratitude to my parents and my wife Ruoyan for their support and encouragement. This thesis is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

In recent years computer experiments have become popular in engineering and scientific applications. The rapidly increasing use of computer models poses great challenges in design, modeling and analysis of computer experiments. This thesis focuses on developing new methodologies that would meet some of the challenges in the field of computer experiments. It consists of four chapters. Descriptions of these chapters are given below.

Chapter 1 is concerned with building surrogate models based on detailed and approximate simulations. Preliminary design of a complex system often involves exploring a broad design space. This may require repeated use of computationally expensive simulations. To ease the computational burden, surrogate models are built to provide rapid approximations to more expensive models. However, the surrogate models themselves are often expensive to build because they are based on experiments with computationally expensive simulations. An alternative approach is to replace the detailed simulations with simplified approximate simulations, thereby sacrificing accuracy for reduced computational time. Naturally, surrogate models built from these approximate simulations are also imprecise. A strategy is needed for improving and assessing the precision of surrogate models based on approximate simulations without significantly increasing computation. In this work, a new approach is taken to integrate data from approximate and detailed simulations to build a surrogate model that describes the relationship between output and input parameters. Experimental results from approximate simulations form the bulk of the data, and they are used to build a model based on a Gaussian process. The fitted model is then "adjusted" by incorporating a small amount of data from detailed simulations to obtain a more accurate prediction model. The effectiveness of this approach is demonstrated with an example involving the design of cellular materials for an electronics cooling application.

In Chapter 2, a new Bayesian procedure for integrating low-accuracy and high-accuracy

experiments is proposed. Standard practice in analyzing data from different types of experiments is to treat data from each type separately. By borrowing strength across multiple sources, an integrated analysis can produce better results. Careful adjustments need to be made to incorporate the systematic differences among various experiments. To this end, some Bayesian hierarchical Gaussian process models are proposed. The heterogeneity among different sources is accounted for by performing flexible location and scale adjustments. The approach tends to produce prediction closer to that from the high-accuracy experiment. The Bayesian computations are aided by the use of Markov chain Monte Carlo and Sample Average Approximation algorithms. The proposed method is illustrated with two examples: one with detailed and approximate finite elements simulations for mechanical material design and the other with physical and computer experiments for a fluidized bed process in the food industry to coat certain food products with additives.

Chapter 3 is devoted to the development of a structural equation method for the temperature modeling in data center computer experiment. Temperature modeling is a key in designing and running a reliable data center with many computer components operating constantly and generating heat. How different configurations affect the data center thermal distribution is largely unknown. This is because the physical thermal process is complex, depending on many factors, and detailed temperature measurements are not monitored in actual data centers. It is possible to build physics-based mathematical models, implemented in computer code, to study the air movement and temperature distribution mechanisms. Each run in this type of computer experiment takes several days to complete, requiring the stabilization of the algorithm with a large number of reference points. Hence, the use of an efficient and informative experimental design is necessary. A statistical method based on latent variables is introduced for analyzing the multivariate temperature readings produced by the computer experiment. A two-stage estimation procedure is developed for the proposed latent variable model by making use of sufficient statistics and pseud-likelihood method. Also discussed is a method using the fitted statistical model for determining practical configurations of a data center to meet some physical and usage requirements.

Construction of designs for multiple experiments with different levels of accuracy is a

new issue in design of experiments because traditional methods deal almost exclusively with a single experiment. In Chapter 4, a method is proposed for constructing nested space-filling designs for this type of multiple experiments. The construction is aided by the use of Galois field and orthogonal arrays. Multiple design sets generated by the proposed method are guaranteed to have some space-filling property.

CHAPTER I

BUILDING SURROGATE MODELS BASED ON DETAILED AND APPROXIMATE SIMULATIONS

1.1 Introduction

Preliminary design of a complex system often involves exploring a broad design space or region of design variable values. Many detailed analysis programs are available for use in the latter stages of design, but they can be extremely expensive for exploring broad regions. One solution has been to simplify the simulations and obtain data from more approximate simulations. For these approximate simulations, accuracy is sacrificed to reduce computational time. However, when it is desirable to explore a large design space that includes broad ranges of design variables, repeated approximate simulations still generate substantial computational loads.

Another approach is to create surrogate models to replace individual simulations. These surrogate models have been used widely in design. Computer experiments in which the design variables cover a carefully chosen range of values are used to create the surrogate models. Values of the design variables are chosen in specific patterns called experimental designs (Wu and Hamada 2000; Montgomery 1997) and performance is simulated at these points. The responses and input values are combined statistically to create functional relationships between input variables and performance; these functional relationships are the surrogate models. The surrogate models can be used for robust design (Chen et al. 1996) or linked to optimization routines, or they can serve as a bridge for integration across multiple functions (Seepersad et al. 2004) or across different levels of abstraction (Michelena, Park and Papalambors 2002).

Familiar methods for creating surrogate models include response surface modeling (Myers and Montgomery 1995) and kriging (Matheron 1963; Cressie 1988; Laslett 1994), and

an example of their use in design is presented by Chen et al. (1996). However a wide variety of techniques are available (Simpson et al. 2001). In addition to the choice of the metamodeling method, the accuracy of a surrogate model is determined by the experimental design used to select data points, the size of the design space or range of explored values of design variables, the accuracy of the simulation at each data point and the numbers of data points available to compute the surrogate model (Simpson et al. 2001).

In the last decade, methods for improving the accuracy and computational efficiency of metamodeling procedures have been actively studied. One approach has been to successively reduce the design space, thus simultaneously reducing the extent of the approximation of the metamodels. There are several ways to accomplish this, including the use of trust regions (Wujek and Renaud 1988ab; Rodriguez et al. 2001; Akexandrov et al. 1998), heuristics (Chen et al. 1997), move limits (Toropov et al. 1996), and an adaptive response surface method in which the design space is systematically reduced by discarding regions with large objective function values at each modeling-optimization iteration (Wang, Dong and Atchison 2001; Wang 2003). Entropy maximization has also been studied (Farhang-Mehr and Azarm 2001). Wang and Simpson (2004) proposes an intuitive metamodeling method based on hierarchical fuzzy clustering which helps a designer reduce metamodels to regions of interest to a designer.

Another way of reducing the design space is by reducing its dimensionality (Box and Draper 1969). Typically, the design space is screened to identify and remove design variables that are less important. However, it can be difficult to obtain substantial reductions of dimensionality for large-scale problems (Koch et al. 1999). Super-efficient screening methods for removing less important design variables are also available. Both group-screening (Watson 1961) and sequential bifurcation (Bettonvil 1990; Bettonvil and Kleijnen 1996) must be applied cautiously for designs in which multiple responses are considered; screening using supersaturated statistical experimental designs is preferable for situations with multiple responses (Wu 1993; Holcomb, Montgomery and Carlyle 2003).

We believe that the choice of metamodeling method must take into consideration both computational time and metamodel accuracy because different aspects of metamodeling

may be important in different circumstances. Our method involves creating metamodels based on both approximate and detailed (accurate) simulations and thus using information that is developed necessarily when creating the simulations; a preliminary report of our approach has appeared in Qian et al. (2004). Osio and Amon (1996) and Pacheco, Amon and Finger (2003) also propose a multistage kriging method to sequentially update and improve model accuracy. This method is compared with our approach in greater detail in Section 1.2.4. Further, our approach is consistent with space mapping and provides an alternative method for aligning and enhancing a coarse model with a fine model (Bandler et al. 2004; Bakr et al. 2000).

In general there is a trade-off between the accuracy of a surrogate model and the resources needed to build it. If surrogate models are built with a reduced number of data points, they are generally less accurate than models built with a larger number of data points. If detailed, computationally expensive simulations are replaced with approximate simulations, many more data points can be obtained. However, a surrogate model built with approximate information may produce biased results. A practical, alternative strategy is to run a large number of approximate simulations and a smaller number of detailed simulations and then combine the two sets of results to produce a final surrogate model.

In this chapter, we develop a framework in which we can combine results from both detailed simulations and approximate simulations to create surrogate that are as accurate as possible, given the resources available. Since the approximate simulations form the bulk of the data, they are used to build a model based on a Gaussian process that assumes a simple mean part with a flexible residual part. The fitted model is then adjusted by incorporating information from the detailed simulations.

In Section 1.2, we briefly review our approach along with the procedure of Gaussian process modeling that is foundational to it. As an illustration, we apply this approach for designing linear cellular alloys in Section 1.3. Discussions and possible extensions of our approach are presented in Section 1.4.

1.2 Building a surrogate model based on detailed and approximate simulations

Integration of results from detailed simulations (DS) and approximate simulations (AS) is not a straightforward task because the two sets of results have significantly different distributional assumptions. One possible way to combine the AS and DS data is to link them by a simple structure and then build a prediction model for DS directly. This one-step approach has one major disadvantage. Due to the paucity of the DS runs, the resulting surrogate model can be very imprecise and can lead to inaccurate predictions. To overcome this problem and create an accurate surrogate model, we propose a novel *two-step approach* based on Gaussian process modeling. In this work, we assume that the DS produces results that are in agreement with the results from the true process. Thus, we neglect the error in the DS results compared to the true process. This is a reasonable assumption in many computer experiments including the example in Section 1.3. Thus, the objective is to create a surrogate model that can produce predictions close to the DS results.

A generic diagram is presented for the new two-stage approach in Fig. 1. Stage 1 involves designing and generating computer experiments for detailed and approximate simulations. Key to the approach is Stage 2—a novel two-step modeling strategy. This sets our method apart from existing surrogate model building techniques. The basic idea is to use AS results to provide a *base surrogate model* and adjust the model by DS results. The detailed description of these two steps will be given in Sections 1.2.3 and 1.2.4, respectively. Stage 3 consists of the application part of the procedure. When a *final surrogate model* is available, various further investigations, such as optimization, sensitivity analysis, and calibration can be performed.

The modeling part of the procedure consists of the following two steps:

(1) Fit a Gaussian process model using only AS data.

(2) Adjust the fitted model in step 1 with DS data.

Since AS results form the bulk of the data, AS results can be used to fit a smooth response surface in the first step. In the second step, this fitted surface is adjusted by DS

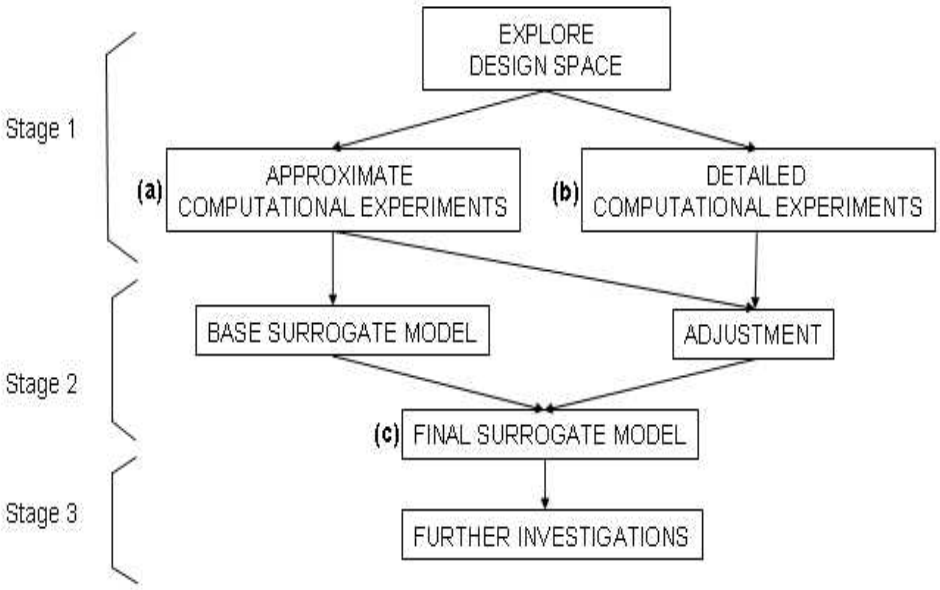

Figure 1: Diagram of the proposed approach for combining detailed and approximate data into a surrogate model.

data, so that the resulting model is close to DS data. The detailed description of these two steps is given in Sections 1.2.2 through 1.2.4.

1.2.1 Gaussian process modeling

Gaussian process modeling (also referred to as a kriging model in spatial statistics and other fields) is widely used in computer experiments because of its many desirable properties (Santner, Williams and Notz 2003). A brief introduction is given here. Suppose that the data consist of n vectors of input variable values denoted by $\mathbf{X} = (\mathbf{x}_1^t, \ldots, \mathbf{x}_n^t)$ for d covariates and the corresponding response values $\mathbf{y} = (y_1, \ldots, y_n)^t$. The Gaussian process

model assumes the following structure:

$$y(\mathbf{x}_i) = \boldsymbol{\beta}^t \mathbf{f}(\mathbf{x}_i) + \epsilon(\mathbf{x}_i), i = 1, \ldots, n, \tag{1}$$

where $\mathbf{f}(\mathbf{x}) = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_m))^t$ is a set of pre-specified functions and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m)^t$ is a set of unknown coefficients. The $\epsilon(\mathbf{x})$ is assumed to be a realization of a stationary Gaussian process with covariance

$$\text{cov}(\epsilon(\mathbf{x}_i), \epsilon(\mathbf{x}_j)) = \sigma^2 R(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp[-d(\mathbf{x}_i, \mathbf{x}_j)]. \tag{2}$$

The correlation function $R(\mathbf{x}_i, \mathbf{x}_j)$ in (2) is a function of the "distance" between $\mathbf{x}_i$ and $\mathbf{x}_j$. If the "distance" is measured as a Euclidean distance, there will be a tendency to give the same weight to all variables and therefore the Euclidean distance cannot be used to distinguish different factor effects. To overcome this, the following flexible "weighted" distance function is adopted:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{h=1}^{d} \theta_h |\mathbf{x}_{ih} - \mathbf{x}_{jh}|^{p_h}, \tag{3}$$

where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)$ and $\mathbf{p} = (p_1, \ldots, p_d)$ in (3) are *scale* and *power* parameters, respectively. The Gaussian correlation is for the case $p_h = 2, h = 1 \ldots, d$, and its associated processes are infinitely differentiable in the mean square sense (Santner, Williams and Notz 2003). As a result, the Gaussian correlation is often adopted in the modeling (Simpson et al. 2001; Welch et al. 2002). In the example given in Section 1.3, we will follow this convention.

In the general case, we observe $\mathbf{y} = (y_1, \ldots, y_n)^t$ and are interested in predicting y at a new point $\mathbf{x}^*$. The empirical best linear unbiased predictor (BLUP) (Santner, Williams and Notz 2003) is adopted to predict the value at an untried $\mathbf{x}^*$

$$\widehat{y}(\mathbf{x}^*) = \mathbf{f}_*^t \widehat{\boldsymbol{\beta}} + \mathbf{r}\mathbf{R}^{-1}(\mathbf{y} - \mathbf{F}\widehat{\boldsymbol{\beta}}), \tag{4}$$

where $\mathbf{r} = (R(\mathbf{x}^*, \mathbf{x}_1), \ldots, R(\mathbf{x}^*, \mathbf{x}_n))^t$, $\mathbf{f}_* = \mathbf{f}(\mathbf{x}^*)$, $\widehat{\boldsymbol{\beta}} = (\mathbf{F}^t\mathbf{R}^{-1}\mathbf{F})^{-1}\mathbf{F}^t\mathbf{R}^{-1}\mathbf{y}$, $\mathbf{R}$ is the $(n \times n)$ matrix with entries $R(\mathbf{x}_i, \mathbf{x}_j)$ for $i, j = 1, \ldots, n$ and $\mathbf{F} = (\mathbf{f}(\mathbf{x}_1)^t, \ldots, \mathbf{f}(\mathbf{x}_n)^t)^t$ is the regression matrix of (1). It can be shown that $\widehat{y}(\mathbf{x}_i)$ equals y_i. Thus, the BLUP smoothly

interpolates all the observed data points. The predictor in (4) involves unknown correlation parameters $\boldsymbol{\theta}$ that can be estimated by maximizing

$$-\frac{1}{2}(n\ln(\widehat{\sigma}^2)) + \ln|\mathbf{R}|), \tag{5}$$

where $\widehat{\sigma}^2 = \frac{(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}})^t\mathbf{R}^{-1}(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}})}{n}$. In the example in Section 3, a version of quasi-Newton algorithm (Byrd et al. 1995), implemented in the optim function in R (R Development Core Team 2004), is used to solve the optimization problem in (5). The estimated $\mathbf{r}$ and $\mathbf{R}$ will be denoted as $\widehat{\mathbf{r}}$ and $\widehat{\mathbf{R}}$.

1.2.2 Modeling the approximate simulation data

Using the Gaussian process modeling described in Section 1.2.1, we now develop an approach for building a surrogate model. We first build a surrogate model based on the approximate simulations only. This model is further refined later. Usually only a constant term (i.e., $\boldsymbol{\beta}^t\mathbf{f}(\mathbf{x}_i) = \beta_0$ in (1)) is used in the mean part of the Gaussian process model (Welch et al. 1992). However, in some circumstances it is reasonable to assume that the factors considered in the experiment have linear effects on the output (Handcock and Stein 1993; Handcock and Wallis 1994). By following this convention, we choose the model below for the output of the approximate simulation y_a,

$$y_a(\mathbf{x}) = \beta_{a0} + \sum_{h=1}^{d}\beta_{ah}x_h + \epsilon_a(\mathbf{x}), \tag{6}$$

where $\beta_{a0} + \sum_{h=1}^{d}\beta_{ah}x_h$ is the linear mean part and $\epsilon_a(\mathbf{x})$ is the residual part that is assumed to be a stationary Gaussian process with mean zero, variance σ_a^2 and correlation parameters $\boldsymbol{\theta}_a$. Because a large number of AS runs are available, ($\boldsymbol{\beta}_a$, σ_a^2, $\boldsymbol{\theta}_a$) can usually be estimated accurately. The BLUP for $y_a(\mathbf{x}^*)$ at an untried $\mathbf{x}^*$is

$$\widehat{y}_a(\mathbf{x}^*) = \mathbf{f}_a^t\widehat{\boldsymbol{\beta}}_a + \widehat{\mathbf{r}}_a\widehat{\mathbf{R}}_a^{-1}(\mathbf{y}_a - \widehat{\mathbf{F}}_a\widehat{\boldsymbol{\beta}}_a), \tag{7}$$

where $\mathbf{f}_a$, $\widehat{\mathbf{r}}_a$, $\widehat{\mathbf{R}}_a$ and $\widehat{\mathbf{F}}_a$ are defined as in Section 1.2.1. Throughout the remaining part of this chapter, we shall refer to the model in (7) as the *base surrogate model*.

1.2.3 Adjustment based on detailed simulation data

Because approximate and detailed models typically differ by modeling assumptions, numerical solution methods, mesh resolutions, and other factors, the associated data values can be moderately or significantly different. For the example analyzed in Section 1.3, when the same input values are used for the AS and DS, the worst-case difference between AS and DS results is on the order of 16% with respect to the DS value. Therefore the DS data can be used to adjust the base surrogate model. The accuracy of the adjusted model depends on the degree of difference between AS and DS results and the parametric relationship between the AS and DS results. Because these are all computer experiments, the results are deterministic, and there is no experimental error to consider. In this case, we simplify the adjustment procedure by modeling the adjustment terms conditioned on the value of y_a. If n_d AS runs share the same input values as n_d DS runs, a very simple adjustment can be done by using a *location-scale adjustment*, i.e.,

$$y_d(\mathbf{x}_i) = \rho y_a(\mathbf{x}_i) + \delta, i = 1, \ldots, n_d, \tag{8}$$

However, some cases may also exhibit a non-linear discrepancy between AS and DS. As an extension of the above procedure, a more sophisticated adjustment can be obtained by making the following two changes in (8): (a) substitute the constant ρ with a linear regression function $\rho(\mathbf{x})$, and (b) replace the constant δ by a Gaussian process $\delta(\mathbf{x})$. These modifications lead to the following model:

$$y_d(\mathbf{x}_i) = \rho(\mathbf{x}_i) y_a(\mathbf{x}_i) + \delta(\mathbf{x}_i), i = 1, \ldots, n_d, \tag{9}$$

where

$$\rho(\mathbf{x}_i) = \rho_0 + \sum_{j=1}^{d} \rho_j x_{ij} \tag{10}$$

is the linear regression function. Conditioning on y_a, $\delta(\mathbf{x})$ is assumed to be a stationary Gaussian process with mean δ_0, variance σ_δ^2 and correlation parameters $\boldsymbol{\theta}_\delta$. Thus, conditioning on $(y_a(\mathbf{x}_i), \ldots, y_a(\mathbf{x}_{n_d}))$, the distribution of $\mathbf{y}_d = (y_d(\mathbf{x}_i), \ldots, y_d(\mathbf{x}_{n_d}))^t$ is normal and the log likelihood of $\mathbf{y}_d$, up to an additive constant, can be written as

$$-\frac{1}{2}[n_d \ln \sigma_\delta^2 + \ln |\mathbf{R}_\delta| - \frac{(\mathbf{y}_d - \mathbf{F}_d \boldsymbol{\alpha})^t \mathbf{R}_\delta^{-1} (\mathbf{y}_d - \mathbf{F}_d \boldsymbol{\alpha})}{2\sigma_\delta^2}], \tag{11}$$

where $\mathbf{F}_d$ is the regression matrix

$$\begin{pmatrix} 1, & y_a(\mathbf{x}_1), & y_a(\mathbf{x}_1)x_{11}, & \cdots, & y_a(\mathbf{x}_1)x_{1d} \\ & \cdots & \cdots & \cdots & \cdots \\ 1, & y_a(\mathbf{x}_{n_d}), & y_a(\mathbf{x}_{n_d})x_{n_d1}, & \cdots, & y_a(\mathbf{x}_{n_d})x_{n_dd} \end{pmatrix}$$

and $\boldsymbol{\alpha} = (\delta_0, \rho_0, \rho_1, \cdots, \rho_d)^t$ is the collection of unknown parameters associated with the mean part in (9). The estimates $\widehat{\boldsymbol{\alpha}}$ and $\widehat{\boldsymbol{\theta}}_\delta$ can be obtained by maximizing the function in (11). The optimization procedure is very similar to the one described in Section 1.2.1, so its details are omitted.

For given values of $\widehat{\rho}_i$'s $(i = 0, \ldots, d)$, we can compute the values of $\boldsymbol{\delta} = (\delta(\mathbf{x}_1), \ldots, \delta(\mathbf{x}_{n_d}))$ by using

$$\delta(\mathbf{x}_i) = y_d(\mathbf{x}_i) - \widehat{\rho}(\mathbf{x}_i) y_a(\mathbf{x}_i), i = 1, \ldots, n_d, \tag{12}$$

where

$$\widehat{\rho}(\mathbf{x}_i) = \widehat{\rho}_0 + \sum_{j=1}^{d} \widehat{\rho}_j x_{ij} \tag{13}$$

is the fitted regression function for the scale adjustment.

At an untried point $\mathbf{x}^*$, a BLUP predictor can be constructed as

$$\widehat{\delta}(\mathbf{x}^*) = \widehat{\delta}_0 + \widehat{\mathbf{r}}_\delta \widehat{\mathbf{R}}_\delta^{-1} (\boldsymbol{\delta} - \mathbf{F}_\delta \widehat{\delta}_0), \tag{14}$$

where $\widehat{\mathbf{r}}_\delta$ and $\widehat{\mathbf{R}}_\delta$ are defined in Section 1.2.1, and $\widehat{\delta}_0$ is obtained previously as part of $\widehat{\boldsymbol{\alpha}}$. The predictor $\widehat{\delta}(\mathbf{x}^*)$ in (14) is used as a building block to establish the final surrogate model.

1.2.4 Building and evaluating the final surrogate model

From the base surrogate model in (7) and the adjustments results in (13) and (14), a simple plug-in method is used to establish the *final surrogate model* for an untried $\mathbf{x}^*$,

$$\widehat{y}_d(\mathbf{x}^*) = \widehat{\rho}(\mathbf{x}^*) \widehat{y}_a(\mathbf{x}^*) + \widehat{\delta}(\mathbf{x}^*), \tag{15}$$

where $\widehat{\rho}(\mathbf{x}^*)$ is the fitted scale adjustment term in (13), $\widehat{y}_a(\mathbf{x}^*)$ is the predicted value from the base surrogate model in (7), and $\widehat{\delta}(\mathbf{x}^*)$ is the fitted location adjustment term in (14). As mentioned in Section 1.2.1, the prediction from the base surrogate model is not very

accurate. Because we have adjusted this model using detailed simulation data, the prediction from (15) will be closer to the output from the detailed simulations than the prediction from the base surrogate model (7). In addition, it can be shown that the final surrogate model, $\widehat{y}_d(\cdot)$ in (15), smoothly interpolates all the detailed simulation data. This is another benefit of our two-step procedure. If we are interested in making accurate predictions based on detailed simulations in some regions of specific interest, we can select a few more points in these regions and conduct the appropriate detailed simulations.

In some situations, the multistage Bayesian approach proposed by Osio and Amon (1996) and Pacheco, Amon and Finger (2003) can be adapted to deal with approximate and detailed simulations data. In their approach, a kriging model is fit to the AS data. Then this model is used as the prior mean for modeling DS data. In comparison with our approach, the first stage modeling with AS data is exactly the same. The difference is in the second stage. It is well known that a kriging predictor is pulled towards the prior mean in regions where data are scarce. Thus in their approach, the final surrogate model will pass through the DS data due to the interpolating property, but it will be pulled towards the base surrogate model in regions where DS data are not available. This feature can lead to a rough final surrogate model, particularly when the DS is very different from the AS. In contrast, we only do a location and scale adjustment and therefore, the profile of the base surrogate model is approximately preserved. Our approach is more suitable when there are very few DS data points compared with AS data, a characteristic of our example.

To illustrate our approach in the next section, we consider the design of a linear cellular material, which is used to dissipate heat from a microprocessor.

1.3 Designing linear cellular materials with the surrogate model building approach

Consider the design of a heat exchanger for a representative electronic cooling application. As illustrated in Fig. 2, the device is used to dissipate heat generated by a heat source such as a microprocessor. The mechanism for heat dissipation is forced convection via air with entry temperature, T_{in}, in degrees Kelvin and total mass flow rate, $\dot{m}$, measured in

kilograms per second. Steady state, incompressible laminar flow is assumed. The device is assumed to have fixed overall width (W), depth (D), and height (H) of 9, 25, and 17.4 millimeters, respectively. It is insulated on the left, right, and bottom sides and is subjected to a heat source at constant temperature, T_{wall}, in degrees Kelvin on the top face.

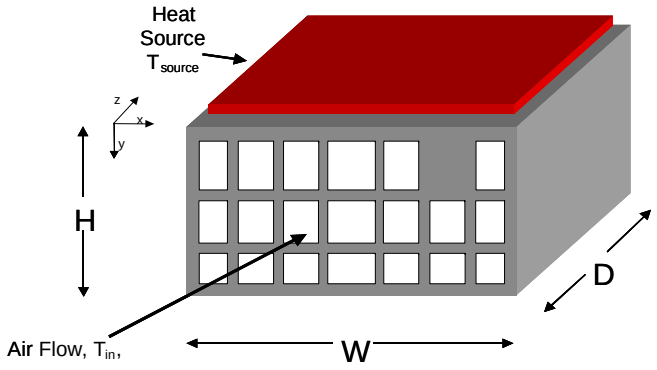

Figure 2: Compact, forced convection heat exchanger with graded rectangular linear cellular alloys.

The device is comprised of linear cellular material—ordered, metallic cellular material with extended prismatic cells. These materials can be produced with nearly arbitrary two-dimensional topologies, metallic base materials, and wall thicknesses as small as 50 microns via a thermo-chemical extrusion fabrication process developed at Georgia Tech (Cochran et al. 2000). Prismatic cellular materials have a combination of properties that make them especially suitable for many multifunctional applications, including actively cooled, lightweight structures (Seepersad et al. 2004; Gibson and Ashby 1997; Hayes et al. 2001; Evans et al. 2004). Although cell topology and dimensions can be varied, the

prismatic cellular material is composed exclusively of rectangular cells for this example. There are four columns of cells with interior cell widths of 2 mm, and three rows of cells with interior cell heights of 10, 5, and 2 mm for the uppermost, middle, and lower rows of cells, respectively. The solid material in the walls of the prismatic cellular material is assumed to have thermal conductivity, k, in Watts per meter-Kelvin.

The design objective is to maximize the total rate of steady state heat transfer achieved by the device. Some of the factors affecting this objective include the topology and dimensions of the cells and cell walls, the flow rate and temperature of the incoming air, the temperature of the heat source, and the thermal conductivity of the solid material in the walls of the device. In other design activities, we have adjusted the dimensions of the device (Seepersad et al. 2004); here, we intend to explore the heat transfer rate as a function of the mass flow rate of entry air, $\dot{m}$, the temperature of entry air, T_{in}, the temperature of the heat source, T_{wall}, and the solid material thermal conductivity, k.

To analyze the impact of these factors on heat transfer rates, we use two types of simulations—computationally expensive FLUENT (Fluent 1998) finite element simulations and relatively fast but more approximate finite difference simulations. Details of the two approaches are available in the literature, but it is important to highlight their differences and their relative costs and benefits in terms of accuracy and computational time. First, the models are based on different methods. The finite difference approach, used here for approximate simulations (AS), is a numerical technique for solving two- or three-dimensional heat transfer problems (Incropera and DeWitt 1996). Finite difference models are based on difference equations that approximate continuous variables as quantities at discrete points or nodes on a grid (Incropera and DeWitt 1996). FLUENT is a commercial software package for analyzing fluid flow and heat transfer problems with a computational fluid dynamics (CFD) solver (Fluent 1998). FLUENT models, used here as detailed simulations, are based on finite volume methods that approximate governing partial differential equations over a control volume and are more flexible than finite difference methods that require a structured mesh (Fluent 1998). FLUENT models also account for details such as entry effects that are not modeled explicitly in the finite difference models. Secondly, as described for the present

example by Seepersad et al. (2004), the FLUENT grid is approximately four times denser than the finite difference grid for this example. Finally, for examples similar to the present one, each FLUENT simulation requires two to three orders of magnitude more computing time than the corresponding finite difference simulation. For example, on a 2.0 GHz Pentium 4 PC with 1 GB of RAM, the first data point in Table 2 requires approximately 1.75 hours of computing time for a FLUENT (DS) simulation versus approximately 2 seconds for the finite difference simulation, However, the FLUENT simulations are generally more accurate than the finite difference simulations by 10 to 15% or more.

Our objective is to build a surrogate model that can be used in the design process and represents the functional relationship between design factors and the total rate of steady state heat transfer. To build the surrogate model, we utilize results from both FLUENT and finite difference simulations. A large number of data points are generated using the finite difference simulation with fewer data points obtained from the FLUENT simulation. We show that even a limited amount of data from FLUENT simulations can be used to improve the accuracy of surrogate models based on approximate finite difference models alone.

1.3.1 Generating design points for detailed and approximated simulations

An orthogonal array-based Latin hypercube design (Santner, Williams and Notz 2003) with a run size of 64 data points is used to determine the appropriate set of approximate (finite difference) simulations. The assumed ranges of design variables are shown in Table 1. The Latin Hypercube design has good space-filling properties. This can be seen in Fig. 3 in which the four-variable design is projected onto spaces of two variables. For each pair of variables the data points are uniformly distributed in each of the 64 reference square bins. Also, if we divide each bin in Fig. 3 into 8 equally spaced new bins with smaller size (64 new bins in each dimension), we find that each individual variable in each dimension has a nearly uniform distribution in these 64 bins. Among these 64 approximate simulation experiments, results for detailed simulations are generated for 22 of them. Sixteen of the twenty-two experiments are identified using a simulated annealing algorithm and a minimax

Table 1: Assumed ranges for design variables values

	Design Variables			
	$\dot{m}(kg/s)$	$T_{in}(K)$	$k(W/mK)$	$T_{wall}(K)$
Lower bound	0.00055	270.00	202.4	330
Upper Bound	0.001	303.15	360.0	400

distance criterion (Santner, Williams and Notz 2003). The remaining six detailed simulation experiments are chosen with a roughly uniform distribution in the portion of the design space in which the value of air flow rate, $\dot{m}$, of entry air is small. Background information suggests that there may be a special relationship between the detailed (FLUENT) results, y_d, and the approximate (finite difference) results, y_a, in this subregion. The six additional points are added to explore this relationship. The sample data and corresponding response values are listed in Table 2. In this table, the results for the 64 approximate experiments are shown in the y_a column, and the 22 detailed simulation experiments are listed in the y_d column. It is clear from Table 1 that the four input variables have very different scales. These variables are standardized (subtracting their means and multiplying by the reciprocal of their standard deviations) before the analysis.

1.3.2 Building a base surrogate model

The first step is to build a surrogate model using the approximate simulation results only. Based on background knowledge of the physics of this problem, we know that there should be a significant linear component in the relationship between the response and the four factors. As a result, a linear structure is included when modeling the mean part of the Gaussian process in (9). As described in Section 1.2, the maximum likelihood method is used for estimation. Table 3 lists the linear main effects $\widehat{\beta}_{ai}$ for $i = 1, \ldots, 4$ (corresponding to $\dot{m}$, T_{in}, k, and T_{wall}, respectively) with their p-values for the t-test for $i = 1, \ldots, 4$ and $\widehat{\sigma}_a^2$. The linear main effects for T_{in} and T_{wall} are relatively large, -2.77 and 5.450, respectively and their p-values are quite small, 1.59e-08 and 1.543e-22, respectively; therefore T_{in} and T_{wall} are the two most significant factors. The values of $\widehat{\beta}_{a2}$ and $\widehat{\beta}_{a4}$ have different signs, implying that T_{in} and T_{wall} have opposite effects on the response. This agrees with the

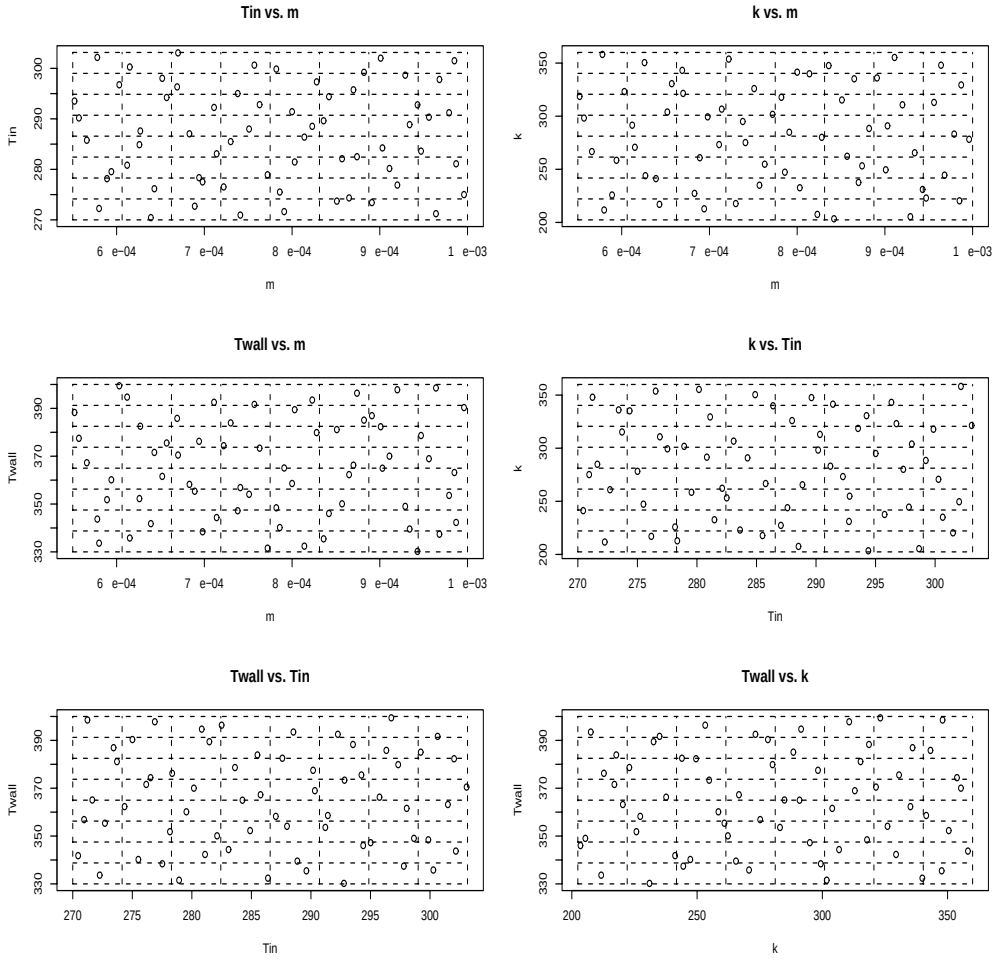

Figure 3: 64 points of an orthogonal array-based Latin hypercube sample. In each plot, there is one point in each of the square bins bounded by dashed lines.

Table 2: Sample data for approximate and detailed simulations

	Design variables				Responses	
	1	2	3	4	5	6
Run	$\dot{m}(kg/s)$	$T_{in}(K)$	$k(W/mK)$	$T_{wall}(K)$	y_a	y_d
1	0.000552	293.53	318.63	388.29	25.61	23.54
2	0.000557	290.18	298.27	377.49	23.24	
3	0.000566	285.77	266.71	367.27	21.23	20.15
4	0.000578	302.17	358.13	343.72	11.44	10.17
5	0.000580	272.26	211.71	333.65	15.03	15.29
6	0.000589	278.16	225.78	351.83	18.55	18.39
7	0.000594	279.54	258.51	360.13	20.74	20.52
8	0.000603	296.75	323.15	399.45	28.40	
9	0.000612	280.83	291.53	394.72	30.22	30.12
10	0.000615	300.28	270.74	335.79	9.53	
11	0.000626	284.89	350.46	352.29	18.13	18.17
12	0.000627	287.60	243.96	382.54	25.02	24.68
13	0.000639	270.45	241.21	341.81	17.92	19.05
14	0.000643	276.17	216.99	371.60	24.20	24.96
15	0.000652	298.04	303.96	361.58	17.47	16.95
16	0.000657	294.24	330.63	375.53	22.48	22.3
17	0.000669	296.33	343.16	385.81	25.07	
18	0.000670	303.07	321.41	370.48	18.93	
19	0.000683	287.05	227.31	358.24	18.61	
20	0.000689	272.70	260.91	355.37	21.31	
21	0.000694	278.35	212.79	376.24	25.11	
22	0.000698	277.52	299.39	338.40	16.02	
23	0.000711	292.26	273.31	392.54	27.47	
24	0.000714	283.08	306.69	344.34	16.43	
25	0.000722	276.53	353.75	374.41	26.50	
26	0.000730	285.51	217.74	383.92	25.88	
27	0.000738	295.01	295.02	347.22	14.37	
28	0.000741	270.95	275.19	356.87	22.36	
29	0.000751	287.99	326.02	354.08	18.17	19.57
30	0.000757	300.64	235.03	391.68	14.37	
31	0.000763	292.82	254.84	373.38	21.96	23.33
32	0.000772	278.93	301.75	331.55	14.02	
33	0.000782	299.86	317.84	348.41	13.68	
34	0.000786	275.51	247.29	340.19	16.82	
35	0.000791	271.64	284.88	365.09	25.06	
36	0.000800	291.42	341.48	358.59	18.83	
37	0.000803	281.47	232.64	389.46	28.69	
38	0.000814	286.39	339.92	332.40	12.68	14.36
39	0.000823	288.53	207.55	393.49	27.96	
40	0.000828	297.33	280.13	379.86	23.17	
41	0.000836	289.62	347.65	335.44	12.79	
42	0.000842	294.39	203.45	346.05	13.75	15.12
43	0.000851	273.71	315.27	381.14	29.08	34.8
44	0.000857	282.12	262.30	350.10	18.25	21.31
45	0.000865	274.35	335.16	362.30	23.89	
46	0.000870	295.76	237.65	366.25	19.36	
47	0.000874	282.50	253.25	396.36	30.90	36.11
48	0.000882	299.22	288.45	385.07	24.45	27.36
49	0.000891	273.43	336.04	386.95	31.05	
50	0.000901	302.02	249.57	382.33	22.64	
51	0.000903	284.25	290.90	364.99	22.22	25.37
52	0.000911	280.17	355.34	370.03	25.03	
53	0.000920	276.89	310.73	397.78	33.27	
54	0.000929	298.65	205.40	349.02	13.67	
55	0.000934	288.86	265.53	339.54	13.89	
56	0.000943	292.77	231.01	330.19	10.16	
57	0.000947	283.62	222.95	378.66	25.48	
58	0.000956	290.33	312.97	368.96	22.22	
59	0.000964	271.23	348.00	398.52	35.05	
60	0.000968	297.80	244.50	337.41	10.99	
61	0.000979	291.21	283.10	353.60	17.45	
62	0.000985	301.50	220.37	363.20	17.14	
63	0.000987	281.11	329.45	342.32	16.95	
64	0.000996	275.01	278.27	390.35	31.35	

Table 3: Results of estimation

	$\widehat{\beta}_{a0}$	$\widehat{\beta}_{a1}$	$\widehat{\beta}_{a0}$	$\widehat{\beta}_{a0}$	$\widehat{\beta}_{a0}$	$\widehat{\sigma}_a^2$
Values	20.606	0.409	-2.77	0.673	5.450	3.352
P-values		0.449	1.59e-08	0.106	1.543e-22	

known physics of the problem, i.e., a decrease in T_{in} or an increase in T_{wall} causes an increase in the total rate of steady state heat transfer. As shown in Table 3, the p-values for $\widehat{\beta}_{a1}$ and $\widehat{\beta}_{a3}$ are quite large. Therefore, $\dot{m}$ and k do not have significant linear main effects on the response in this region of the design space.

The maximum likelihood estimators for the correlation parameters $\widehat{\boldsymbol{\theta}}_a$ are (1.1780, 0.904, 0.300, and 0.01). These values are quite different from each other; therefore different factors affect the correlation of two close points in different scales. Among them, the correlation parameters for $\dot{m}$ and T_{in} are relatively high. The responses of two points, even if there is a small distance between them in the $\dot{m}$-dimension or the T_{in}-dimension, may still have a low correlation. Note that $\dot{m}$ does not have a significant linear main effect but has a large value for its correlation parameter. This implies that the relationship between $\dot{m}$ and the response is nonlinear. This observation may aid our understanding of its physical relationship.

The data used to build the base surrogate model cannot be used to assess the fit of the model, because the Gaussian process model interpolates the training data. Therefore, we generate a testing set of 14 AS runs and compare the prediction results using the base surrogate model and the observed values of these 14 runs. The data is also used to validate the final surrogate model, so a detailed description of these runs is deferred to Section 3.5. Columns $\widehat{y}_a$ and $\widehat{y}_a$ in Table 4 of Section 3.5 give the values of predictions and the responses from the approximate simulations. The root-mean-square-errors (RMSE) for these 14 runs are only 2.588. This is relatively small, since the mean of the values of y_a is 21.499 and the range (max-min) is 29.54. Thus, the base surrogate we constructed for y_a is a decent proxy.

The basic surrogate model is consistent with our background knowledge of the physics of the problem. In general, one would expect the mass flowrate, $\dot{m}$, the temperature of the heat source, T_{wall}, and the thermal conductivity of the material, k, to have positive

linear main effects on the total rate of steady state heat transfer; on the other hand, T_{in} should have a negative linear main effect. The signs of the linear main effects in Table 3 correspond to our expectations. Also, one would expect the temperatures, T_{in} and T_{wall}, to have more significant linear main effects on the response than the mass flowrate, $\dot{m}$, or the thermal conductivity, k—two factors that have much more complex relationships with the response via the Reynold's number and the temperature gradients throughout the structure, respectively. Their linear main effects are dominated in this region of the design space by the strong linear relationship between the temperatures and the response. However, we might expect them to have significant nonlinear relationships with the response, and we observe this for the mass flowrate, $\dot{m}$.

1.3.3 Using detailed simulation data to adjust the base surrogate model

Both y_d and y_a are generated for 22 factor level combinations. Fig. 4 presents a plot of y_d vs. y_a for these 44 experiments. It is clear that the detailed simulation and the approximate simulation values are quite different. Some detailed simulation values are higher than approximate simulation values, while some are lower. This demonstrates the need for modeling $\rho(\mathbf{x})$ as a function of $\mathbf{x}$ in (13).

Next we use the more accurate detailed simulation output, $y_d(\mathbf{x}_i)$, to adjust the fitted model of $y_a(\mathbf{x}_i)$, as described in Section 1.2.4. Overall, we have a good fit for the adjusted model as $\widehat{\sigma}_\delta^2$ has a small value of 0.00515. For the scale adjustment term $\rho(\mathbf{x})$ the parameter estimates are $(\widehat{\rho}_0, \widehat{\rho}_1, \widehat{\rho}_2, \widehat{\rho}_3, \widehat{\rho}_4) = (1.130, 0.090, -0.032, 0.004, -0.012)$. Among these estimates, the coefficients for $\dot{m}$ and T_{in} are relatively large with significant p-values of 2.165e-23 and 3.839e-13. For the location adjustment term $\delta(\mathbf{x})$, the results are $\widehat{\delta}_0 = -0.690$ with the p-value 0.0102 and $\widehat{\boldsymbol{\theta}}_\delta = (0.173, 0.176, 0.01, 3.66)$. In Fig. 5, plots of $\widehat{\delta}$ vs. different pairs of variables are plotted. In each plot, a 40 by 40 equally spaced grid is chosen for the two variables used for plotting and the values of the other two remaining variables are fixed at their mean values.

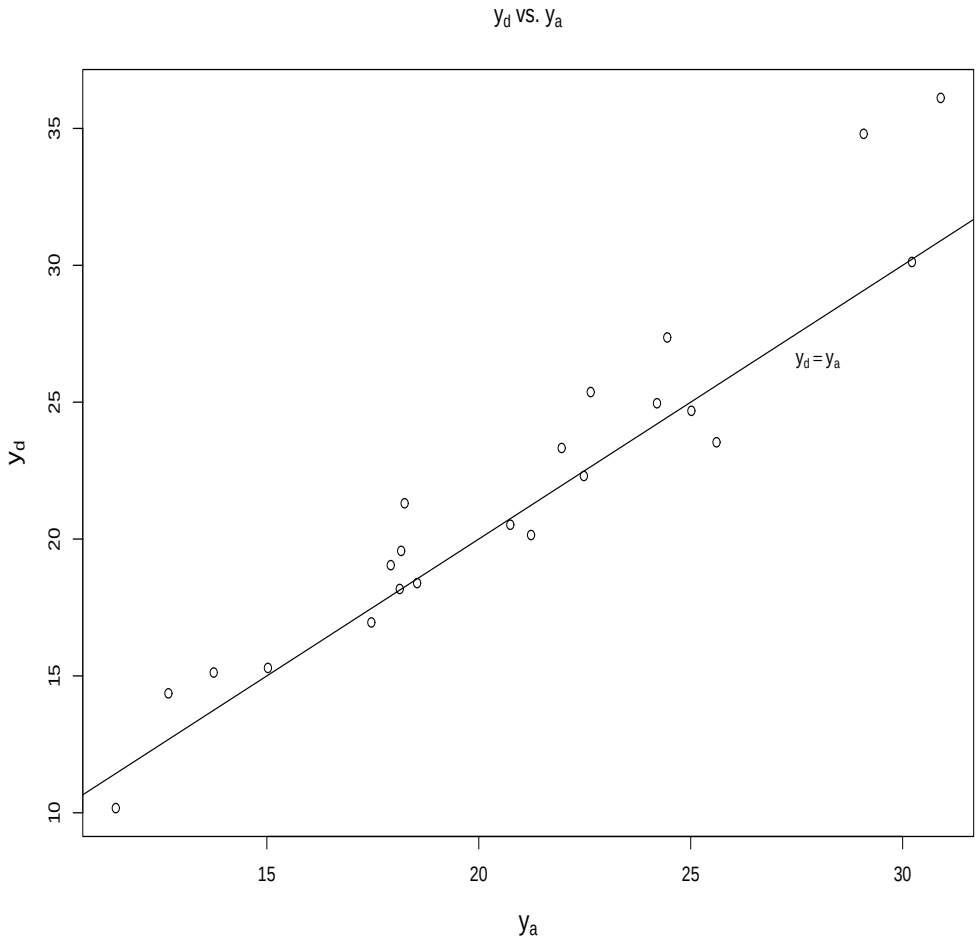

Figure 4: y_d vs. y_a for the same design values, where the straight line is $y_d = y_a$.

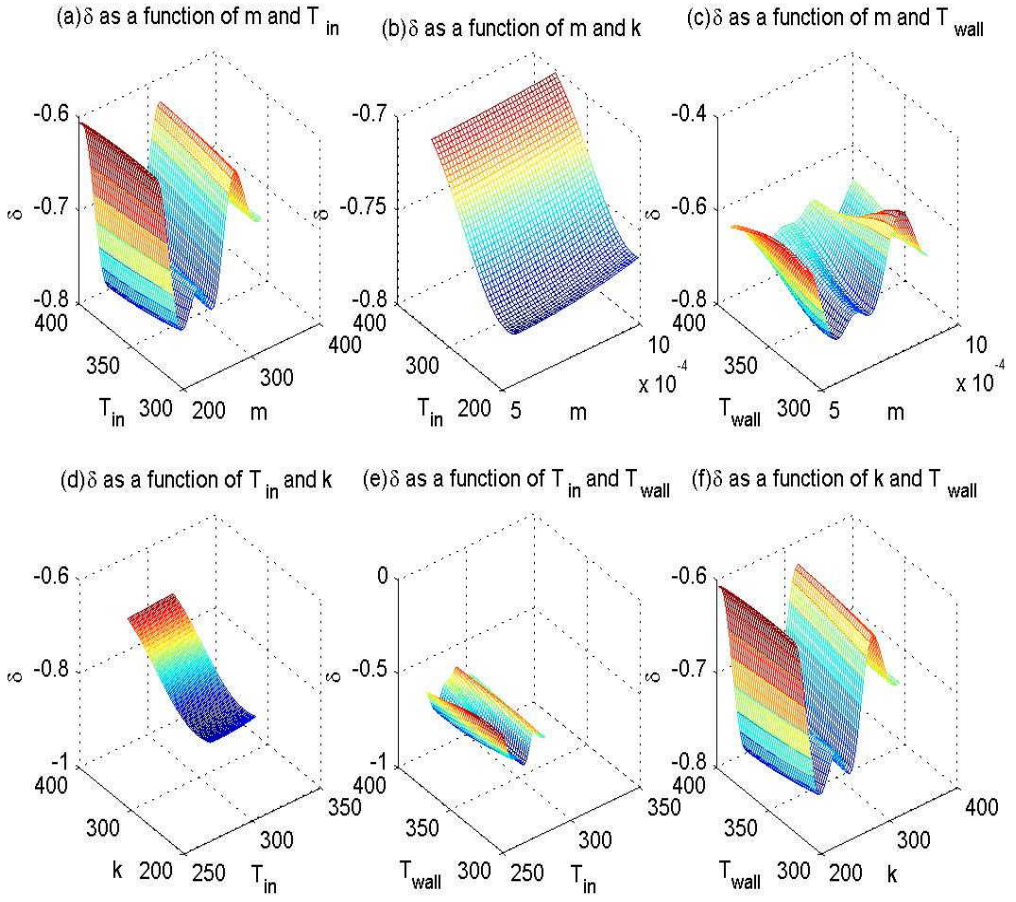

Figure 5: $\widehat{\delta}$ for different pairs of factors.

Table 4: Additional simulations for validation

Run	$\dot{m}(kg/s)$	$T_{in}(K)$	$k(W/mK)$	$T_{wall}(K)$	y_d	$\widehat{y}_d^1$	$\widehat{y}_d^2$	$\widehat{y}_a$	y_a
1	0.00050	293.15	362.73	393.15	25.82	23.85	24.09	26.96	27.24
2	0.00055	315	310	365	7.48	10.31	11.19	12.44	7.02
3	0.00056	277.01	354.98	374	19.77	26.02	24.99	26.38	25.53
4	0.00062	275	225	340	18.78	16.64	16.72	16.14	16.40
5	0.00068	313.28	259.12	350	4.55	6.44	9.04	7.32	10.23
6	0.00070	288.15	300	400	34.45	31.93	31.83	30.97	30.90
7	0.00078	292.73	267.84	369	21.97	23.70	22.49	22.01	20.92
8	0.00080	303.15	250	350	14.83	6.34	13.42	6.45	13.08
9	0.00085	270	325	385	32.85	37.88	37.32	31.34	31.14
10	0.00085	301.31	317.85	341	11.92	12.99	12.64	11.94	11.30
11	0.00091	248.87	206.74	398	47.05	51.77	47.04	39.63	36.56
12	0.00094	271.32	362.73	400	42.93	44.97	43.51	35.63	35.53
13	0.00095	280	270	330	17.41	16.82	17.54	13.51	13.54
14	0.00100	293.15	202.4	373.15	22.89	25.74	26.88	21.1	21.60

Finally, for a new input $\mathbf{x}^*$ we can create the final surrogate model:

$$\widehat{y}_d(\mathbf{x}^*) = \widehat{\rho}(\mathbf{x}^*)\widehat{y}_a(\mathbf{x}^*) + \widehat{\delta}(\mathbf{x}^*), \tag{16}$$

where $\widehat{\rho}(\mathbf{x}^*) = 1.130 + 0.090x_1^* - 0.032x_2^* + 0.004x_3^* - 0.012x_4^*$. $\widehat{y}_a(\cdot)$ is the BLUP of $y_a(\cdot)$ as described in (7) and $\widehat{\delta}(\cdot)$ is the BLUP of $\delta(\cdot)$ in (14).

1.3.4 Validation of the final surrogate model

In order to test and validate the method, 14 additional experiments are performed. These 14 runs are chosen at random in a space slightly larger than the original design space. For each experimental point, both detailed and approximate simulations are performed. Table 4 lists the factor levels for these experiments, the y_a and y_d values, the predicted obtained using (16) and the predicted obtained using (7) and the results presented in Section 3.3.

Root-mean-square-errors (RMSE) are computed to assess prediction performance. Here we present three different comparisons. The first is a comparison between predictions with the final surrogate model in (15) and detailed simulation data. The second is a comparison between predictions using the base surrogate model in (7) and the detailed simulation data,

and the third is a comparison between approximate and detailed simulation data.

$$\text{RMSE}_1 = \sqrt{\frac{\sum_{j=1}^{14}(\widehat{y}_d(\mathbf{x}_j) - y_d(\mathbf{x}_j))^2}{14}} = 3.795,$$

$$\text{RMSE}_2 = \sqrt{\frac{\sum_{j=1}^{14}(\widehat{y}_a(\mathbf{x}_j) - y_d(\mathbf{x}_j))^2}{14}} = 4.595,$$

and

$$\text{RMSE}_3 = \sqrt{\frac{\sum_{j=1}^{14}(y_a(\mathbf{x}_j) - y_d(\mathbf{x}_j))^2}{14}} = 4.431.$$

The proposed method provides a significant improvement in terms of prediction accuracy. The RMSE between and is 3.795, which is 14% smaller than the RMSE (4.430) between y_a and y_d, and 17% smaller than the RMSE (4.595) between $\widehat{y}_a$ and y_d given in Table 2. The difference between these RMSE's is statistically significant. Fig. 5 shows the nonlinear nature of the location adjustment in our procedure. The flexible scale-location adjustment is capable of refining the base surrogate model and obtaining a more accurate surrogate model. To get a sense of the relative size of the RMSE between $\widehat{y}_d$ and y_d (3.795), we calculated the mean of 14 DS runs (23.05) and their range (42.5). The RMSE is only 16% of the mean value and 8.9% of the range and thus is small for this case.

At this point, it is important to determine whether the improvement in prediction accuracy realized with the proposed method justifies the computational expense of building the final surrogate model. Whereas the RMSE of the base surrogate model, $\widehat{y}_a$, is 17% larger than the RMSE of the final surrogate model, $\widehat{y}_d$, the cost of building the base surrogate model is essentially negligible compared with the cost of building the final surrogate model, requiring minutes versus days of computing time to obtain the approximate and detailed experimental data reported in Table 2. Based on this comparison, a designer may conclude that the improvement in prediction accuracy is not sufficient to justify the increased computational expense of the proposed method. However, the comparison is misleading. In typical engineering applications, a designer would not rely exclusively on data from an un-calibrated approximate model. Because the accuracy of an approximate model is not known *a priori* in an engineering application, data from detailed simulations or physical

experiments are typically conducted throughout the region of interest for validation and calibration. If a number of detailed experiments are conducted anyway, the proposed method is both effective and efficient. By gathering only a few additional detailed simulation data points (beyond the number typically required for validating the approximate model) and by strategically choosing their locations, it is possible to assess the accuracy of an approximate model *and* reduce its predication error using the proposed method.

1.3.5 Maximize the total rate of steady state heat transfer

Note that one of the design objectives is to maximize the total heat transfer rate. The ranges of design variables are listed in Table 1. Table 5 contains the maximization results of $\widehat{y}_d(\mathbf{x})$ over the ranges. All the optimal values of four design variables are attained at the boundaries of the ranges. These results are not surprising. For this problem we know that as $\dot{m}$ increases, T_{in} decreases, k increases, or T_{wall} increases, the heat transfer rate increases. The maximum value of $\widehat{y}_d(\mathbf{x})$, 46.93, is larger than the values given in Tables 2 and 4, except for run 11 in Table 4. This outcome can be explained by noting that the design variable values in Table 5 that maximize heat transfer are not identical to any of the experiments in Tables 2 and 4.

Table 5: Maximizing $\widehat{y}_d(\mathbf{x})$ over the acceptable ranges

$\dot{m}(kg/s)$	$T_{in}(K)$	$k(W/mK)$	$T_{wall}(K)$	$\widehat{y}_d(\mathbf{x})$
0.001	270.00	360.0	400	46.93

1.4 Closure

In summary, we have presented an approach for building surrogate models based on data from both detailed and approximate simulations. From a design perspective, surrogate models reduce the computational cost of exploring large regions of the design space by replacing repeated detailed simulations. However, there can be a substantial computational cost involved in using data from detailed simulations to build surrogate models. Using the

approach presented in this chapter, it is possible to improve the accuracy of surrogate models obtained from approximate simulations by supplementing the data from the approximate simulations with relatively few data points from more computationally expensive detailed simulations. Thus, it is possible to explore a design space with improved or enhanced surrogate models that are more accurate that surrogate models based entirely on approximate simulations but less computationally expensive than surrogate models based exclusively on detailed simulations.

An advantage of our method is that surrogate models can be modified adaptively when new simulation results are available. Updating surrogate models requires negligible computational cost because it only involves refitting the model with both old and new data. Therefore it is relatively convenient to improve an existing surrogate model to a desired level of accuracy, if more accurate predictions are required.

The approach is broadly applicable to examples and phenomena from structural, electrical, financial, and other domains. The models usually correspond to different physics-based models or approximations of a problem (e.g., Euler Equations vs. Navier-Stokes, etc.). The primary assumptions are that multiple models or data sources are available and that one model or data source is generally more accurate than the other(s). The method is presented currently to integrate simulation models at only two levels, namely, detailed and approximate. Work is in progress to extend the method for more than two levels of models or data sources.

1.5 References

Akexandrov, N., Dennis, J. E. J., Lewis, R. M. and Torczon, V. (1998), "A trust region framework for managing the use of approximation models in optimization," *Structural Optimization*, **15**, 16-23.

Bakr, M. H., Bandler, J. W., Madsen, K. and Sndergaard, J. (2000), "Review of the space mapping approach to engineering optimization and modeling," *Optimization and Engineering*, **1**, 241-276.

Bandler, J., Cheng, Q., Dakroury, S., Mohamed, A., Bakr, M., Madsen, K. and Sndergaard, J. (2004), "Space mapping: the state of the art," *IEEE Transactions Microwave Theory Tech*, . **52**, 337-361.

Bettonvil, B. (1990), *Detection of Important Factors by Sequential Bifurcation*, Tilburg, The Netherlands: Tilburg University Press.

Bettonvil, B. and Kleijnen, J. J. C. (1996), "Searching for important factors in simulation models with many factors: sequential bifurcation," *European Journal of Operational Research*,**96**, 180-194.

Box, G. E. P. and Draper, N. R. (1969), *Evolutionary Operation: A Statistical Method for Process Management*, New York, NY: John Wiley and Sons.

Byrd, R. H., Lu, P., Nocedal, J. and Zhu, C. (1995), "A limited memory algorithm for bound constrained optimization," *SIAM Journal Scientific Computing*, **16**, 1190-1208.

Chen, W., Allen, J. K., Schrage, D. P. and Mistree, F. (1997), "Statistical experimentation methods for achieving affordable concurrent systems design," *AIAA Journal*, **35**, 893-900.

Chen, W., Allen, J. K., Tsui, K. L. and Mistree, F. (1996), "A procedure for robust design," *ASME Journal of Mechanical Design*, **118**, 478-485.

Cochran, J. K., Lee, K. J., McDowell, D. L. and Sanders, T. H. (2000), "Low density monolithic honeycombs by thermal chemical processing," in *Proceedings of the 4th Conference on Aerospace Materials, Processes, and Environmental Technology*, Huntsville, AL.

Cressie, N. (1988), "Geostatistics," *American Statistician*,**43**, 197-202.

Dempsey, B. M. (2002), *Thermal Properties of linear cellular alloys*, MS Thesis, G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology: Atlanta, GA.

Evans, A.G., Hutchinson, J. W., Fleck, N. A., Ashby, M. F. and Wadley, H. N. G. (2001), "The topological design of multifunctional cellular materials," *Progress in Materials Science*, **46**, 309-327.

Farhang-Mehr, A. and Azarm, S. (2003), "An information-theoretic entropy metric for assessing multiobjective optimization solution set quality," *ASME Journal of Mechanical Design*, **125**, 655-663.

FLUENT, 1998, Fluent, Inc., Release 5.5.14 (3d, segregated, laminar).

Gibson, L. J. and Ashby, M. F. (1997), *Cellular Solids: Structure and Properties*, Cambridge, UK: Cambridge University Press.

Handcock, M. S. and Stein, M. L. (1993), "A bayesian analysis of kriging," *Technometrics*, **35**, 403-410.

Handcock, M. S. and Wallis, J. R. (1994), "An approach to statistical spatial-temporal modeling of meteorological fields," *Journal of the American Statistical Association*, **89**, 386-378.

Hayes, A. M., Wang, A., Dempsey, B. M. and McDowell, D. L. (2001), "Mechanics of linear cellular alloys," *Mechanics and Materials*, **36**, 691-713.

Holcomb, D. R., Montgomery, D. C. and Carlyle, W. M. (2003), "Analysis of supersaturated designs," *Journal of Quality Technology*, **35**, 13-27.

Incropera, F. P. and DeWitt, D. P. (1996), *Fundamentals of Heat and Mass Transfer*, 3rd Edition, New York: John Wiley & Sons.

Koch, P. N., Simpson, T. W., Allen, J. K. and Mistree, F. (1999), "Statistical approximations for multidisciplinary optimization: the problem of size," *Journal of Aircraft: Special Multidisciplinary Design Optimization Issue*, **36**, 275-286.

Laslett, G. M. (1994), "Kriging and splines: an empirical comparison of their predictive performance in some applications," *Journal of the American Statistical Association*, **89**,391-400.

Lin, Y., Mistree, F. and Allen, J. K. (2004), "A sequential exploratory experimental design method: development of appropriate empirical models in design," in *ASME Design Engineering Technical Conferences - Advances in Design Automation Conference*, Salt Lake City, UT: ASME, Paper Number DETC2004/DAC-57527.

Matheron, G. (1963), "Principles of Geostatistics," *Economic Geology*, **58**, 1246-1266.

Michelena, N., Park, H. and Papalambros, P. (2002), "Convergence properties of analytical target cascading," *AIAA Journal*, **41**, 897-905.

Montgomery, D. C.(1997), *Design and Analysis of Experiments*, Fourth Edition, New York: John Wiley & Sons.

Myers, R. H. and Montgomery, D. C. (1995), *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, Wiley Series in Probability and Statistics. New York, NY: John Wiley and Sons.

Osio, I. C. and Amon, C. H. (1996), "An engineering design methodology with multistage bayesian surrogates and optimal sampling," *Research in Engineering Design*, **8**, 189-206.

Pacheco, J. E., Amon, C. H. and Finger, S. (2003),"Bayesian surrogates applied to conceptual stages of the engineering design process," *ASME Journal of Mechanical Design*, **125**, 664-672.

Qian, A., Seepersad, C. C., Joseph, V. R., Wu, C. F. J. and Allen, J. K. (2004), "Building surrogate models based on detailed and approximate simulations," in *ASME Design Engineering Technical Conferences - Advances in Design Automation Conference*, Salt Lake City, UT: ASME, Paper Number DETC2004/ 57486.

R Development Core Team (2004), "R: A language and environment for statistical computing," http://www.Rproject.org, date accessed (08/2004).

Rodriguez, J. F., Perez, V. M., Padmanabhan, D. and Renaud, J. E. (2001), "Sequential approximate optimization using variable fidelity response surface approximations,"

Structural and Multidisciplinary Optimization, **22**, 24-34.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer.

Seepersad, C. C., Dempsey, B. M., Allen, J. K., Mistree, F. and McDowell, D. L. (2004), "Design of multifunctional honeycomb materials," *AIAA Journal*, **42**, 1025-1033.

Simpson, T. W., Mauery, T. M., Korte, J. J. and Mistree, F. (2001), "Kriging models for global approximation in simulation-based multidisciplinary design optimization," *AIAA Journal*, **39**, 2233-2241.

Simpson, T. W., Peplinski, J. D., Koch, P. N. and Allen, J. K. (2001), "Metamodels for computer-based engineering design," *Engineering with Computers: An International Journal for Simulation-Based Engineering (Special issue in honor of Professor S.J. Fenves)*, **17**,129-150.

Toropov, V., van Keulen, F., Markine, V. and de Doer, H. (1996), "Refinements in the multi-point approximation method to reduce the effects of noisy structural response," in *6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Bellevue, WA: AIAA, **2**, 941-951.

Wang, G., Dong, Z. and Atchison, P. (2001), "Adaptive response surface method - a global optimization scheme for computation-intensive design problems," *Engineering Optimization*, **33**, 707-734.

Wang, G. (2003), "Adaptive response surface method using inherited latin hypercube designs," *ASME Journal of Mechanical Design*, **125**, 210-220.

Wang, G. G. and Simpson, T. W. (2004), "Fuzzy clustering based hierarchical metamodeling for design space reduction and optimization," Engineering Optimization, **36**, 313-335.

Watson, G. S. (1961), "A study of the group screening method," *Technometrics*, **3**, 371-388.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J. and Morris, M. D. (1992), "Screening, predicting and computer experiments," *Technometrics*, **34**, 15-25.

Wu, C. F. J. (1993), "Construction of supersaturated designs through partially aliased interactions," *Biometrika*, **80**, 661-669.

Wu, C. F. J. and Hamada, M. (2000), *Experiments, Planning, Analysis and Parameter Design Optimization*, New York: John Wiley and Sons.

Wujek, B. A. and Renaud, J. E. (1998a), "New adaptive move-limit management strategy for approximate optimization: part 1," *AIAA Journal*, **35**, 1911-1921.

Wujek, B. A. and Renaud, J. E. (1998b), "New adaptive move-limit management strategy for approximate optimization: part 2," *AIAA Journal*, **36**, 1922-1934.

CHAPTER II

BAYESIAN HIERARCHICAL MODELING FOR INTEGRATING LOW-ACCURACY AND HIGH-ACCURACY EXPERIMENTS

2.1 Introduction

A challenging and fascinating problem in design and analysis of experiments is the synthesis of data from different types of experiments. With the advances in computing and experimentation, scientists can quickly access data from different sources. Complex mathematical models, implemented in large computer codes, are widely used to study real systems. Doing the corresponding physical experimentation would be more time-consuming and costly. For example, each physical run of the fluidized bed process (to be discussed in Section 2.4) can take days or even weeks to finish while running the associated computer code only takes minutes per run. Furthermore, a large computer program can often be run at different levels of sophistication with vastly varying computational times. Consider, for example, two codes that simulate linear cellular alloys for electronic cooling systems (to be discussed in Section 2.3). One code uses finite element analysis while the other is based on finite difference method. The two codes differ in the numerical method and the resolution of the grid, resulting in an accurate but slow version and a crude but fast approximation. In this chapter, we consider a generic situation in which two sources (or experiments) are available and one source is generally more accurate than the other but also more expensive to run. The two experiments considered are called *low-accuracy experiment* and *high-accuracy experiment* and referred to as LE and HE respectively. The pair can be physical vs. computer experiments or detailed vs. approximate computer experiments. Experimenters are often faced with the problem of how to integrate these multiple data sources efficiently. There is a recent surge of interests in this problem. For example, Kennedy and O'Hagan

(2000) and Qian et al. (2006) consider integrating data from detailed and approximate computer experiments, and Reese et al. (2004) deals with integrating physical and computer experiments.

The purpose of this chapter is to introduce Bayesian hierarchical Gaussian process (BHGP) models to integrate multiple data sources. The heterogeneity among different sources is accounted for by performing flexible location and scale adjustments. The chapter is organized as follows. The BHGP models are developed in Section 2.2. Sections 2.3 and 2.4 illustrate the method with two real examples: one with detailed and approximate computer experiments and the other with physical and computer experiments. Concluding remarks and extensions are given in Section 2.5. Some computational details are included in the Appendix.

2.2 Bayesian hierarchical Gaussian process models

Standard approaches to the synthesis of low-accuracy and high-accuracy experiments analyze data from each type separately. By borrowing strength across multiple sources, an integrated analysis can produce better results. Qian et al. (2006) introduces a two-step approach to integrate results from detailed and approximate computer experiments. It starts with fitting a Gaussian process model for the approximate experiment data. In the second step, the fitted model is adjusted by incorporating the more accurate data from the detailed experiment. The present work can be viewed as an extension of theirs. The essential differences between the two approaches are two-fold. First, new hierarchical Gaussian process models are introduced to carry out location and scale adjustments more flexibly. Second, the present approach adopts the Bayesian formulation and can absorb uncertainty in the model parameters in the prediction. Reese et al. (2004) proposes another hierarchical method by using linear models to integrate data from physical and computer experiments. Although this approach has advantages such as the ease of computation and interpretation, the linear models cannot serve as interpolators whereas the Gaussian process models have this feature when modeling deterministic computer experiments. Also the linear models are not as flexible as the Gaussian process models in representing complex nonlinear relationships.

Suppose that the LE and HE involve the same k factors $\mathbf{x} = (x_1, \ldots, x_k)$. Denote by $D_l = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ the design set for the LE with n runs, and $\mathbf{y}_l = (y_l(\mathbf{x}_1), \ldots, y_l(\mathbf{x}_n))^t$ the corresponding LE data. Because an HE run requires more computational effort to generate than an LE run, usually there are fewer HE runs available. Without loss of generality, we assume that D_h the design set of the HE consists of the first n_1 ($n_1 < n$) runs of D_l. The outputs from the HE are denoted by $\mathbf{y}_h = (y_h(\mathbf{x}_1), \ldots, y_h(\mathbf{x}_{n_1}))^t$. Note that the subscripts h and l denote "high" and "low". The main goal of the proposed method is to predict y_h at some *untried* points (i.e., these points outside D_h). Central to the method are Bayesian hierarchical Gaussian process (BHGP) models, which consist of the following two parts:

1. Fit a smooth model for the LE data.

2. Fit a flexible model to "link" the LE and the HE data.

Detailed descriptions of these two models are given in Sections 2.2.2 and 2.2.3.

2.2.1 Bayesian Gaussian process model

In this section, we present the basics of Bayesian analysis of a Gaussian process model as the basis for later development. A good reference for Gaussian process models is Santner, Williams and Notz (2003) (hereafter abbreviated as SWN 2003). For simplicity, throughout the chapter a Gaussian process with mean μ and variance σ^2 is denoted by $GP(\mu, \sigma^2, \boldsymbol{\phi})$, where $\boldsymbol{\phi}$ will be defined below. Suppose $y(\mathbf{x})$ is a real-valued stationary Gaussian process on the real line with mean $E\{y(\mathbf{x})\} = \mathbf{f}(\mathbf{x})^t\boldsymbol{\beta}$, where $\mathbf{x} = (x_1, \ldots, x_k)$, $\mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}), \ldots, f_q(\mathbf{x})\}^t$ is a known vector-valued function and $\boldsymbol{\beta}$ is a vector of unknown regression coefficients. Furthermore, the covariance function is represented by $\mathrm{cov}(y(\mathbf{x}_1), y(\mathbf{x}_2)) = \sigma^2 K_{\boldsymbol{\phi}}(\mathbf{x}_1, \mathbf{x}_2)$, where σ^2 is the variance and $K_{\boldsymbol{\phi}}(\cdot, \cdot)$ is the correlation function and depends on the unknown *correlation parameters* $\boldsymbol{\phi}$. Although the proposed method works for general correlation functions, we specifically apply it to the following Gaussian correlation function (SWN 2003)

$$K_{\boldsymbol{\phi}}(\mathbf{x}_1, \mathbf{x}_2) = \prod_{i=1}^{k} \exp\{-\phi_{i1}(x_{1i} - x_{2i})^2\}. \tag{17}$$

Here, the scale correlation parameters ϕ_{i1} are positive. The power correlation parameters are fixed at 2 (SWN 2003), thus reducing the complication of estimating the correlation parameters. In addition, the sample path of the Gaussian process is infinitely differentiable, which is a reasonable assumption for many applications including the examples in Sections 2.3 and 2.4. As a result, this correlation is often adopted in the computer experiments literature (Welch et al. 1992, SWN 2003). In general, we observe $\mathbf{y} = \{y(\mathbf{x}_1), \ldots, y(\mathbf{x}_n)\}$ and are interested in predicting y at a given point $\mathbf{x}_0$.

The priors for the model parameters $\boldsymbol{\beta}, \sigma^2, \boldsymbol{\phi}$ take the following structure

$$p(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\phi}) = p(\boldsymbol{\beta}, \sigma^2)p(\boldsymbol{\phi}) = p(\boldsymbol{\beta}|\sigma^2)p(\sigma^2)p(\boldsymbol{\phi}). \tag{18}$$

The choice of priors requires some care. As pointed out in Berger et al. (2001), improper priors chosen for $\boldsymbol{\phi}$ may lead to improper posteriors as well. To avoid this problem, proper priors are adopted as follows:

$$\begin{aligned} p(\sigma^2) &\sim IG(\alpha, \gamma), \\ p(\boldsymbol{\beta}|\sigma^2) &\sim N(\mathbf{u}, v\mathbf{I}_{q\times q}\sigma^2), \end{aligned}$$

and

$$\phi_i \sim G(a, b), \text{ for } i = 1, \ldots, k, \tag{19}$$

where $IG(\alpha, \gamma)$ denotes the inverse gamma distribution with density function

$$p(z) \sim z^{-(\alpha+1)} \exp\{-\frac{\gamma}{z}\}, \ z > 0,$$

$G(a, b)$ is the gamma distribution with density function

$$p(z) = \frac{b^a}{\Gamma(a)} z^{a-1} e^{-bz}, \ z > 0,$$

$N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate normal distribution with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$ and $\mathbf{I}_{q\times q}$ is the $q \times q$ identity matrix.

It can be shown (SWN 2003) that the conditional distribution of y at $\mathbf{x}_0$, giving the observed $\mathbf{y}$, is the non-central t distribution

$$T_1(n + \nu_0, \mu_1, \sigma_1^2), \tag{20}$$

where

$$\begin{aligned}
\mu_1 &= \mathbf{f}_0^t\boldsymbol{\mu}_{\boldsymbol{\beta}|n} + \mathbf{r}_0^t\mathbf{R}^{-1}(\mathbf{y} - \mathbf{F}\boldsymbol{\mu}_{\boldsymbol{\beta}|n}),\\
\boldsymbol{\mu}_{\boldsymbol{\beta}|n} &= (\mathbf{F}^t\mathbf{R}^{-1}\mathbf{F} + v^{-1}\mathbf{I}_{q\times q})^{-1}(\mathbf{F}^t\mathbf{R}^{-1}\mathbf{y} + \mathbf{u}v^{-1}\mathbf{I}_{q\times q}),\\
\widehat{\boldsymbol{\beta}} &= (\mathbf{F}^t\mathbf{R}^{-1}\mathbf{F})^{-1}(\mathbf{F}^t\mathbf{R}^{-1}\mathbf{y}),\\
\sigma_1^2 &= \frac{Q_1^2}{\nu_1}\{1 - (\mathbf{f}_0^t, \mathbf{r}_0^t)\begin{bmatrix} -v^{-1}\mathbf{I}_{q\times q} & \mathbf{F}^t \\ \mathbf{F}^t & \mathbf{R}\end{bmatrix}^{-1}\begin{pmatrix}\mathbf{f}_0\\ \mathbf{r}_0\end{pmatrix}\},\\
Q_1^2 &= c_0 + \mathbf{y}^t[\mathbf{R}^{-1} - \mathbf{R}^{-1}\mathbf{F}(\mathbf{F}^t\mathbf{R}^{-1}\mathbf{F})^{-1}\mathbf{F}^t\mathbf{R}^{-1}]\mathbf{y}\\
&\quad +(\mathbf{u} - \widehat{\boldsymbol{\beta}})^t[v\mathbf{I}_{q\times q} + (\mathbf{F}^t\mathbf{R}^{-1}\mathbf{F})^{-1}]^{-1}(\mathbf{u} - \widehat{\boldsymbol{\beta}}),
\end{aligned}$$

$\nu_0 = 2a$, $\nu_1 = n + 2a$, $c_0 = \sqrt{\frac{b}{a}}$, $\mathbf{f}_0 = \mathbf{f}(\mathbf{x}_0)$, $\mathbf{r}_0 = (R(\mathbf{x}_0, \mathbf{x}_1), \ldots, R(\mathbf{x}_0, \mathbf{x}_n))^t$, $\mathbf{R}$ is the correlation matrix with entry $R(\mathbf{x}_i, \mathbf{x}_j)$ for $i, j = 1, \ldots, n$, $\mathbf{F} = (\mathbf{f}(\mathbf{x}_1)^t, \ldots, \mathbf{f}(\mathbf{x}_n)^t)^t$ is the regressor matrix and the density of $T_1(n + \nu_0, \mu_1, \sigma_1^2)$ is

$$p(z) = \frac{\Gamma((n + \nu_0 + 1)/2)}{\sigma_1[(n + \nu_0)\pi]^{1/2}\Gamma((n + \nu_0)/2)}[1 + \frac{1}{n + \nu_0}\frac{(z - \mu_1)^2}{\sigma_1^2}]^{-(n+\nu_0+1)/2}.$$

2.2.2 Low-accuracy experiment data

We assume that $y_l(\mathbf{x}_i)$ can be described by

$$y_l(\mathbf{x}_i) = f_l^t(\mathbf{x}_i)\boldsymbol{\beta}_l + \epsilon_l(\mathbf{x}_i),\ i = 1, \ldots, n, \tag{21}$$

where $f_l(\mathbf{x}_i) = (1, x_{i1}, \ldots, x_{ik})^t$, $\boldsymbol{\beta}_l = (\beta_{l0}, \beta_{l1}, \ldots, \beta_{lk})^t$ and $\epsilon_l(\cdot)$ is assumed to be $GP(0, \sigma_l^2, \boldsymbol{\phi}_l)$. Here, the mean function includes linear effects, because in many circumstances (including the two examples given later) it is reasonable to assume the factors considered in the experiments have linear effects on the outputs. In addition, inclusion of "weak" main effects in the mean of a Gaussian process can bring additional numerical benefits for estimating the correlation parameters. Suppose, instead, the mean in (21) includes only a constant μ, the likelihood of $\mathbf{y}_l$ will be

$$\propto \frac{1}{|\boldsymbol{\Sigma}|^{\frac{1}{2}}}\exp\{-(\mathbf{y}_l - \mu\mathbf{1}_n)^t\boldsymbol{\Sigma}^{-1}(\mathbf{y}_l - \mu\mathbf{1}_n)\}, \tag{22}$$

where the covariance matrix $\boldsymbol{\Sigma}$ depends on the unknown correlation parameters $\boldsymbol{\phi}_l$ and $\mathbf{1}_n$ represents the n-unity column vector. For a large number of observations, (22) can

be extremely small regardless of the values of $\boldsymbol{\phi}_l$. As a result, $\boldsymbol{\phi}_l$ cannot be accurately estimated. The inclusion of some weak main effects in the mean can partially mitigate this problem by "dampening" the Mahalabonis distance between $\mu\mathbf{1}_n$ and $\mathbf{y}_l$.

If LE were the only source considered, then this model would be fitted using the Bayesian Gaussian process model discussed in Section 2.2.1. Because the LE data are not very accurate, the HE data need to be incorporated to improve the quality of the fitted model.

2.2.3 High-accuracy experiment data

Because LE and HE are conducted by using different mechanisms (physical or computational) or distinct numerical methods with different mesh sizes, orders of elements, or other important aspects, their outputs can be different. In general we can classify the relationship between y_l and y_h into three broad categories:

1. LE produces outputs almost as good as HE;
2. No similarities can be found (or defined) between y_l and y_h;
3. LE and HE give different outputs but share similar trends.

For category (1), the differences between y_l and y_h can be largely ignored, and using a single model for both data sources will suffice. Furthermore, the HE runs can be replaced by the LE runs, resulting in huge computational savings. However, these scenarios do not occur often in practice. The second category consists of cases, where LE and HE are "oranges" and "apples". No sensible methods can be used to adjust the LE results and to integrate the LE and HE data. In such situations, the experimenters need to scrutinize the underlying assumptions or the set-ups of the LE and try to make improvements by better understanding the differences between LE and HE. Most problems in practice fall in category (3), which is the focus of the chapter.

In order to "link" the HE data with the LE data, we consider the following *adjustment model*

$$y_h(\mathbf{x}_i) = \rho(\mathbf{x}_i)y_l(\mathbf{x}_i) + \delta(\mathbf{x}_i) + \epsilon(\mathbf{x}_i), i = 1, \ldots, n_1. \tag{23}$$

Here $\rho(\cdot)$, assumed to be $GP(\rho_0, \sigma_\rho^2, \boldsymbol{\phi}_\rho)$, accounts from *scale change* from LE to HE. We assume $\delta(\cdot)$ to be $GP(\delta_0, \sigma_\delta^2, \boldsymbol{\phi}_\delta)$ and represent *location adjustment*. The measurement error $\epsilon(\cdot)$ is assumed to be $N(0, \sigma_\epsilon^2)$. Furthermore, $y_l(\cdot)$, $\delta(\cdot)$, $\rho(\cdot)$ and $\epsilon(\cdot)$ are assumed to be independent.

The unknown parameters $\boldsymbol{\theta}$ involved in models (21) and (23) can be collected into three groups: mean parameters $\boldsymbol{\theta}_1 = (\boldsymbol{\beta}_l, \rho_0, \delta_0)$, variance parameters $\boldsymbol{\theta}_2 = (\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2, \sigma_\epsilon^2)$ and correlation parameters $\boldsymbol{\theta}_3 = (\boldsymbol{\phi}_l, \boldsymbol{\phi}_\rho, \boldsymbol{\phi}_\delta)$. The description of the hierarchical models in (21) and (23) is complete with the specification of priors. It is similar to that of the Bayesian Gaussian process model in Section 2.2.1. The chosen priors take the following form

$$p(\boldsymbol{\theta}) = p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3) = p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_3), \tag{24}$$

where

$$\begin{aligned}
p(\sigma_l^2) &\sim IG(\alpha_l, \gamma_l), \\
p(\sigma_\rho^2) &\sim IG(\alpha_\rho, \gamma_\rho), \\
p(\sigma_\delta^2) &\sim IG(\alpha_\delta, \gamma_\delta), \\
p(\sigma_\epsilon^2) &\sim IG(\alpha_\epsilon, \gamma_\epsilon), \\
p(\boldsymbol{\beta}_l|\sigma_l^2) &\sim N(\mathbf{u}_l, v_l\mathbf{I}_{(k+1)\times(k+1)}\sigma_l^2), \\
p(\rho_0|\sigma_\rho^2) &\sim N(u_\rho, v_\rho\sigma_\rho^2), \\
p(\delta_0|\sigma_\delta^2) &\sim N(u_\delta, v_\delta\sigma_\delta^2), \\
\phi_{l_i} &\sim G(a_l, b_l), \phi_{\rho_i} \sim G(a_\rho, b_\rho), \phi_{\delta_i} \sim G(a_\delta, b_\delta), \text{ for } i = 1, \ldots, k.
\end{aligned} \tag{25}$$

2.2.4 Bayesian prediction

Recall that we are interested in predicting y_h at an untried point $\mathbf{x}_0$. For the ease of methodological development, we first assume that the untried point $\mathbf{x}_0$ belongs to D_l but is not a point in D_h (otherwise $y_h(\mathbf{x}_0)$ is readily available). This assumption shall be relaxed later. Assume for the moment that the value of $\boldsymbol{\theta}_3$ is given. In Section 2.2.5, we shall discuss the fitting of $\boldsymbol{\theta}_3$. The prediction is based on the *Bayesian predictive density function*

$$p(y_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l) = \int_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2} p(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_3)d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2. \tag{26}$$

In this approach, uncertainty in the model parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ is naturally absorbed in the prediction.

The integration of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ in (26) needs to be done numerically. A Markov Chain Monte Carlo (MCMC) (Liu 2001) algorithm to approximate $p(y_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ is given as follows:

1. Generate $(\boldsymbol{\theta}_1^{(1)}, \boldsymbol{\theta}_2^{(1)}), \ldots, (\boldsymbol{\theta}_1^{(m)}, \boldsymbol{\theta}_2^{(m)})$ from $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_3)$.

2. Approximate $p(y_h(\mathbf{x}_0)|\mathbf{y}_h, \mathbf{y}_l)$ by

$$\widehat{p}_m(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_3) = \frac{1}{m}\sum_{i=1}^{m} p(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \boldsymbol{\theta}_3). \tag{27}$$

For the ease of posterior sampling in step 1, we introduce new parameters $\tau_1 = \frac{\sigma_\delta^2}{\sigma_\rho^2}$ and $\tau_2 = \frac{\sigma_\epsilon^2}{\sigma_\rho^2}$ and use $(\sigma_\rho^2, \tau_1, \tau_2)$ instead of $(\sigma_\rho^2, \sigma_\delta^2, \sigma_\epsilon^2)$ in the model. With some abuse of notation, we shall still use $\boldsymbol{\theta}_2$ to denote $(\sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2)$. From (25), the prior for σ_ρ^2, τ_1 and τ_2 is easily shown to be

$$\begin{aligned} p(\sigma_\rho^2, \tau_1, \tau_2) &= \frac{\gamma_\rho^{\alpha_\rho}}{\Gamma(\alpha_\rho)}(\sigma_\rho^2)^{-(\alpha_\rho+1)}\exp\{-\gamma_\rho/\sigma_\rho^2\} \\ &\quad \cdot\frac{(\gamma_\delta)^{\alpha_\delta}}{\Gamma(\alpha_\delta)}(\sigma_\rho^2\tau_1)^{-(\alpha_\delta+1)}\exp\{-\gamma_\delta/(\sigma_\rho^2\tau_1)\} \\ &\quad \cdot\frac{(\gamma_\epsilon)^{\alpha_\epsilon}}{\Gamma(\alpha_\epsilon)}(\sigma_\rho^2\tau_2)^{-(\alpha_\epsilon+1)}\exp\{-\gamma_\epsilon/(\sigma_\rho^2\tau_2)\}(\sigma_\rho^2)^2. \end{aligned} \tag{28}$$

The key for deriving the full conditional distributions of $(\boldsymbol{\beta}_l, \delta_0, \rho_0, \sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2)$ that is used in the MCMC is to note that by conditioning on $\boldsymbol{\theta}_3$, these parameters can be viewed as coming from some general linear models. Given $\boldsymbol{\theta}_3$, the full conditional distributions for

$(\boldsymbol{\beta}_l, \delta_0, \rho_0, \sigma_l^2, \sigma_\rho^2, \tau_1, \tau_2)$ can be shown to be

$$
\begin{aligned}
p(\boldsymbol{\beta}_l|\mathbf{y}_l, \mathbf{y}_h, \overline{\boldsymbol{\beta}_l}) &\sim N((\frac{\mathbf{u}_l}{v_l\sigma_l^2}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{y}_l^t\mathbf{R}_l^{-1}\mathbf{F}_l)(\frac{1}{v_l\sigma_l^2}\mathbf{I}_{(k+1)\times(k+1)} + \frac{\mathbf{F}_l^t\mathbf{R}_l^{-1}\mathbf{F}_l}{\sigma_l^2})^{-1}, \\
&\quad (\frac{1}{v_l}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^t\mathbf{R}_l^{-1}\mathbf{F}_l)^{-1}\sigma_l^2), \\
p(\rho_0|\mathbf{y}_l, \mathbf{y}_h, \overline{\rho_0}) &\sim N\left(\frac{\frac{u_\rho}{v_\rho} + \mathbf{y}_{l_1}^t\mathbf{M}^{-1}(\mathbf{y}_h - \delta_0\mathbf{1}_{n_1})}{\frac{1}{v_\rho} + \mathbf{y}_{l_1}^t\mathbf{M}^{-1}\mathbf{y}_{l_1}}, \frac{\sigma_\rho^2}{\frac{1}{v_\rho} + \mathbf{y}_{l_1}^t\mathbf{M}^{-1}\mathbf{y}_{l_1}}\right), \\
p(\delta_0|\mathbf{y}_l, \mathbf{y}_h, \overline{\delta_0}) &\sim N\left(\frac{\frac{u_\delta}{v_\delta\tau_1} + \mathbf{1}_{n_1}^t\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1})}{\frac{1}{v_\delta\tau_1} + \mathbf{1}_{n_1}^t\mathbf{M}^{-1}\mathbf{1}_{n_1}}, \frac{\sigma_\delta^2}{\frac{1}{v_\delta\tau_1} + \mathbf{1}_{n_1}^t\mathbf{M}^{-1}\mathbf{1}_{n_1}}\right), \\
p(\sigma_l^2|\mathbf{y}_l, \mathbf{y}_h, \overline{\sigma_l^2}) &\sim IG(\frac{n}{2} + \frac{k+1}{2} + \alpha_l, \\
&\quad \frac{1}{2}\frac{(\boldsymbol{\beta}_l - \mathbf{u}_l)^t(\boldsymbol{\beta}_l - \mathbf{u}_l)}{v_l} + \frac{1}{2}(\mathbf{y}_l - \mathbf{F}_l\boldsymbol{\beta}_l)^t\mathbf{R}_l^{-1}(\mathbf{y}_l - \mathbf{F}_{\boldsymbol{l}}\boldsymbol{\beta}_l) + \gamma_l), \\
p(\sigma_\rho^2|\mathbf{y}_l, \mathbf{y}_h, \overline{\sigma_\rho^2}) &\sim IG(\frac{n_1}{2} + \frac{1}{2} + \alpha_\rho + \alpha_\delta + \alpha_\epsilon, \frac{(\rho_0 - u_\rho)^2}{2v_\rho} + \gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{2} \\
&\quad + \frac{(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^t\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})}{2}), \\
p(\tau_1, \tau_2|\mathbf{y}_l, \mathbf{y}_h, \overline{\tau_1, \tau_2}) &\propto \frac{1}{\tau_1^{\alpha_\delta + \frac{3}{2}}}\frac{1}{\tau_2^{\alpha_{\epsilon+1}}}\exp\{-\frac{1}{\tau_1}(\frac{\gamma_\delta}{\sigma_\rho^2} + \frac{(\delta_0 - u_\delta)^2}{2v_\delta\sigma_\rho^2}) - \frac{\gamma_\epsilon}{\tau_2\sigma_\rho^2}\}\frac{1}{|\mathbf{M}|^{\frac{1}{2}}} \\
&\quad \cdot\exp\{-\frac{(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})^t\mathbf{M}^{-1}(\mathbf{y}_h - \rho_0\mathbf{y}_{l_1} - \delta_0\mathbf{1}_{n_1})}{2\sigma_\rho^2}\}, \qquad (29)
\end{aligned}
$$

where $\overline{\omega}$ represents all the components of $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$ except for ω, $\mathbf{M} = \mathbf{W}_\rho + \tau_1\mathbf{R}_\delta + \tau_2\mathbf{I}_{n_1\times n_1}$ and depends on $\boldsymbol{\phi}_\rho$, $\boldsymbol{\phi}_\delta$, τ_1 and τ_2, $\mathbf{y}_{l_1} = (y_l(\mathbf{x}_1), \ldots, y_l(\mathbf{x}_{n_1}))^t$,$\mathbf{W}_\rho = \mathbf{A}_1\mathbf{R}_\rho\mathbf{A}_1$, $\mathbf{A}_1 = \text{diag}\{y_l(\mathbf{x}_1), \ldots, y_l(\mathbf{x}_{n_1})\}$ and $\mathbf{R}_\rho$ and $\mathbf{R}_\delta$ are the correlation matrices of $\boldsymbol{\rho} = (\rho(\mathbf{x}_1), \ldots, \rho(\mathbf{x}_{n_1}))^t$ and $\boldsymbol{\delta} = (\delta(\mathbf{x}_1), \ldots, \delta(\mathbf{x}_{n_1}))^t$ respectively.

The Gibbs sampler cannot be directly applied here, because the full conditional distribution for τ_1 and τ_2 in (29) is non-standard. To circumvent this problem, the Metropolis-within-Gibbs algorithm (Liu 2001) is used, where a Metropolis draw is added to sample τ_1 and τ_2 within the usual Gibbs loop.

The second step of the approximation in (27) is straightforward. The analytic form of $p(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$ can be obtained by rewriting it as

$$\frac{p(y_h(\mathbf{x}_0), \mathbf{y}_h|\mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)}{p(\mathbf{y}_h|\mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)}. \qquad (30)$$

From the assumption that $\rho(\cdot)$, $\delta(\cdot)$ and $\epsilon(\cdot)$ are independent of $\mathbf{y}_l$ in (23), the distributions

of the numerator and the denominator in (30) are as follows:

$$\begin{aligned} p(y_h(\mathbf{x}_0), \mathbf{y}_h|\mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) &\sim N(\rho_0 \mathbf{y}_{l_1}^* + \delta_0 \mathbf{1}_{n_1+1}, \sigma_\rho^2 \mathbf{W}_\rho^* + \sigma_\delta^2 \mathbf{R}_\delta^* + \sigma_\epsilon^2 \mathbf{I}_{(n_1+1)\times(n_1+1)}), \\ p(\mathbf{y}_h|\mathbf{y}_l, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) &\sim N(\rho_0 \mathbf{y}_{l_1} + \delta_0 \mathbf{1}_{n_1}, \sigma_\rho^2 \mathbf{W}_\rho + \sigma_\delta^2 \mathbf{R}_\delta + \sigma_\epsilon^2 \mathbf{I}_{n_1\times n_1}), \end{aligned} \tag{31}$$

where

$$\begin{aligned} \mathbf{y}_{l_1}^* &= (y_l(\mathbf{x}_0), y_l(\mathbf{x}_1), \ldots, y_l(\mathbf{x}_{n_1}))^t, \\ \mathbf{W}_\rho^* &= \mathbf{A}_1^* \mathbf{R}_\rho^* \mathbf{A}_1^*, \\ \mathbf{A}_1^* &= \mathrm{diag}\{y_l(\mathbf{x}_0), y_l(\mathbf{x}_1), \ldots, y_l(\mathbf{x}_{n_1})\}, \end{aligned}$$

and $\mathbf{R}_\rho^*$ and $\mathbf{R}_\delta^*$ are the correlation matrices of $\boldsymbol{\rho}^* = (\rho(\mathbf{x}_0), \rho(\mathbf{x}_1), \ldots, \rho(\mathbf{x}_{n_1}))^t$ and $\boldsymbol{\delta}^* = (\delta(\mathbf{x}_0), \delta(\mathbf{x}_1), \ldots, \delta(\mathbf{x}_{n_1}))^t$ respectively.

Once the predictive density has been computed, we can use

$$\widehat{y}_h(\mathbf{x}_0) = E(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h) \tag{32}$$

as the *predictor* for $y_h(\mathbf{x}_0)$ and $\mathrm{Var}(y_h(\mathbf{x}_0)|\mathbf{y}_l, \mathbf{y}_h)$ as the *prediction variance*.

Next, we relax the assumption $\mathbf{x}_0 \in D_l/D_h$ and consider the prediction when $\mathbf{x}_0$ does not belong to D_l. The additional difficulty is that the value of $y_l(\mathbf{x}_0)$ is not observed. In the Bayesian framework, we can fit the Bayesian Gaussian process model as described in Section 2.1 and impute $y_l(\mathbf{x}_0)$ by $\widehat{y}_l = E(y_l(\mathbf{x}_0)|\mathbf{y}_l)$ (the mean of a non-central t distribution). Then we can add $\widehat{y}_l$ to the set of $\mathbf{y}_l$ so that $\mathbf{x}_0$ belongs to the expanded set $D_l \cup \{\mathbf{x}_0\}$.

2.2.5 Estimation of correlation parameters

Next, we discuss the fitting of the correlation parameters $\boldsymbol{\theta}_3$.

Note that

$$p(\boldsymbol{\theta}_3|\mathbf{y}_h, \mathbf{y}_l) \propto p(\boldsymbol{\theta}_3) \int_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2} p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) p(\mathbf{y}_l, \mathbf{y}_h|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2, \tag{33}$$

which can be shown (see Appendix) to be proportional to

$$\begin{aligned} L_1 = p(\boldsymbol{\theta}_3) \int_{\tau_1, \tau_2} \tau_1^{-(\alpha_\delta + \frac{3}{2})} \tau_2^{-(\alpha_\epsilon + 1)} |\mathbf{a}_1|^{-\frac{1}{2}} |\mathbf{R}_l|^{-\frac{1}{2}} |\mathbf{M}|^{-\frac{1}{2}} (a_2 a_3)^{-\frac{1}{2}} (\gamma_l + \frac{4c_1 - \mathbf{b}_1^t \mathbf{a}_1^{-1} \mathbf{b}_1}{8})^{-(\alpha_l + \frac{n}{2})} \\ \cdot (\gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{4a_3 c_3 - b_3^2}{8a_3})^{-(\alpha_\rho + \alpha_\delta + \alpha_\epsilon + \frac{n_1}{2})} d\tau_1 d\tau_2. \end{aligned} \tag{34}$$

The posterior mode estimator $\widehat{\boldsymbol{\theta}}_3$ is given by an optimal solution to the optimization problem

$$\max_{\boldsymbol{\phi}_l,\boldsymbol{\phi}_\rho,\boldsymbol{\phi}_\delta} L_1, \tag{35}$$

which is equivalent to solving the following two separable problems

$$\max_{\boldsymbol{\phi}_l} p(\boldsymbol{\phi}_l)|\mathbf{R}_l|^{-\frac{1}{2}}|\mathbf{a}_1|^{-\frac{1}{2}}(\gamma_l + \frac{4c_1 - \mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8})^{-(\alpha_l+\frac{n}{2})} \tag{36}$$

and

$$\max_{\boldsymbol{\phi}_\rho,\boldsymbol{\phi}_\delta} \int_{\tau_1,\tau_2} p(\boldsymbol{\phi}_\rho)p(\boldsymbol{\phi}_\delta)\tau_1^{-(\alpha_\delta+\frac{3}{2})}\tau_2^{-(\alpha_\epsilon+1)}|\mathbf{M}|^{-\frac{1}{2}}(a_2a_3)^{-\frac{1}{2}}$$
$$\cdot(\gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{4a_3c_3 - b_3^2}{8a_3})^{-(\alpha_\rho+\alpha_\delta+\alpha_\epsilon+\frac{n_1}{2})}d\tau_1 d\tau_2. \tag{37}$$

The optimization problem in (36) can be solved by using standard non-linear optimization algorithms like the quasi-Newton method. Solving the optimization problem in (37) is more elaborate because its objective function involves integration. The problem in (37) can be recast as

$$\max_{\boldsymbol{\phi}_\rho,\boldsymbol{\phi}_\delta}\{L_2 = E_{\tau_1,\tau_2}f(\tau_1,\tau_2)\}, \tag{38}$$

where

$$f(\tau_1,\tau_2) = \frac{p(\boldsymbol{\phi}_\rho)p(\boldsymbol{\phi}_\delta)\exp(\frac{2}{\tau_1})\exp(\frac{2}{\tau_2})}{|\mathbf{M}|^{\frac{1}{2}}(a_2a_3)^{\frac{1}{2}}(\gamma_\rho + \frac{\gamma_\delta}{\tau_1} + \frac{\gamma_\epsilon}{\tau_2} + \frac{4a_3c_3-b_3^2}{8a_3})^{\alpha_\rho+\alpha_\delta+\alpha_\epsilon+\frac{n_1}{2}}}, \tag{39}$$

$p(\tau_1) \sim IG(\alpha_\delta + \frac{1}{2}, 2)$, $p(\tau_2) \sim IG(\alpha_\epsilon, 2)$, and $p(\tau_1)$ and $p(\tau_2)$ are independent.

The problem in (39) can be viewed as a stochastic programming problem and solved by using the Sample Average Approximation method (Ruszczynski and Shapiro 2003). Generate Monte Carlo samples (τ_1^s, τ_2^s) from $p(\tau_1,\tau_2), s = 1,\cdots,S$, and estimate L_2 by

$$\widehat{L}_2 = \frac{1}{S}\sum_{s=1}^{S} f(\tau_1^s,\tau_2^s). \tag{40}$$

We refer to an optimal solution $\widetilde{\boldsymbol{\phi}}_\rho$ and $\widetilde{\boldsymbol{\phi}}_\delta$ of the problem

$$\max_{\boldsymbol{\phi}_\rho,\boldsymbol{\phi}_\delta} \widehat{L}_2 \tag{41}$$

as the *simulated posterior mode*. When S is large, the simulated posterior mode will be close to the true posterior mode (Ruszczynski and Shapiro 2003).

2.2.6 Simplifications when y_h is deterministic

Suppose y_h is deterministic (i.e., $\epsilon(\cdot) = 0$ in (23)), which is the case for the problem of detailed vs. approximate computer experiments. Some parts of the aforementioned procedure can be simplified as follows.

1. Sampling from $p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2 | \mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_3)$.

Because $\tau_2 = 0$ in the model, $p(\sigma_\rho^2, \tau_1, \tau_2)$ in (28) is simplified by dropping its parts involving α_ϵ,γ_ϵ and τ_2; similarly, $p(\tau_1, \tau_2 | \mathbf{y}_l, \mathbf{y}_h, \overline{\tau_1, \tau_2})$ and $\mathbf{M}$ are simplified by removing the parts involving α_ϵ,γ_ϵ and τ_2.

2. Bayesian prediction.

The simplification gives another desirable property of the proposed method.

Theorem 2.1. *If y_h is deterministic, the predictor $E(y_h(\mathbf{x}_i)|\mathbf{y}_l, \mathbf{y}_h) = y_h(\mathbf{x}_i)$ and the prediction variance $Var(y_h(\mathbf{x}_i)|\mathbf{y}_l, \mathbf{y}_h) = 0$ for $\mathbf{x}_i \in D_h$.*

Proof. It is clear from (30) that for $\mathbf{x}_i \in D_h$ the posterior density $p_{y_h(\mathbf{x}_i)}(t|\mathbf{y}_l, \mathbf{y}_h, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) = I\{t = y_h(\mathbf{x}_i)\}$. Therefore, $E(y_h(\mathbf{x}_i)|\mathbf{y}_l, \mathbf{y}_h) = y_h(\mathbf{x}_i)$ and $Var(y_h(\mathbf{x}_i)|\mathbf{y}_l, \mathbf{y}_h) = 0$. □

This property implies that the predictor from the integrated analysis smoothly interpolates all the HE data points.

3. Estimation of correlation parameters.

Because $\tau_2 = 0$ in the model, L_1 in (34) is simplified by dropping its parts involving α_ϵ,γ_ϵ and τ_2, and becomes a one-dimensional integral; similarly, L_2 in (38) is simplified by removing the part involving α_ϵ,γ_ϵ and τ_2, and becomes a stochastic program with one random variable.

2.2.7 Comparison with existing methods

There are major differences between the proposed method and those in Kennedy and O'Hagan (2000) and Qian et al. (2006). The latter two consider integrating data from two deterministic experiments, while ours is applicable to experiments with or without

measurement errors. Ours is also more flexible in the modeling strategy. Kennedy and O'Hagan uses an autoregressive model as an adjustment model with a constant chosen for scale adjustment, which cannot handle complex scale change from LE to HE. Qian et al. uses a regression model, which captures linear part, for the scale change. By utilizing a Gaussian process model, the scale adjustment in (23) can account for non-linear and complex changes as evidenced in the analysis of two examples given later. Qian et al. adopts a frequentist formulation and thus cannot account for uncertainties in the model parameters. In the Bayesian approach of Kennedy and O'Hagan, non-informative priors for the model parameters are assumed. They also use a plug-in estimate $\widehat{\rho}$ in the prediction, which cannot account for the variation in $\widehat{\rho}$. Our approach uses informative priors for the adjustment parameters. The prediction in our approach is based on the Bayesian predictive density function in (26) so that the uncertainties in the model parameters are reflected.

2.3 Example 1: designing linear cellular alloys

We consider part of the data used in Qian et al. (2006), which consists of the outputs from computer simulations for a heat exchanger used in an electronic cooling application. As illustrated in Figure 6, the device is used to dissipate heat generated by a heat source such as a microprocessor. The response y of interest is the total rate of steady state heat transfer of the device, which depends on the mass flow rate of entry air $\dot{m}$, the temperature of entry air T_{in}, the temperature of the heat source T_{wall} and the solid material thermal conductivity k. The device is assumed to have fixed overall width (W), depth (D), and height (H) of 9, 25, and 17.4 millimeters, respectively. Two types of simulations are used in this study: a detailed but slow simulation based on FLUENT finite element analysis and an approximate but fast simulation using finite difference method. These two simulations are referred to as detailed simulation (DS) with response y_d and approximate simulation (AS) with response y_a respectively. Each DS run requires two to three orders of magnitude more computing time than the corresponding AS run. For example, the first run in Table 1 requires 1.75 hours and 2 seconds for DS and AS respectively on a 2.0 GHz Pentium 4 PC with 1 GB of RAM. Details on the engineering background can be found in Qian et al. (2006).

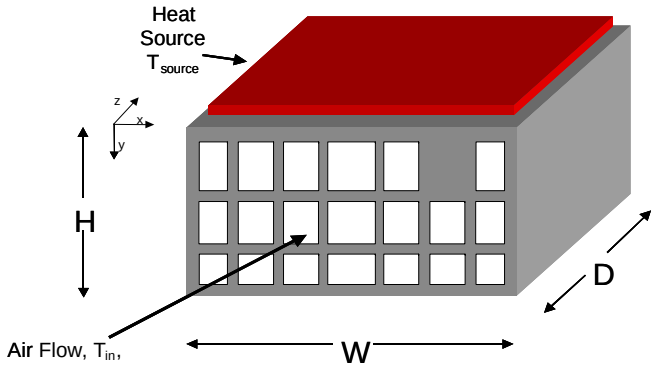

Figure 6: A generic example of linear alloy array.

Table 6 gives the data consisting of 32 AS runs and 32 DS runs. The values of design variables are given in columns 1-4 of the table and the responses from the two experiments are given in columns 5 and 6. The data is divided into a training set and a testing set. We fit BHGP models using a training set consisting of 24 randomly selected DS runs and all 32 AS runs. The remaining 8 DS runs (i.e., run no. 1, 4, 9, 11, 13, 23, 25 and 27 as in the table) are left to form the testing set for model validation. Column 7 in the table gives the status of each DS run as training or testing.

Table 6 shows that the four design variables have different scales and are thus standardized.

The values of hyper-parameters used in this example are given in Table 7. They are chosen to reflect our understanding of the model parameters. The "vague" prior $IG(2,1)$ is chosen for σ_l^2,σ_ρ^2 and σ_δ^2. The "location-flat" priors $N(0, \mathbf{I}_{7\times 7}\sigma_l^2)$ and $N(0, \sigma_\delta^2)$ are chosen for $\boldsymbol{\beta}_l$ and δ_0, and the "scale-flat" prior $N(1, \sigma_\rho^2)$ for ρ_0. The prior for each correlation parameters in $\boldsymbol{\theta}_3$ is $G(2, 0.1)$, having high variance of 200.

Posterior modes of the correlation parameters are given in Table 8. Because calculating $\widehat{\boldsymbol{\phi}}_\rho$ and $\widehat{\boldsymbol{\phi}}_\delta$ needs solving a stochastic programming problem in (38) with one random variable, the Monte Carlo sample size S in (40) is fixed at 20 to achieve good approximation to the one-dimensional expectation. The *optim* function in R is used for non-linear optimization.

The intensive Bayesian computation is implemented in *WinBugs*, a general-purpose Bayesian computing environment. It is found that convergence of Markov Chain is achieved after the first 5000 burn-in iterations. Additional 5000 runs are then generated for posterior calculations.

The posterior mean (255) of σ_l^2 is large, indicating high uncertainty about AS. The posterior mean of β_0 is 22.29. The posteriors of the linear coefficients $\beta_{li}, i = 1, \ldots, 4$ are given in Figures 7(a)-(d). Table 9 gives the posterior means and 95% credible HPD intervals. It is clear from the table that β_{l4} and β_{l1} are more significant than β_{l2} and β_{l3}. The latter's intervals are relatively large and contain zero.

Figures 8(a)-(d) show the posteriors of the adjustment parameters ρ_0, σ_ρ^2, δ_0 and σ_δ^2.

Table 6: Data from linear cellular alloy experiment

Run #	$\dot{m}(kg/s)$	$T_{in}(K)$	$k(W/mk)$	$T_{wall}(K)$	y_a	y_d	Status
1	0.0005	293.15	362.73	393.15	25.61	23.54	Test
2	0.00055	315	310	365	21.23	20.15	Train
3	0.000552	293.53	318.63	388.29	11.44	10.17	Train
4	0.000557	290.18	298.27	377.49	15.03	15.29	Test
5	0.00056	277.01	354.98	374	18.55	18.39	Train
6	0.000566	285.77	266.71	367.27	20.74	20.52	Train
7	0.000578	302.17	358.13	343.72	30.23	30.12	Train
8	0.00058	272.26	211.71	333.65	18.13	18.18	Train
9	0.000589	278.16	225.78	351.83	25.02	24.68	Test
10	0.000594	279.54	258.51	360.13	17.92	19.05	Train
11	0.000603	296.75	323.15	399.45	24.20	24.96	Test
12	0.000612	280.83	291.53	394.72	17.47	16.95	Train
13	0.000615	300.28	270.74	335.79	22.48	22.30	Test
14	0.00062	275	225	340	25.07	19.57	Train
15	0.000626	284.89	350.46	352.29	18.93	23.33	Train
16	0.000627	287.6	243.96	382.54	18.17	14.36	Train
17	0.000652	298.04	303.96	361.58	13.75	21.31	Train
18	0.000657	294.24	330.63	375.53	29.08	36.11	Train
19	0.00067	303.07	321.41	370.48	22.21	25.37	Train
20	0.00068	313.28	259.12	350	21.6	22.89	Train
21	0.000683	287.05	227.31	358.24	30.9	34.45	Train
22	0.000689	272.7	260.91	355.37	13.08	14.83	Train
23	0.000694	278.35	212.79	376.24	16.4	18.78	Test
24	0.000698	277.52	299.39	338.4	31.14	32.85	Train
25	0.0007	288.15	300	400	13.54	17.41	Test
26	0.000711	292.26	273.31	392.54	7.02	7.48	Train
27	0.000714	283.08	306.69	344.34	35.53	42.93	Test
28	0.00073	285.51	217.74	383.92	20.92	21.97	Train
29	0.000738	295.01	295.02	347.22	25.53	19.77	Train
30	0.000741	270.95	275.19	356.87	10.23	4.55	Train
31	0.000751	287.99	326.02	354.08	36.56	47.05	Train
32	0.000757	300.64	235.03	391.68	27.24	25.82	Train

Table 7: Prior hyper-parameters for linear cellular alloy experiment

Hyper-parameter	Value
α_l	2
γ_l	1
α_ρ	2
γ_ρ	1
α_δ	2
γ_δ	1
$\mathbf{u}_l$	$(0, 0, 0, 0, 0)^t$
v_l	1
u_ρ	1
v_ρ	1
u_δ	0
v_δ	1
a_l	2
b_l	0.1
a_ρ	2
b_ρ	0.1
a_δ	2
b_δ	0.1

Table 8: Posterior modes of correlation parameters for linear cellular alloy experiment

Correlation parameter	Posterior mode
ϕ_{l1}	2.83
ϕ_{l2}	2.13
ϕ_{l3}	22.65
ϕ_{l4}	12.87
$\phi_{\rho 1}$	3.22
$\phi_{\rho 2}$	7.23
$\phi_{\rho 3}$	1.26
$\phi_{\rho 4}$	1.38
$\phi_{\delta 1}$	2.26
$\phi_{\delta 2}$	0.74
$\phi_{\delta 3}$	6.92
$\phi_{\delta 4}$	7.24

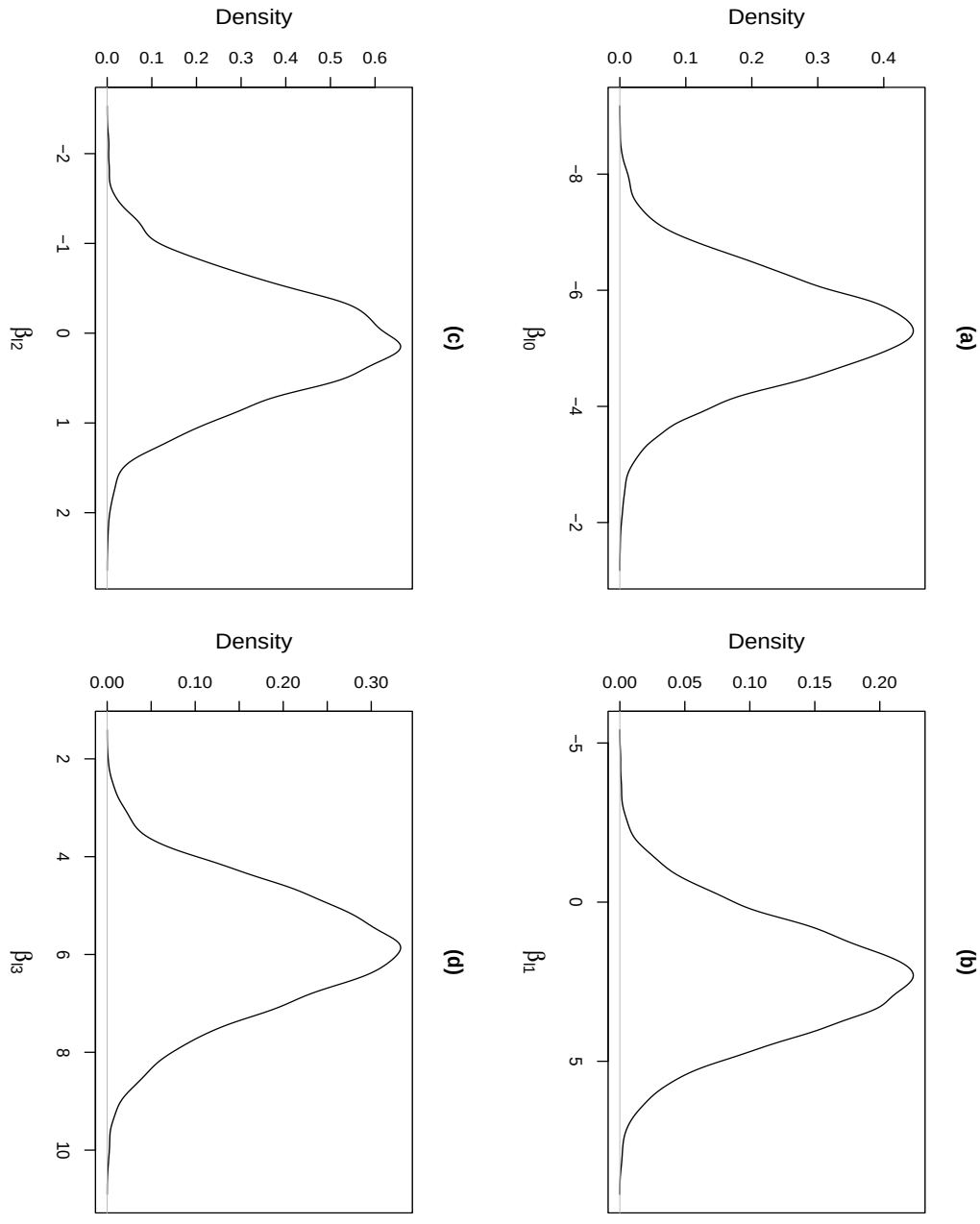

Figure 7: Posteriors of $\boldsymbol{\beta}_l$ for linear cellular alloy experiment.

Table 9: Posterior means and 95% percent credible intervals for $\boldsymbol{\beta}_l$ in linear cellular alloy experiment

	Posterior mean	Lower Bound	Upper Bound
β_{l1}	-5.334	-7.141	-3.504
β_{l2}	2.379	-1.171	5.78
β_{l3}	0.09424	-1.128	1.255
β_{l4}	5.896	3.639	8.373

Their means and 95% credible HPD intervals are given in Table 10. Several interesting observations have emerged. First, the plot for ρ_0 is multi-modal, indicating complex scale change from AS to DS. Second, σ_ρ^2 and σ_δ^2 are relatively small, indicating a good fit of the adjustment model. Third, the average response 21.89 for 24 DS runs is close to 21.13, the average for the corresponding 24 AS runs. Table 5 shows no consistent pattern in comparing the DS and AS values. This is different from the example in Section 2.4, where one experiment consistently gives higher values than the other. For the current example, a simple mean comparison analysis will yield little information, whereas the proposed method can unveil complex relationships between AS and DS.

Table 10: Posterior means and 95% credible HPD intervals for ρ_0, σ_ρ^2, δ_0 and σ_δ^2 in linear cellular alloy experiment

Parameter	Posterior mean	Lower Bound	Upper Bound
ρ_0	1.05	0.94	1.13
σ_ρ^2	0.29	0.16	0.49
δ_0	0.14	-1.24	1.93
σ_δ^2	0.78	0.18	2.70

Finally, we compare predictions on eight untried runs using BGHP models with those from the separate analysis as well as those using the methods of Kennedy and O'Hagan (2000) and Qian et al. (2006). The separate analysis builds a Bayesian Gaussian process model using 24 DS runs, while the other three methods fit both the AS and DS data. The predictions of the eight runs are given in Table 11. In the table, column 1 gives the corresponding run numbers in Table 5; columns 2-5 give the values of $\widehat{y}_d$ of the integrated

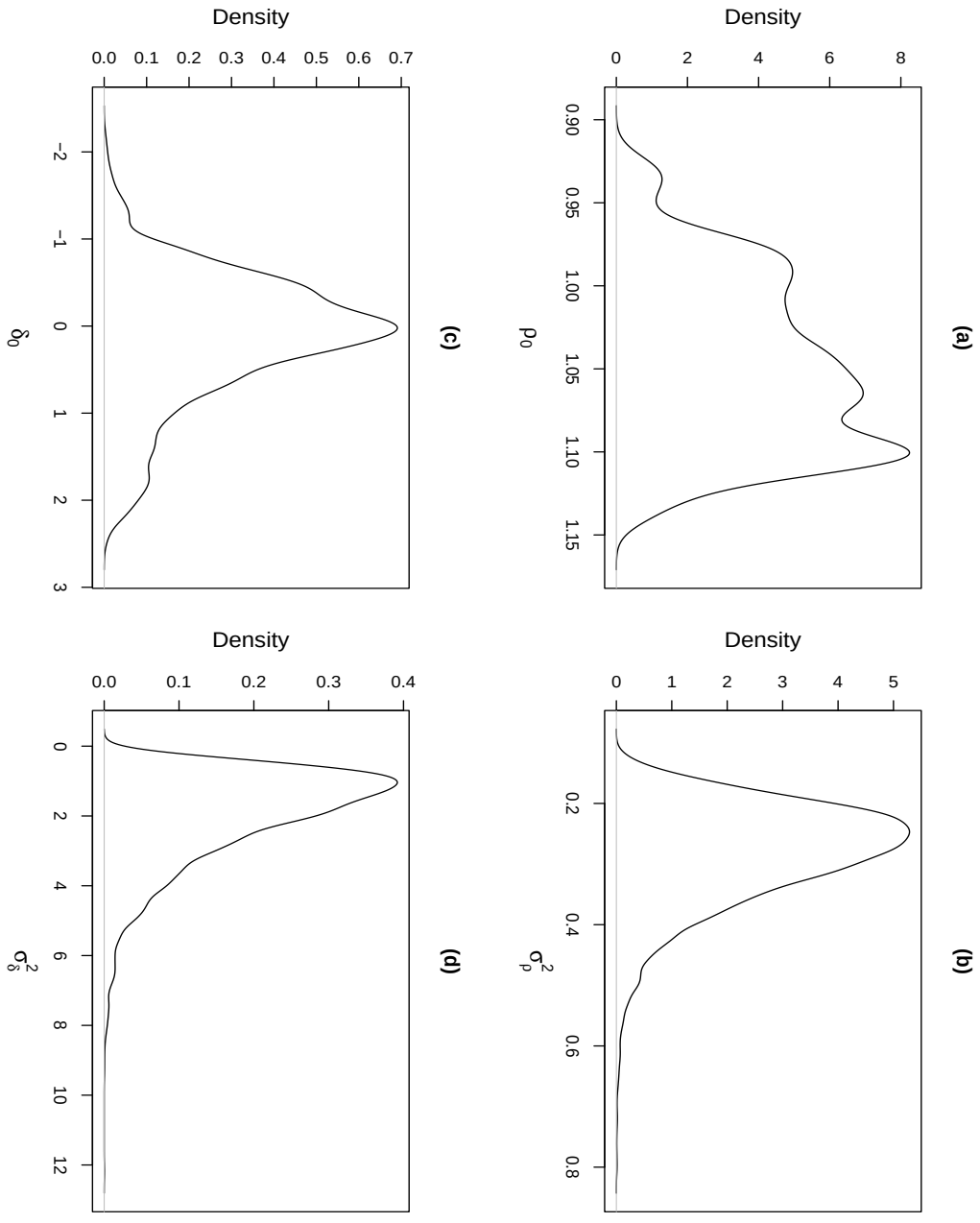

Figure 8: Posteriors of ρ_0, σ_ρ^2, δ_0 and σ_δ^2 for linear cellular alloy experiment.

analysis, the separate analysis, the Kennedy-O'Hagan method and the Qian et al. method, respectively, and column 6 gives the y_d values from DS. The RMSEs (root-mean-square-errors) for the four methods (in the same order) are 7.83, 9.48, 9.29 and 8.77, respectively. The three methods that fit both AS and DS runs give better prediction results than the separate analysis. Among these three, the proposed method outperforms the other two by 16% and 11% respectively.

Table 11: Prediction results on eight untried points for linear cellular alloy experiment

Run #	$\widehat{y}_d^1$	$\widehat{y}_d^2$	$\widehat{y}_d^3$	$\widehat{y}_d^4$	y_d
1	26.41	23.35	21.77	21.35	23.54
4	16.23	23.74	14.33	14.76	15.29
9	23.66	22.57	22.76	23.83	24.68
11	25.51	18.22	15.87	15.44	24.96
13	22.07	29.55	21.32	22.79	22.30
23	16.74	20.04	17.58	18.21	18.78
25	16.99	27.56	15.46	15.56	17.41
27	21.11	21.95	20.16	18.62	42.93

2.4 Example 2: fluidized bed processes

Dewettinck et al. (1999) reported a physical experiment and several associated computer models for predicting the steady-state thermodynamic operation point of a GlattGPC-1 fluidized-bed unit. The base of the unit consists of a screen and an air jump, with coating sprayers at the side of the unit. Reese et al. (2004) proposed a linear model approach to analyze a sample example in Dewettinck et al. The same data will be analyzed using the proposed BHGP models.

Several variables that can potentially affect the steady-state thermodynamic operating point are: fluid velocity of the fluidization air (V_f), temperature of the air from the pump (T_a), flow rate of the coating solution (R_f), temperature of the coating solution (T_s), coating solution dry matter content (M_d), pressure of atomized air (P_a), temperature (T_r) and humidity (H_r).

Dewettinck et al. (1999) considered 28 different process conditions with coating solution used for distilled water (i.e., $M_d = 0$) and the room temperature set at 20^oC. As a result, six factors ($H_r, T_r, T_a, R_f, P_a, V_f$) with different values are considered in the analysis. These values are given in Table 12.

For each factor combination, one physical run ($T_{2,\text{exp}}$) and three computer runs ($T_{2,1}$,$T_{2,2}$ and $T_{2,3}$) were conducted. The results are given in Table 13.

There are major differences among the three computational models (see Dewettinck et al. 1999 for details). In summary, $T_{2,3}$, which includes adjustments for heat losses and inlet airflow, is the most accurate (i.e., producing the closest response to $T_{2,\text{exp}}$). The computer model $T_{2,2}$ includes only the adjustment for heat losses. The model $T_{2,1}$ does not adjust for heat losses or inlet airflow and is thus the least accurate.

For illustration, we only synthesize data from the physical experiment and the second computer model $T_{2,2}$, which has medium accuracy. The responses from $T_{2,\text{exp}}$ and $T_{2,2}$ are denoted by $y_{2,\text{exp}}$ and $y_{2,2}$ respectively.

It is clear from Table 12 that the six process variables have different scales and should be standardized. The data set is divided into a training set and a testing set. The training set, used to build BHGP models, consists of 20 randomly sampled $T_{2,\text{exp}}$ runs and all 28

Table 12: Six process variables for fluidized bed process experiment

Run #	$H_r(\%)$	$T_r(C)$	$T_a(C)$	$R_f(g/min)$	P_a(bar)	$V_f(m/s)$
1	51.00	20.70	50.00	5.52	2.50	3.00
2	46.40	21.30	60.00	5.53	2.50	3.00
3	46.60	19.20	70.00	5.53	2.50	3.00
4	53.10	21.10	80.00	5.51	2.50	3.00
5	52.00	20.40	90.00	5.21	2.50	3.00
6	45.60	21.40	60.00	7.25	2.50	3.00
7	47.30	19.50	70.00	7.23	2.50	3.00
8	53.30	21.40	80.00	7.23	2.50	3.00
9	44.00	20.10	70.00	8.93	2.50	3.00
10	52.30	21.60	80.00	8.91	2.50	3.00
11	55.00	20.20	80.00	7.57	1.00	3.00
12	54.00	20.60	80.00	7.58	1.50	3.00
13	50.80	21.10	80.00	7.40	2.00	3.00
14	48.00	21.20	80.00	7.43	2.50	3.00
15	42.80	22.40	80.00	7.51	3.00	3.00
16	55.70	20.80	50.00	3.17	1.00	3.00
17	55.20	20.70	50.00	3.18	1.50	3.00
18	54.40	20.70	50.00	3.19	2.00	3.00
19	55.40	19.80	50.00	3.20	2.50	3.00
20	52.90	20.00	50.00	3.19	3.00	3.00
21	28.50	18.30	80.00	7.66	2.50	3.00
22	26.10	19.00	80.00	7.69	2.50	4.00
23	24.20	18.90	80.00	7.69	2.50	4.50
24	25.40	18.50	80.00	7.70	2.50	5.00
25	45.10	19.60	50.00	3.20	2.50	3.00
26	43.10	20.30	50.00	3.23	2.50	4.00
27	42.70	20.40	50.00	3.20	2.50	4.50
28	38.70	21.60	50.00	3.22	2.50	5.00

Table 13: Results from fluidized bed process experiment

Run#	$T_{2,\text{exp}}$	$T_{2,1}$	$T_{2,2}$	$T_{2,3}$
1	30.40	32.40	31.50	30.20
2	37.60	39.50	38.50	37.00
3	45.10	46.80	45.50	43.70
4	50.20	53.80	52.60	51.00
5	57.90	61.70	59.90	58.20
6	32.90	35.20	34.60	32.60
7	39.50	42.40	41.00	39.10
8	45.60	49.50	48.50	46.40
9	34.20	37.50	36.60	34.80
10	41.10	45.50	44.30	42.00
11	45.70	50.50	49.00	47.00
12	44.60	49.80	48.40	46.30
13	44.70	49.80	48.40	46.30
14	44.00	49.20	48.00	45.70
15	43.30	48.60	47.50	45.40
16	37.00	39.50	38.00	37.70
17	37.20	39.50	38.50	37.10
18	37.10	39.50	37.50	36.70
19	36.90	39.50	38.50	36.10
20	36.80	37.70	37.20	36.20
21	46.00	48.70	47.30	45.10
22	54.70	57.70	56.20	54.20
23	57.00	60.10	58.70	57.00
24	58.90	62.00	60.50	58.70
25	35.90	37.90	37.10	36.10
26	40.30	41.70	40.80	40.10
27	41.90	43.00	42.30	41.40
28	43.10	43.90	43.30	42.60

$T_{2,2}$ runs. The remaining eight $T_{2,\text{exp}}$ runs (i.e., run no. 4, 15, 17, 21, 23, 25, 26 and 28 as in Table 12) are left to form the testing set for model validation.

First, we fit a model for $T_{2,2}$. As reported in Reese et al. (2004), some of the covariates are highly correlated (as high as 0.82), indicating possible collinearity. Although the problem of collinearity poses difficulty for a linear model approach, it is not as severe for a Gaussian process model, in which a correlation function instead of linear correlations plays a key role in model building.

As stated in Reese et al., a full second-order linear model is saturated for this example, given its relatively small run size. As a solution, they implemented a Bayesian variable selection procedure (Wu and Hamada 2000) to find several "most likely" sub-models. Instead of relying on linear models, the proposed method fits a Gaussian process model including all model parameters (mean and correlation parameters) at once, thus avoiding the complex sub-model selection procedure.

Table 14 gives the values of the hyper-parameters used in this example. Because little knowledge about model parameters is known beforehand, "vague" priors are chosen. The priors for $\sigma_l^2, \sigma_\rho^2, \sigma_\delta^2$ and σ_ϵ^2 are $IG(2,1)$. The "location-flat" priors $N(0, \mathbf{I}_{7\times 7}\sigma_l^2)$ and $N(0, \sigma_\delta^2)$ are chosen for $\boldsymbol{\beta}_l$ and δ_0, and "scale-flat" prior $N(1, \sigma_\rho^2)$ for ρ_0. The prior for each correlation parameter is $G(2, 0.1)$, having variance as high as 200.

The posterior modes for the correlation parameters are given in Table 15. Because calculating $\widehat{\boldsymbol{\phi}}_\rho$ and $\widehat{\boldsymbol{\phi}}_\delta$ needs solving a stochastic programming problem in (38) with two random variables, the Monte Carlo sample size S in (40) is fixed at 50 to achieve good approximation for the two-dimensional expectation. The *optim* function in R is used for non-linear optimization.

The intensive Bayesian computation is implemented in *WinBugs*. Convergence of Markov Chain is achieved after the first 5000 burn-in iterations. Additional 5000 runs are then generated for posterior calculations.

The posterior mean of β_0 is 35.52. Figures 9(a)-(f) plot the posteriors for the linear coefficients $\beta_{li}, i = 1, \ldots, 6$. The means and 95% credible HPD intervals are shown in Table 16. These intervals are relatively large and contain zero. If these results were obtained

Table 14: Prior hyper-parameters for fluidized bed process experiment

Hyper-parameter	Value
α_l	2
γ_l	1
α_ρ	2
γ_ρ	1
α_δ	2
γ_δ	1
α_ϵ	2
γ_ϵ	1
$\mathbf{u}_l$	$(0, 0, 0, 0, 0, 0, 0)^t$
v_l	1
u_ρ	1
v_ρ	1
u_ρ	0
v_ρ	1
a_l	2
b_l	0.1
a_ρ	2
b_ρ	0.1
a_δ	2
b_δ	0.1

Table 15: Posterior modes of correlation parameters in fluidized bed process experiment

Correlation Parameter	Posterior Mode
ϕ_{l1}	3.43
ϕ_{l2}	6.36
ϕ_{l3}	3.79
ϕ_{l4}	5.74
ϕ_{l5}	4.57
ϕ_{l6}	5.49
$\phi_{\rho 1}$	3.13
$\phi_{\rho 2}$	3.95
$\phi_{\rho 3}$	3.23
$\phi_{\rho 4}$	11.41
$\phi_{\rho 5}$	13.17
$\phi_{\rho 6}$	9.19
$\phi_{\delta 1}$	11.16
$\phi_{\delta 2}$	5.34
$\phi_{\delta 3}$	8.97
$\phi_{\delta 4}$	6.98
$\phi_{\delta 5}$	6.72
$\phi_{\delta 6}$	21.88

from a linear model, we would suspect that some of these effects may not be statistically significant and further analysis is needed to remove insignificant ones from the model. Because a Gaussian process model has a simple mean structure (i.e., including linear effects only), any further simplification of the mean part will yield little benefit. Furthermore, for a Gaussian process model the complex relationship between the inputs and the response is primarily explained by the correlation structure rather than the mean structure. Therefore, all the linear coefficients are retained in the model. The posterior mean (137) for σ_l^2 is large, indicating high uncertainty about $T_{2,2}$. The posterior mean 0.2623 of the measurement error σ_ϵ^2 is relatively small with standard deviation 0.1823. Therefore the model uncertainty (σ_l^2) is much more pronounced than the observation uncertainty (σ_ϵ^2).

The posteriors of ρ_0, σ_ρ^2, δ_0 and σ_δ^2, associated with the adjustments are shown in Figures 10. The means and 95% credible HPD intervals are given in Table 17. The results indicate several important and appealing aspects of the integrated analysis. First, the density plot of ρ_0 has three modes, implying *intricate scale change* from $y_{2,2}$ to $y_{2,\text{exp}}$. Any attempt

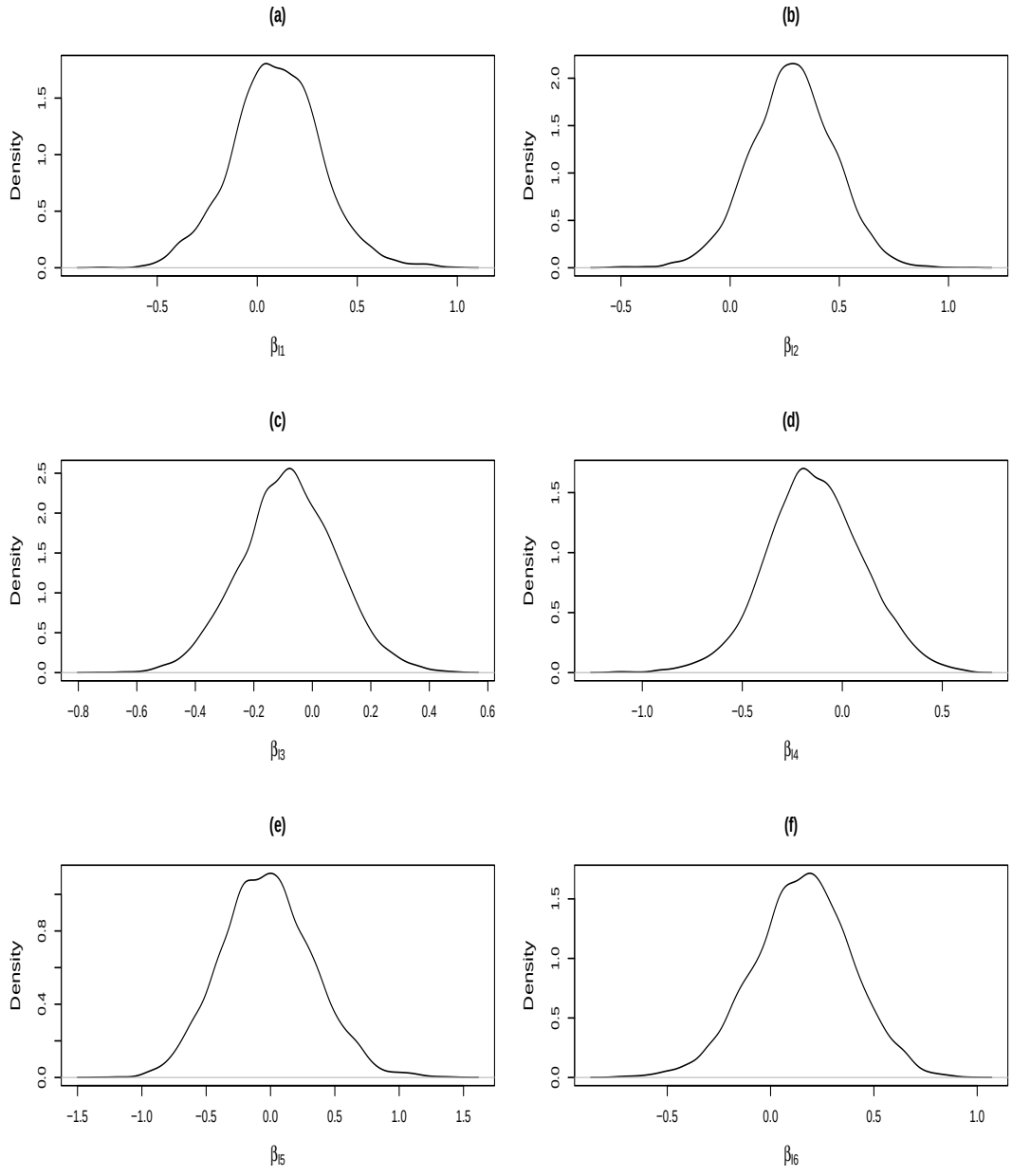

Figure 9: Posteriors of $\boldsymbol{\beta}_l$ for fluidized bed process experiment.

Table 16: Posterior means and 95% percent credible intervals for $\boldsymbol{\beta}_l$ in fluidized bed process experiment

	Posterior mean	Lower Bound	Upper Bound
β_{l1}	0.095	-0.362	0.553
β_{l2}	0.284	-0.100	0.651
β_{l3}	-0.080	-0.401	0.247
β_{l4}	-0.137	-0.618	0.346
β_{l5}	-0.024	-0.691	0.695
β_{l6}	0.155	-0.318	0.617

to simplify the scale term to a constant will fail to model this change adequately. The capability of modeling complex scale change comes as a benefit of the Bayesian formulation. A frequentist's analysis can only produce a point estimate for ρ_0 unless a complicated mixture model is correctly employed and asymptotic distributions are obtained. Second, from Table 12 σ_ρ^2 and σ_δ^2 are relatively small in relation to σ_l^2 (137). Although we cannot make a conclusive statement about the utility of the adjustment model for the current example, these small values do indicate that the two data sources are well integrated in the analysis. Third, the average response 42.33 for 20 $T_{2,\text{exp}}$ runs is lower than 44.22, the average for the corresponding 20 $T_{2,2}$ runs. This observation comes as no surprise, as for each run in Table 16, $T_{2,\text{exp}}$ consistently produces a lower response than $T_{2,2}$. On average, the difference is -1.89. However, the posterior mean for δ_0 is -0.01, which is much smaller than -1.89 in magnitude. This is due to the inclusion of scale adjustment in the model. The scale change may be of significant interest to the experimenters and this treasured information is uncovered by the proposed analysis.

Table 17: Posterior means and 95% credible hpd intervals for ρ_0, σ_ρ^2, δ_0 and σ_δ^2 in fluidized bed process experiment

Parameter	Posterior Mean	Lower Bound	Upper Bound
ρ_0	0.89	0.73	1.10
σ_ρ^2	0.12	0.06	0.22
δ_0	-0.01	-1.76	1.82
σ_δ^2	2.15	0.36	5.92

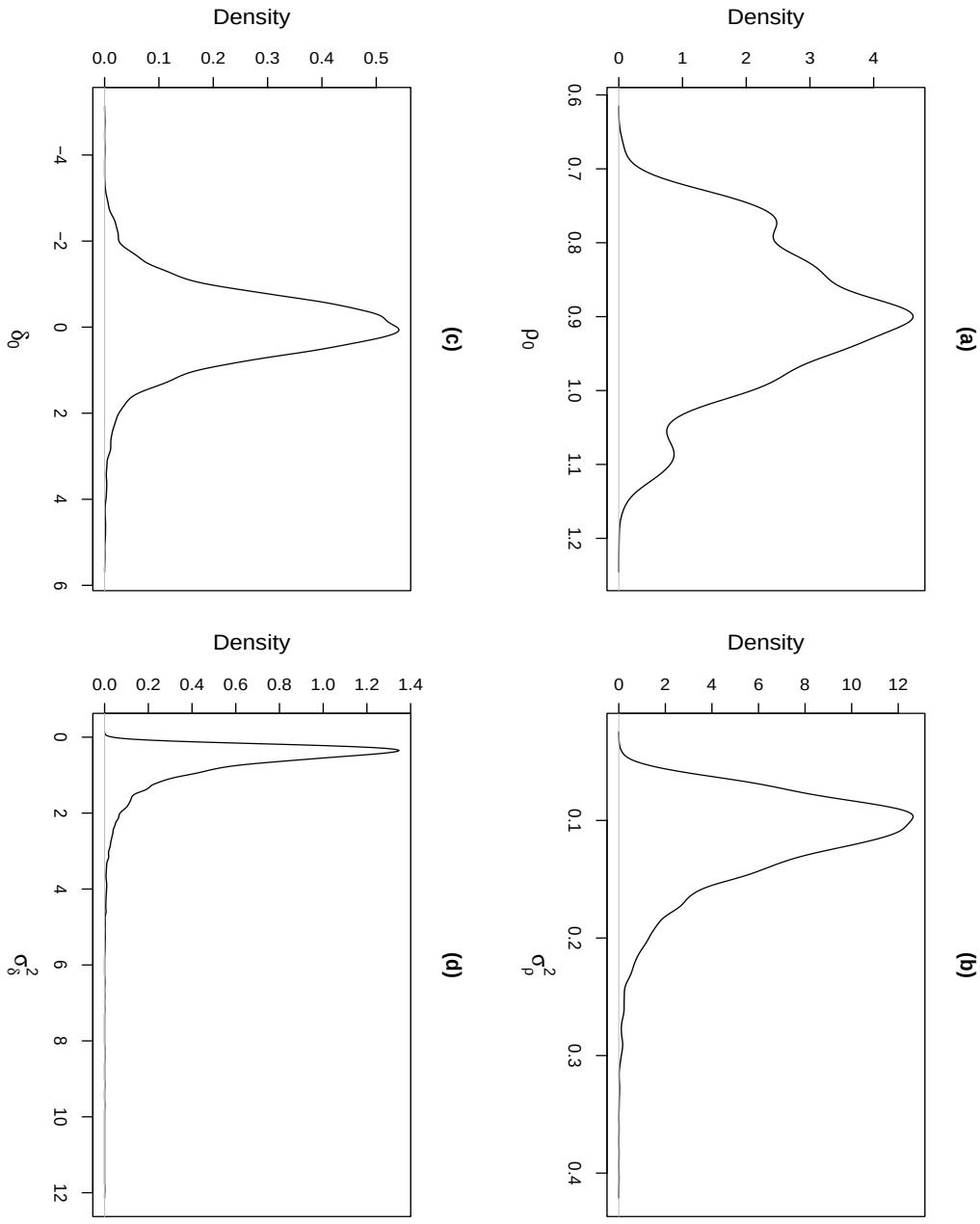

Figure 10: Posterior of ρ_0, σ_ρ^2, δ_0 and σ_δ^2 for fluidized bed process experiment.

Finally, we assess the prediction accuracy of the proposed method by comparing it with that of a separate analysis. For the latter, 20 T_{exp} runs are used to fit a Bayesian Gaussian process model. Table 18 lists the prediction results on eight untried points with column 1 giving the run no's of the testing runs and columns 2, 3, 4 giving the values of $\widehat{y}_{2,\text{exp}}$ from the integrated analysis, $\widehat{y}_{2,\text{exp}}$ from the separate analysis, the values of $y_{2,\text{exp}}$, respectively. In general, the integrated analysis produces better results. The RMSE (root-mean-square-error) of the integrated analysis is 8.40, which is 10% smaller than the RMSE (9.33) of the separate analysis. The two RMSEs are relatively large. This is not unexpected since the numbers of runs used for model building and validation are both limited. Nevertheless, even for such small run sizes, the integrated analysis significantly improves prediction over the separate analysis.

Table 18: Prediction results on eight untried points for fluidized bed process experiment

Run #	$\widehat{y}_h$ from integrated analysis	$\widehat{y}_h$ from separate analysis	y_h
4	48.06	38.34	50.20
15	31.34	26.42	43.30
17	48.27	30.00	37.20
21	39.66	48.41	46.00
23	48.54	50.24	57.00
25	28.47	34.65	35.90
26	40.00	34.26	40.30
28	31.81	31.72	43.10

2.5 Concluding remarks and extensions

This chapter has developed some hierarchical Gaussian process models for modeling and integrating LE and HE data. Use of the adjustment model in (23) allows a flexible location and scale adjustment of the more abundant but less accurate LE data to be closer to the HE data. Use of MCMC and Sample Average Approximation algorithms makes it feasible to carry out the Bayesian computation. By using the Bayesian predictive density in (26), the prediction can incorporate uncertainties in the model parameters. As demonstrated by the results in Sections 4 and 5, the proposed method can better account for the heterogeneity between the LE and HE data and increase the prediction accuracy.

Extensions of the present work can be made in several directions. First, the Bayesian prediction in (26) uses a point estimate of the correlation parameters $\boldsymbol{\theta}_3$, which is developed in Section 2.5. A strictly Bayesian approach would also compute the posterior of $\boldsymbol{\theta}_3$ for the prediction. While this may produce better results, it would greatly increase the computational work because the formulas in (33) and (34) have no tractable form to render the use of MCMC. This extension would be feasible only if a computational shortcut can be found. Second, the proposed method can be extended to more than two sources like low-, medium- and high-accuracy experiments in a relatively straightforward way.

2.6 References

Berger, J. O., Oliveira, V. D. and Sansó, B. (2001), "Objecitve Bayesian analysis of spatially correlated Data," *Jounral of the American Statistical Association*, **96**, 1361-1374.

Belisle, C. J. P. (1992), "Convergence theorems for a class of simulated annealing algorithms on R^d," *Journal of Applied Probability*, **29**, 885-895.

Dewettinck, K., Visscher, A. D., Deroo, L. and Huyghbbaet, A. (1999), "Modeling the steady-state thermodynamic operation point of top-spray fluidized bed processing," *Journal of Food Engineering*, **39**, 131-143.

Handcock, M. S. and Stein, M. L. (1993), "A Bayesian analysis of kriging," *Technometrics*, **35**, 403-410.

Handcock, M. S. and Wallis, J. R. (1994), "An approach to statistical spatial-temporal modeling of meteorological fields," *Jounral of the American Statistical Association*, **89**, 368-378.

Kennedy, M. C. and O'Hagan (2000), "Predicting the output from a complex computer code when fast approximations are available," *Biometrika*, **87**, 1133-1152.

Liu, J. S. (2001), *Monte Carlo Strategies in Scientific Computing*, New York: Springer.

Qian, Z., Seepersad, C. C., Joseph, V. R., Allen, J. K. and Wu, C. F. J. (2006), "Building surrogate models based on detailed and approximate simulations," *ASME Journal of Mechanical Design, to appear.*

Reese, C. S., Wilson, A. G, Hamada, M. and Martz, H. F., and Ryan, K. J. (2004), "Intergated analysis of computer and physical experiments," *Technometrics*, **46**, 153-164.

Ruszczynski, A. and Shapiro A.(eds) (2003), *Stochastic Programming. Handbooks in Operations Research and Management Science*, **10**, Elsevier.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J. and Morris, M. D. (1992), "Screening, predicting and computer experiments," *Technometrics*, **34**, 15-25.

Wu, C. F. J. and Hamada, M. (2000), *Experiments, Planning, Analysis and Parameter Design Optimization*, New York: John Wiley and Son.

CHAPTER III

A STRUCTURAL EQUATION METHOD FOR TEMPERATURE MODELING IN DATA CENTER COMPUTER EXPERIMENT

3.1 Introduction

In recent years there has been an increasing need in storing, manipulating, accessing to and managing data sets for a wide community of users in public and private sectors of the economy. As an integrated facility housing multiple-unit servers, a data center provides application services or management for various data processing, e.g., web hosting internet, intranet, telecommunication and information technology. Figure 11 shows a schematic layout of an Internet data center using Sun Microsystems (Lawrence Berkeley National Laboratory 2002). Driven by advances in hardware and data storage techniques, data centers now sprawl over thousands of square feet, whose size and capacity are limited only by cost, adequate electricity and the ability to cool the systems.

In designing and running a reliable data center, maintaining the system operating at a temperature within a functional range is essential. Data center facilities are extremely energy intensive. Computer equipments housed within a data center are electrically operated, constantly generating heat. Currently, heat loads of data processing equipment continues to increase at a rapid rate. For example, a recent study (Schmidt 2001a) reports that a rack dissipates 28,500 watts and generates a heat flux based on the footprint of the rack of 20,900 watt/m^2. Therefore it is necessary to install cooling system to maintain a data center at a temperature that meets the user's requirements. Failure to cool the racks will lead to a temperature rise and the system collapses subsequently. Recently, there is a surge of interest in tackling the problem of data center cooling in engineering (Patel et al. 2002; Schmidt 2001ab). There are two primary enclosure cooling options: water cooled and air

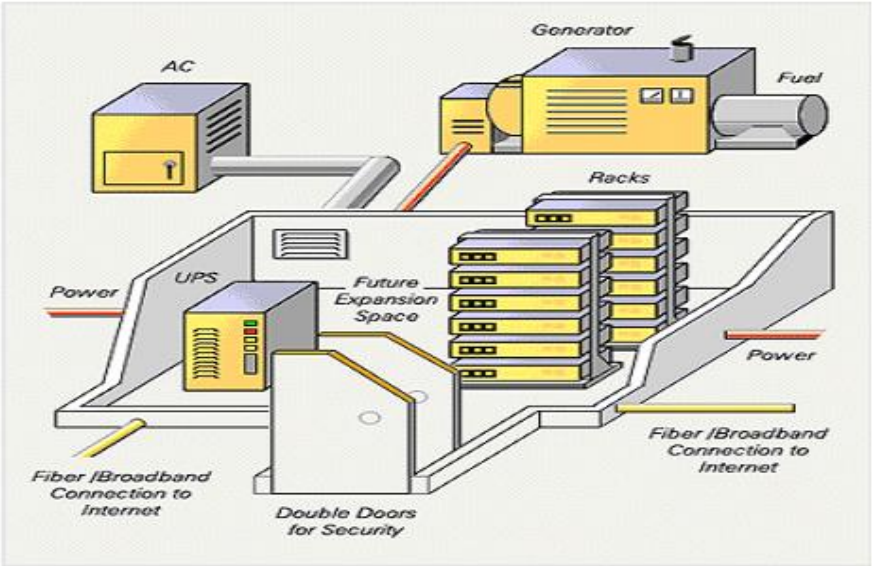

Figure 11: Schematic layout of an internet data center (Sun Microsystems) (Lawrence Berkeley National Laboratory 2002).

cooled cabinets. While the former is an efficient means for mitigating heat loads for some special infrastructure, e.g., a fully loaded, high-density rack, the latter is more general with typically low infrastructure requirements. To achieve efficient cooling and reduce energy consumption, cable racks and cooling systems need to be carefully arranged in a data center. Two widely used layouts are *raised floor layout* and *non-raised floor layout* (Schmidt 2001a). The former includes a raised floor, under which rack cables are connected to maintain a neat structure, while in the latter, chilled cooling air is supplied from the ceiling and warm air exits through exhausts installed on the walls.

Monitoring and studying the temperature of a data center is no easy task. How different configurations affect the thermal distribution is largely unknown. The physical thermal process is complex, depending on many factors, e.g., diffusor angle and ceiling height, and

detailed temperature at different locations cannot be actually measured. The physical experimentation becomes especially difficult when many possible configurations need to be considered. Computer experiment, built on computational fluid dynamics (CFD) models, implemented in professional software packages like Flotherm (Flometrics 2005) and FLUENT (Fluent 1998), is widely used as a proxy to study the air movement and thermal distribution of a data center. In addition to savings in experimental cost and time, this type of computer experiment has other advantages. For example, it is possible to use this experiment to simultaneously produce temperature responses over a continuous region or at many locations of a data center.

Two new defining features of data center computer experiment set it apart from other computer experiments. First, it tends to produce high-dimensional responses instead of a univariate response. The temperature measurements of this experiment are taken over a large region. Many monitor points can be chosen because the average computational cost per measurement point tends to decrease rapidly as the number of points increases. In particular, it is relatively cheap to produce temperature values at multiple points within the same rack simultaneously. Monitor points are often placed at various heights on different positions of cable rows, resulting in high-dimensional temperature readings. Second, the air movement and thermal dynamics of a data center modeled in this experiment depend on high-dimensional configuration variables such as rack temperature rise, rack power and diffuser flow rate. The high-dimensional responses and configuration variables must be addressed in building a data center temperature model. Classical kriging or co-kriging models, which are popular in computer experiments (Santer, Williams and Notz 2003; Fang, Li and Sudijanto 2005), cannot handle the high-dimensional responses and configuration variables. New methods are thus needed. With exceptions like (Rolander et al. 2006), surprisingly little has been done on modeling high-dimensional responses from computer experiments with high-dimensional covariates. A systematic approach will be taken in this chapter by utilizing multivariate statistical analysis methods. The two distinctive features of the current application makes traditional multivariate analysis methods, in particular principal component analysis (PCA) and factor analysis (Basilevsky 1994; Johnson and

Wichern 1998), inapplicable and inappropriate because they lack the abilities to handle high-dimensional responses and configuration variables simultaneously. The approach introduced here, incorporating physical structure and various sources of variability based on a *structural equation system* (Bentler 1995; Bollen 1989; Jöreskog and Sörbom 1996; Wall and Amemiya 2000), leads to the first sound statistical method for the temperature modeling in data center computer experiment. Although the proposed method is motivated and developed for the data center application, similar scenarios exist in many other engineering and science applications, e.g., auto-body assembly processes (Apley and Shi 2001; Apley and Shi 1998; Ceglarek and Shi 1996; Ding, Ceglarek and Shi 2002), to which the method is also applicable. The primary assumptions are that a system produces multivariate responses and the system performance is dependent on many configuration or design variables.

3.2 Design of experiments for configuration variables and placement of monitor points

In this section we will address two important data collection issues related to running a data center computer experiment: design of experiments for configuration variables and placement of monitor points. Factors that determine the thermal mechanisms and air movement of a data center are called *configuration variables* (Schmidt 2003), denoted by $\mathbf{x} = (\mathbf{x}_{con}, \mathbf{x}_{cat})$, where $\mathbf{x}_{con}$ and $\mathbf{x}_{cat}$ are continuous and categorical variables respectively. Throughout this chapter, we focus on modeling air-cooling data centers because they have lower infrastructure requirements and recently become dominant in the IT industry (Schmidt 2001a). Key configuration variables of this type of cabinet are briefly addressed below. The interested readers are referred to (Schmidt 2001b2003) for details.

System layout:

System layout affects the air flow and heat distribution of a data center. There are two major layouts: non-raised floor and raised floor layouts as discussed in Section 3.1.

Rack air temperature rise:

Rack air temperature rise, measured in Celsius (C), determines temperature and humidity of the computer equipments housed in a data center. It is related to rack flow rate. A larger

rack air temperature rise requires a smaller blower while a lower rack air temperature rise needs a larger blower. It takes continuous values chosen within a user-specified bound.

Rack power:

Rack power, measured in kilowatts (KJ/s), interacts with rack air inlet temperature. Racks of higher power tend to lower rack air inlet temperature. It takes continuous values chosen within a user-specified bound.

Diffuser height:

Diffuser height determines the height of diffusers. It affects the overall air flow and thermal distribution of a data center. Diffusers are often placed at evenly spaced heights to efficiently control the air flow in the system. The ceiling height of the system needs to be considered in setting the heights of the diffusors.

Diffuser location/configuration:

Multiple diffusers are needed to cool a typical data center with many cable racks. The diffusers in a data center are often placed at the same height but on different horizontal positions to dissipate cool fluid uniformly. Diffuser location/configuration determines relative locations of the diffusers with respect to the cable rows in a data center.

Diffuser angle:

Diffuser angle determines the orientation of diffusers. Diffuser angle can affect the air flow and cooling efficiency.

Diffuser flow rate:

Diffuser flow rate specifies the flow rate of the diffusers in a data center. It is related to rack flow rate, which is determined by rack power, air density and air specific heat. For a given rack flow rate, diffusor flow rate can be chosen at appropriate fractions of this rate.

Ceiling height:

Ceiling height specifies the height of the ceiling of a data center, which takes categorical or continuous values. It can potentially affect the mal-distribution of existing air flows.

Hot air return vent location:

Chilled air enters an air-cooled data center to cool the system and heats up when it removes

heat from the equipments. After the cooling, warm air exits through return vents. The return vents can be placed either towards the ceiling or close to the floor, and either on the perpendicular wall or on the parallel wall. Hence, this variable has four levels: (1) perpendicular - bottom (per-bot), (2) perpendicular - top (per-top), (3) parallel - bottom (par-bot) and (4) parallel - top (par-top).

Remove/mixed power:
Remove/mixed power specifies the distribution of heat loads in cable rows and takes discrete values.

One major challenge in designing a reliable data center stems from the fact that data center thermal dynamics may vary dramatically in relation to different values of configuration variables. To get a sense of the overall thermal properties of a data center, it is necessary to study the thermal distribution over a large design space of configuration variables. The computational cost for running a data center physical experiment dwarfs that of a computer experiment based on computational fluid dynamics (CFD). However, the latter may still be relatively expensive because it entails solving complex mathematical equations with a large number of reference points (Flometrics 2005). Hence, it is necessary to use an informative and efficient experimental design to carefully choose the values of configuration variables in the experiment. A model-based design scheme is used in this chapter. Let $\mathbf{x}_1, \ldots, \mathbf{x}_N$ denote N runs of a design of size N, denoted by D. Although other models can be equivalently used, for convenience a second-order model is imposed to represent the input-and-output relationship in the computer experiment. This model assumption has been supported by many real examples including the one in Section 3.6. Recall that configuration variables $\mathbf{x}$ consists of continuous variables $\mathbf{x}_{con}$ and categorical variables $\mathbf{x}_{cat}$. So the second-order model (Wu and Hamada 2000) includes the main effects of $\mathbf{x}_{con}$ and $\mathbf{x}_{cat}$, the interactions among $\mathbf{x}_{cat}$, the cross-products and quadratic terms of $\mathbf{x}_{con}$, and the cross-interactions between $\mathbf{x}_{con}$ and $\mathbf{x}_{cat}$. Other terms can also be added to the model, depending on specific scenarios. For example, if some continuous variable has more than two levels, its cubic or higher order terms can be added. The design D can be generated in SAS/QC (SAS

2005b) by using an optimal design based on some criteria like D-optimality. Owing to the complex nature of this experiment, care needs to be taken in the design construction. The computer experiment under consideration requires convergence of sophisticated algorithms at many reference points for solving complex mathematical equations. Even implemented in professional software packages, this type of experiment sometimes encounters instability problems, only producing erratic temperature responses. Part of the experimental results may be unreliable, differing significantly from the rest and deviating considerably from the underlying data center physics. Which configurations lead to the erratic responses is largely unknown before actually running the experiment. Hence, it is impossible to extract "ill-fated" runs preemptively. A robust approach is taken in the design construction by including some "safety net" runs in addition to those required by the estimability consideration, which will ensure that enough reliable observations are available even in case of partial experimental failure.

Recent advances in CFD make it possible to simulate temperature measurements at many monitor points in a data center computer experiment. To best explore the thermal properties of a data center, the monitor points are uniformly placed at various heights and horizontal positions of different rack rows. Suppose a data center consists of I rack rows, denoted by $R_i, i = 1, \ldots, I$, and R_i has J_i horizontal rack positions, denoted by $P_{ij}, j = 1, \ldots, J_i$. Throughout this work, the monitor points are assumed to be located at K_{ij} equally spaced heights within rack position P_{ij}, denoted by $h_k, k = 1, \ldots, K_{ij}$. Further assume common heights $h_1, \ldots, h_K$ are used for every position of P_{ij}. Figure 12 presents a data center housing three computer racks with 150 monitor points. Throughout, let $\mathbf{z}(\mathbf{x}) = [z_{ijh}(\mathbf{x}), i = 1, \ldots, I, j = 1, \ldots, J_i, k = 1, \ldots, K]^t$ denote the temperature readings taken at the monitor points of all the rack rows under configuration $\mathbf{x}$, where z_{ijh} is the temperature reading at height h and rack position P_{ij}. Similarly, let $\mathbf{z}_1, \ldots, \mathbf{z}_N$ denote the temperature readings under configurations $\mathbf{x}_1, \ldots, \mathbf{x}_N$ of the design D respectively.

In summary, data center computer experiment is often run under many different configuration scenarios, and, for each configuration scenario, temperature responses are taken at a large number of monitor points. The resultant high-dimensional temperature readings

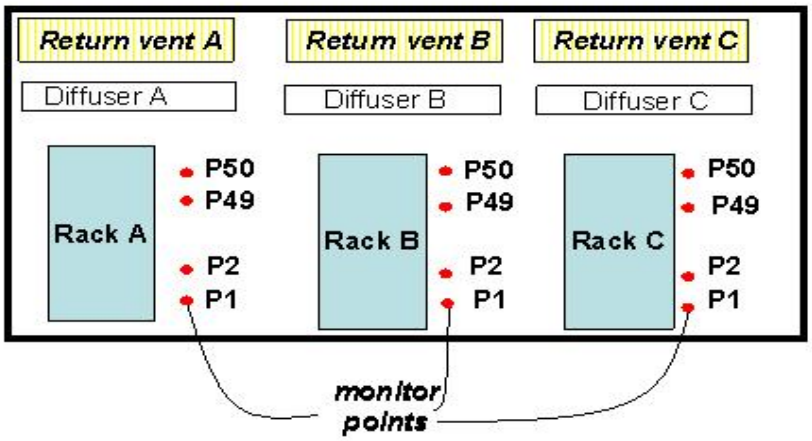

Figure 12: An air-cooling data center system housing three computer racks with 150 monitor points (marked in red).

pose great challenges in building a temperature modeling of a data center. In the next section, a novel three-fold temperature model will be developed.

3.3 Model and motivation

Temperature modeling plays a key role in the data center thermal management. It enables the prediction of the temperature responses at untried locations or under new "what if" configuration scenarios. In this section we present a model structure that is assumed throughout the chapter, and the motivation of this research. To accommodate the high-dimensional responses and configuration variables, an efficient and informative model needs to tackle two major issues:

1. How to summarize the temperature responses by utilizing the underlying data center physics?

2. How to represent various sources of variation in the resultant lower-dimensional physical parameters, and relate them to configuration variables?

To address these issues, we propose a three-fold modeling technique:

Step 1: Obtain a physical summary of temperature responses.

Step 2: Use a measurement model to fit the resultant physical parameters from step 1.

Step 3: Fit a structural model to relate the factors in step 2 to configuration variables.

Details of these steps are discussed below.

Step 1 deals with summarizing the temperature readings based on data center physics. First, the physics suggests that colder air in a data center stays on the bottom while warmer air on the top. Moreover, the air temperature is expected to rise consistently as height increases. As a result, $\mathbf{z}_n$ is assumed to obey the following model

$$z_{ijhn} = m_{ij}(h; \boldsymbol{\theta}_{ijn}) + \epsilon_{ijhn}, \quad n = 1, \ldots, N,$$

where m_{ij} represents a non-decreasing deterministic *temperature trend* that depends on parameters $\boldsymbol{\theta}_{ijn}$, and ϵ_{ijhn} represents a random *temperature fluctuation* whose distribution depends on parameters $\boldsymbol{\phi}_{ijn}$. The nature of heat distribution in a data center renders possible simplifications of the above model. Within each computer rack of a data center,

heat is known to distribute uniformly. Moveover, for every rack row, monitor points, where temperature readings are taken, are evenly placed at equally spaced rack positions. Hence, there is no considerable temperature variation from one position to another within the same rack row, and temperature differences in rack positions can be thereby ignored. These observations lead to the following simplified *row-wise model*

$$z_{ijhn} = m_i(h; \boldsymbol{\theta}_{in}) + \epsilon_{ijhn}, \quad n = 1, \ldots, N, \tag{42}$$

where a common temperature trend m_i is shared between all positions within rack row R_i. Although the proposed method works for general form of m_i, m_i is assumed to be linear for the convenience of presentation. Indeed, this assumption is supported by many applications including the example analyzed in Section 3.6. This assumption leads to the following model

$$z_{ijhn} = \alpha_{in} + \beta_{in} h + \epsilon_{ijhn}, \quad n = 1, \ldots, N, \tag{43}$$

where α_{in} and β_{in} are the intercept and slope of a linear temperature trend and ϵ_{ijhn} is assumed to follow a normal distribution $N(0, \sigma_{in}^2)$. By using this model, temperature measurement $\mathbf{z}_n$ is summarized by physical parameter $\mathbf{y}_n = (\alpha_{1n}, \beta_{1n}, \log \sigma_{1n}, \ldots, \alpha_{In}, \beta_{In}, \log \sigma_{In})$, where the elements $\alpha_{in}, \beta_{in}, \sigma_{1n}$ are associated with the rack row R_i. The dimension of $\mathbf{y}_n$ is much lower than that of $\mathbf{z}_n$, significantly simplifying the subsequent modeling.

Next physical parameters $\mathbf{y}_1, \ldots, \mathbf{y}_N$ are modeled by using a structural equation system (steps 2 and 3). Whereas the *structural-equation method* (SEM) has been developed and widely used in social and behaviorial sciences (Bentler 1995; Bollen 1989; Jöreskog and Sörbom 1996; Wall and Amemiya 2000), the use of this method in physical science and engineering is less frequent. The method is applicable to the present application because it involves multivariate physical parameters and configuration variables. By using a structural equation system, the physical parameters are represented by some *common* unobserved factors (latent variables) in a measurement (factor) model (to be discussed in step 2), and then the factors are in turn related to the observed configuration variables $\mathbf{x}$ in a structural (path) model (to be discussed in step 3). Various sources of variations in physical parameters are then captured in the coefficients of the measurement and structural models. Hence,

inference about the multivariate physical parameters, for different cable racks, can be made using the models involving only the common factors and configuration variables.

Step 2 concerns the fitting of a measurement (factor) model to relate the physical parameter $\mathbf{y}_n$ to a $q \times 1$ factor (latent variable) vector $\mathbf{f}_n$. A general measurement model with an additive measurement error can be written as

$$\mathbf{y}_n = \mathbf{H}_0(\mathbf{f}_n) + \mathbf{u}_n, \quad n = 1, \ldots, N, \tag{44}$$

where $\mathbf{H}_0(\cdot)$ represents an equation mean in a general form, and $\mathbf{u}_n$ represents a measurement error. Without loss of generality, $\mathbf{H}_0$ is assumed to be linear throughout this chapter. This assumption is valid for many practical examples including the one in Section 3.6. Model (44) is not identifiable in the sense that the factor vector $\mathbf{f}_n$ can be transformed without altering the form of the model. Although the model identification issue is not trivial, the errors-in-variables (EV) parametrization (Amemiya and Yalcin 1997; Carrol, Ruppert and Stefanski 1995; Fuller 1987) provides a relatively simple way to represent an identifiable measurement model. Using the EV parametrization, we can write a linear measurement model for $\mathbf{y}_n$ as

$$\begin{pmatrix} \mathbf{y}_{1n} \\ \mathbf{y}_{2n} \end{pmatrix} = \begin{pmatrix} \mathbf{B}_0 \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \mathbf{B} \\ \mathbf{I} \end{pmatrix} \mathbf{f}_n + \mathbf{u}_n, \quad n = 1, \ldots, N. \tag{45}$$

Here $\mathbf{y}_{2n}$ and $\mathbf{y}_{1n}$ are the first q elements of $\mathbf{y}_n$ and their complement respectively, $\mathbf{B}$ is a $(k-q) \times q$ matrix, k is the dimension of $\mathbf{y}_n$, $\mathbf{B}_0$ is a $(k-q) \times 1$ vector and $\mathbf{u}_n$ is a zero-mean measurement error vector independent of $\mathbf{f}_n$. Elements of $\mathbf{f}_n$ are assumed to be correlated to model their interdependencies.

Steps 3 uses a structural (path) model to specify relationships among elements of $\mathbf{f}_n$ and to relate them to the p-dimensional configuration variables $\mathbf{x}$. For a structural model, write

$$\mathbf{f}_n = g(\mathbf{x}_n) + \mathbf{e}_n, \quad n = 1, \ldots, N,$$

where $g(\cdot)$ is a mean function in a general form, and $\mathbf{e}_n$ is an equation error. Although the proposed method applies to general structural models, for convenience g is assumed as a second-order function in this chapter. This assumption is found to work well for the

example in Section 3.6. As discussed in Section 3.2, the configuration variables $\mathbf{x}$ consist of continuous variables $\mathbf{x}_{con}$ and categorical variables $\mathbf{x}_{cat}$. Hence, the second-order model includes: 1. main effects and two-way interactions of $\mathbf{x}_{cat}$; 2. linear, quadratic and cross-products of $\mathbf{x}_{con}$; 3. cross-product terms between $\mathbf{x}_{cat}$ and $\mathbf{x}_{con}$. Let $\mathbf{w}$ denote a $l \times 1$ vector listing all these terms. The factors $\mathbf{f}_n$ can then be expressed as

$$\mathbf{f}_n = \mathbf{a}_0 + \mathbf{A}\mathbf{w}_n + \mathbf{e}_n, \quad n = 1, \ldots, N, \tag{46}$$

where $\mathbf{a}_0$ is a $q \times 1$ vector, and $\mathbf{A}$ is a $q \times l$ matrix.

Collectively, models (43)(45)(46) assumed in steps 1-3 form the following hierarchical model:

$$\begin{aligned} z_{ijhn} &= m_i(h; \boldsymbol{\theta}_{in}) + \epsilon_{ijhn}, \quad i = 1, \ldots, I, j = 1, \ldots, J, h = h_1, \ldots, h_K, \\ \begin{pmatrix} \mathbf{y}_{1n} \\ \mathbf{y}_{2n} \end{pmatrix} &= \begin{pmatrix} \mathbf{B}_0 \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \mathbf{B} \\ \mathbf{I} \end{pmatrix} \mathbf{f}_n + \mathbf{u}_n, \\ \mathbf{f}_n &= \mathbf{a}_0 + \mathbf{A}\mathbf{w}_n + \mathbf{e}_n, \quad n = 1, \ldots, N. \end{aligned} \tag{47}$$

As often used in SEM (Bentler 1995; Bollen 1989; Jöreskog and Sörbom 1996; Wall and Amemiya 2000), $\mathbf{e}_n$ is assumed to be independent of $\mathbf{u}_n$. And $\mathbf{u}_n$'s are assumed to be mutually independent, so are $\mathbf{e}_n$'s. For parameter estimation in (47), we assume that

$$\begin{aligned} \mathbf{u}_n &\sim N(0, \boldsymbol{\Sigma}_{\mathbf{uu}}), \\ \mathbf{e}_n &\sim N(0, \boldsymbol{\Sigma}_{\mathbf{ee}}). \end{aligned} \tag{48}$$

The normality of $\mathbf{u}_n$ and $\mathbf{e}_n$ are considered reasonable for the present application. Elements of $\mathbf{u}_n$ are often assumed to be independent, while elements of $\mathbf{e}_n$ dependent.

The likelihood function of the observations $\mathbf{z}_1, \ldots, \mathbf{z}_N$ is expressed as a high-dimensional integral, which complicates the estimation in (47). Standard maximum likelihood methods cannot be used and alternative procedures are needed. The estimation procedure in Section 3.4 is motivated by one elementary yet important observation: given $\mathbf{y}_1, \ldots, \mathbf{y}_N$, $\mathbf{z}_1, \ldots, \mathbf{z}_N$ follows a *linear model*. By exploiting this fact, a sufficient statistic for model (47) can be found. This sufficient statistic acts as a building block in the estimation procedure.

3.4 Estimation procedure

In this section we develop a two-stage estimation procedure for model (47). For notational simplicity, let $\mathbf{z}_{in} = \{z_{ijkn}, j = 1, \ldots, J_i, k = 1, \ldots, K\}$ denote the part of the temperature measurements $\mathbf{z}_n$ that are taken at the monitor points within rack row R_i and let

$$\bar{z}_{in} = \frac{\sum_{j=1}^{J_i} \sum_{k=1}^{K} z_{ijkn}}{J_i K}.$$

The first stage is concerned with estimating the physical parameters $\mathbf{y}_1, \ldots, \mathbf{y}_N$ by their ordinary least square (OLS) estimators $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$, given as

$$\begin{aligned}
\widehat{\mathbf{y}}_n &= (\widehat{\mathbf{y}}_{1n}, \ldots, \widehat{\mathbf{y}}_{In})^t, \\
\widehat{\mathbf{y}}_{in} &= (\widehat{\alpha}_{in}, \widehat{\beta}_{in}, \log \widehat{\sigma}_{in})^t, \\
\begin{pmatrix} \widehat{\alpha}_{in} \\ \widehat{\beta}_{in} \end{pmatrix} &= (\mathbf{H}_i^t \mathbf{H}_i)^{-1} \mathbf{H}_i^t \mathbf{z}_{in}, \\
\widehat{\sigma}_{in}^2 &= \frac{\sum_{j=1}^{J_i} \sum_{k=1}^{K} (z_{ijkn} - \bar{z}_{in})^2}{J_i K - 1},
\end{aligned} \tag{49}$$

where $\mathbf{H}_i$, the $J_i K \times 2$ regression matrix for the rack-wise model (42), is

$$\mathbf{H}_i = \begin{pmatrix} 1 & h_1 \\ \vdots & \vdots \\ 1 & h_K \\ \vdots & \vdots \\ 1 & h_1 \\ \vdots & \vdots \\ 1 & h_K \end{pmatrix}_{J_i K \times 2}.$$

For $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$, we have the following result.

Proposition 3.1. *The OLS estimator $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$ is a sufficient statistics for model (47).*

Proof. By the independence of $\mathbf{z}_n$, $L(\mathbf{z}_1, \ldots, \mathbf{z}_N) = \prod_{n=1}^{N} L(\mathbf{z}_n)$. Then write $L(\mathbf{z}_n)$ as $\int L(\mathbf{z}_n|\mathbf{y}_n) p(\mathbf{y}_n) d\mathbf{y}_n$. Note that given $\mathbf{y}_n$, $\mathbf{z}_n$ follows a linear model with $\widehat{\mathbf{y}}_n$ as its sufficient statistic. Then, the Factorization theorem implies that $L(\mathbf{z}_n|\mathbf{y}_n) = t(\widehat{\mathbf{y}}_n; \mathbf{z}_n) c(\mathbf{z}_n)$,

where $c(\mathbf{z}_n)$ only depends on $\mathbf{z}_n$ not $\widehat{\mathbf{y}}_n$. Hence, $L(\mathbf{z}_n) = \int t(\widehat{\mathbf{y}}_n; \mathbf{z}_n) c(\mathbf{z}_n) p(\mathbf{y}_n) d\mathbf{y}_n$ and $L(\mathbf{z}_1, \ldots, \mathbf{z}_N) \propto \prod_{n=1}^{N} \int t(\widehat{\mathbf{y}}_n; \mathbf{z}_n) c(\mathbf{z}_n) p(\mathbf{y}_n) d\mathbf{y}_n$. Then, by the Factorization theorem again it follows that $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$ is a sufficient statistics for model (47). □

The second stage deals with fitting a model to $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$. Let $\boldsymbol{\eta}_n = (\boldsymbol{\eta}_{n1}, \ldots, \boldsymbol{\eta}_{nI})^t$ denote the OLS error of $\widehat{\mathbf{y}}_n$. From the fact $\widehat{\mathbf{y}}_n = \mathbf{y}_n + \boldsymbol{\eta}_n$, $\widehat{\mathbf{y}}_n$ can be envisioned to obey the following model:

$$\begin{aligned} \begin{pmatrix} \widehat{\mathbf{y}}_{1n} \\ \widehat{\mathbf{y}}_{2n} \end{pmatrix} &= \begin{pmatrix} \mathbf{B}_0 \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \mathbf{B} \\ \mathbf{I} \end{pmatrix} \mathbf{f}_n + \mathbf{u}_n + \boldsymbol{\eta}_n, \\ \mathbf{f}_n &= \mathbf{a}_0 + \mathbf{A}\mathbf{w}_n + \mathbf{e}_n, \quad n = 1, \ldots, N. \end{aligned} \tag{50}$$

Here $\mathbf{u}_n \sim N(0, \boldsymbol{\Sigma}_{\mathbf{uu}})$ and $\mathbf{e}_n \sim N(0, \boldsymbol{\Sigma}_{\mathbf{ee}})$ as assumed in (48). The description of the above model is complete with the specification of the distribution of $\boldsymbol{\eta}_n$ and the relationship between $(\mathbf{e}_n, \mathbf{u}_n)$ and $\boldsymbol{\eta}_n$. Throughout, for $\boldsymbol{\eta}_n$ we assume the following:

Assumption 1. $\boldsymbol{\eta}_n$ follows a normal distribution $N(\boldsymbol{\mu}_{\boldsymbol{\eta}}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}})$ with mean $\boldsymbol{\mu}_{\boldsymbol{\eta}}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$.

This assumption is considered reasonable for the current application with some justifications to be given in the sequel. Furthermore, the assumption is found to work well for the example in Section 3.6. Note that given $\mathbf{u}_n$ and $\mathbf{e}_n$, $\widehat{\mathbf{y}}_n$ is an unbiased estimator of $\mathbf{y}_n$ as discussed previously, i.e., $E[\boldsymbol{\eta}_n | \mathbf{u}_n, \mathbf{e}_n] = 0$. Hence, by smoothing

$$\boldsymbol{\mu}_{\boldsymbol{\eta}} = E[\boldsymbol{\eta}_n] = 0.$$

The relationship between $(\mathbf{e}_n, \mathbf{u}_n)$ and $\boldsymbol{\eta}_n$ is established in the following proposition.

Proposition 3.2. *$\boldsymbol{\eta}_n$ is independent of $\mathbf{e}_n$ and $\mathbf{u}_n$, i.e,* $\mathrm{Cov}(\boldsymbol{\eta}_n, \mathbf{e}_n) = 0$ *and* $\mathrm{Cov}(\boldsymbol{\eta}_n, \mathbf{u}_n) = 0$.

This can be shown as follows. Note that given $\mathbf{u}_n$ and $\mathbf{e}_n$, $\widehat{\mathbf{y}}_n$ is an unbiased estimator for $\mathbf{y}_n$, i.e., $E[\boldsymbol{\eta}_n | \mathbf{u}_n, \mathbf{e}_n] = 0$. Hence, $E[\boldsymbol{\eta}_n \mathbf{e}_n^t] = E[E[\boldsymbol{\eta}_n \mathbf{e}_n^t | \mathbf{e}_n]] = 0$ and $E[\boldsymbol{\eta}_n \mathbf{u}_n^t] = E[E[\boldsymbol{\eta}_n \mathbf{u}_n^t | \mathbf{u}_n]] = 0$.

Since $\boldsymbol{\mu}_{\boldsymbol{\eta}} = 0$, the unknown parameters in (50) that need to be estimated are $(\boldsymbol{\Omega}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}})$, where $\boldsymbol{\Omega}$ denote $(\mathbf{B}_0, \mathbf{B}, \mathbf{a}_0, \mathbf{A}, \boldsymbol{\Sigma}_{\mathbf{uu}}, \boldsymbol{\Sigma}_{\mathbf{ee}})$, i.e., all the parameters except $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$. Estimation in

(50) seems daunting because (50) is a complicated covariance model involving a large number of unknown parameters and three random disturbances $\mathbf{e}_n$, $\mathbf{u}_n$ and $\boldsymbol{\eta}_n$. It is infeasible to use any available SEM software to compute maximum likelihood estimators (MLEs) of $\boldsymbol{\Omega}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$. For procedural simplicity and computational convenience, a pseudo-likelihood estimation procedure following (Gong and Samaniego 1981) is developed in this chapter. In this procedure, we estimate $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$ by a moment method; then replace $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$ by the moment estimator in the likelihood function of $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$; and finally compute an estimator of $\boldsymbol{\Omega}$ by maximizing the resultant "imputed" likelihood function. Details of this procedure are given below.

We define a moment estimator of $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$ to be

$$\frac{\sum_{n=1}^{N} \mathrm{Cov}[\boldsymbol{\eta}_n|\mathbf{y}_n]}{N}, \tag{51}$$

where $\mathrm{Cov}[\boldsymbol{\eta}_1|\mathbf{y}_1], \ldots, \mathrm{Cov}[\boldsymbol{\eta}_N|\mathbf{y}_N]$ are i.i.d. samples from $\mathrm{Cov}[\boldsymbol{\eta}|\mathbf{y}]$. To calculate $\mathrm{Cov}[\boldsymbol{\eta}_n|\mathbf{y}_n]$, note that

$$\begin{aligned} \begin{pmatrix} \widehat{\alpha}_{in} \\ \widehat{\beta}_{in} \end{pmatrix} |\mathbf{y}_n &\sim N\left(\begin{pmatrix} \alpha_{in} \\ \beta_{in} \end{pmatrix}, (\mathbf{H}_i^t \mathbf{H}_i)^{-1} \sigma_{in}^2 \right), \\ \frac{df_i \widehat{\sigma}_{in}^2}{\sigma_{in}^2} |\mathbf{y}_n &\sim \chi^2(df_i), \end{aligned} \tag{52}$$

where $df_i = J_i K - 2$, and $\widehat{\alpha}_{in}$ and $\widehat{\beta}_{in}$ are independent of $\widehat{\sigma}_{in}^2$. To derive the distribution of $\log \widehat{\sigma}_{in}$, use the Taylor expansion

$$\log \widehat{\sigma}_{in} = \frac{1}{2} \log \widehat{\sigma}_{in}^2 = \frac{1}{2} \log \sigma_{in}^2 + \frac{1}{2\sigma_{in}^2} (\widehat{\sigma}_{in}^2 - \sigma_{in}^2) + o(\widehat{\sigma}_{in}^2 - \sigma_{in}^2). \tag{53}$$

It implies, by ignoring the expansion error, that $\mathrm{Var}(\log \widehat{\sigma}_{in}|\sigma_{in}^2) = \frac{2\sigma_{in}^4}{df_i}$. Further assume

$$\log \widehat{\sigma}_{in}|\mathbf{y}_n \sim N(0, \frac{2\sigma_{in}^4}{df_i}). \tag{54}$$

The conditional normality of $\log \widehat{\sigma}_{in}$ can be justified by the central limit theorem because $J_i K$, the number of monitor points placed within row R_i, is typically large in data center computer experiment. Combining (52) and (54), the conditional distribution $\boldsymbol{\eta}_n|\mathbf{y}_n$ follows a normal distribution as

$$\boldsymbol{\eta}_{in}|\mathbf{y}_n \sim N\left(0, \begin{pmatrix} (\mathbf{H}_i^t \mathbf{H}_i)^{-1} \sigma_{ni}^2 & 0 \\ 0 & \frac{2\sigma_{ni}^4}{df_i} \end{pmatrix}\right), \quad \text{for } i = 1, \ldots, I. \tag{55}$$

Following a similar argument, the distribution of $\mathbf{y}_n$ can be shown to be approximately normal. Therefore, $\boldsymbol{\eta}_n$ in (50) is approximately normally distributed as well, supporting Assumption 1. From (55),

$$\mathrm{Cov}[\boldsymbol{\eta}_n|\mathbf{y}_n] = \begin{pmatrix} (\mathbf{H}_1^t\mathbf{H}_1)^{-1}\sigma_{n1}^2 & 0 & & & & \\ 0 & \frac{2\sigma_1^4}{df_1} & & & & \\ & & \ddots & \ddots & & \\ & & \ddots & \ddots & & \\ & & & & (\mathbf{H}_I^t\mathbf{H}_I)^{-1}\sigma_{nI}^2 & 0 \\ & & & & 0 & \frac{2\sigma_{nI}^4}{df_I} \end{pmatrix}. \tag{56}$$

A complication in using (56) to compute the estimator in (51) is that σ_{in}^2's are not observed. To mitigate this difficulty, we estimate σ_{ni}^2 by the OLS estimator $\widehat{\sigma}_{ni}^2$ defined in (49). Let $\widehat{\mathrm{Cov}}[\boldsymbol{\eta}_n|\mathbf{y}_n]$ denote the resultant estimator of $\mathrm{Cov}[\boldsymbol{\eta}_n|\mathbf{y}_n]$ in (56). Then, the moment estimator in (51) is modified to be

$$\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta}\boldsymbol{\eta}} = \frac{\sum_{n=1}^N \widehat{\mathrm{Cov}}[\boldsymbol{\eta}_n|\mathbf{y}_n]}{N}. \tag{57}$$

We now turn to estimating $\boldsymbol{\Omega}$. In $l(\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N, \boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}, \boldsymbol{\Omega})$, which is the likelihood function of $\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N$, we replace $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$ by $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta}\boldsymbol{\eta}}$ given in (57) and then estimate $\boldsymbol{\Omega}$ by maximizing the function $l(\widehat{\mathbf{y}}_1, \ldots, \widehat{\mathbf{y}}_N, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta}\boldsymbol{\eta}}, \boldsymbol{\Omega})$ after the replacement. Let $\widehat{\boldsymbol{\Omega}}$ denote the resultant estimator for $\boldsymbol{\Omega}$. Calculation of $\widehat{\boldsymbol{\Omega}}$ is made possible through an important observation discussed below. The problem of computing $\widehat{\boldsymbol{\Omega}}$ is equivalent to estimating $\boldsymbol{\Omega}$ in a modified version of model (50) with the value of $\boldsymbol{\Sigma}_{\boldsymbol{\eta}\boldsymbol{\eta}}$ fixed at $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta}\boldsymbol{\eta}}$. The model under this consideration has only two random errors $\mathbf{e}_n$ and $\mathbf{u}_n$ and is a well-studied *standard structural equation model* (Bollen B1989). Recent advances in software development for SEM have made it possible to fast compute the MLEs for this type of model. For the example in Section 3.6, SAS/CALIS (SAS 2005a) is used. The achieved computational convenience of the proposed estimation procedure is significant. It is well known that fitting a non-linear structural equation model in general is no easy task (Wall and Amemiya 2000). By making use of sufficient statistic and pseudo-likelihood method, we have successfully demonstrated a convenient way to estimate a rather complicated hierarchical structural equation model (47). The normality assumptions

of $\mathbf{e}_n$, $\mathbf{u}_n$, and $\boldsymbol{\eta}_n$ imply that once $\widehat{\mathbf{B}}$, $\widehat{\mathbf{A}}$, $\widehat{\boldsymbol{\Sigma}}_{\mathbf{uu}}$ are given, $\widehat{\mathbf{a}}_0$ and $\widehat{\mathbf{B}}_0$ are readily available as

$$\begin{aligned} \widehat{\mathbf{a}}_0 &= \bar{\mathbf{y}}_2 - \widehat{\mathbf{A}}\bar{\mathbf{w}}, \\ \widehat{\mathbf{B}}_0 &= \bar{\mathbf{y}}_1 - \widehat{\mathbf{B}}\widehat{\mathbf{a}}_0 - \widehat{\mathbf{B}}\widehat{\mathbf{A}}\bar{\mathbf{w}}, \end{aligned} \tag{58}$$

where $\bar{\mathbf{y}}_1 = \frac{1}{N}\sum_{n=1}^{N}\mathbf{y}_{1n}$, $\bar{\mathbf{y}}_2 = \frac{1}{N}\sum_{n=1}^{N}\mathbf{y}_{2n}$, and $\bar{\mathbf{w}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{w}_n$. Hence, in the actual implementation, only $\widehat{\mathbf{B}}$, $\widehat{\mathbf{A}}$, $\widehat{\boldsymbol{\Sigma}}_{\mathbf{uu}}$ need to be obtained from a chosen software package.

Next we discuss some large sample properties of $(\widehat{\boldsymbol{\Omega}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta\eta}})$. Throughout, the limit is taken as the number of monitor points $J_iK \to \infty$ for each i and the number of observations $N \to \infty$. Note that given σ_{in}^2, $\widehat{\sigma}_{in}^2 \to \sigma_{in}^2$ in probability as $J_iK \to \infty$ for each i. Hence, given σ_{in}^2, in (53) $\log\widehat{\sigma}_{in} \to \log\sigma_{in}$ in probability as $J_iK \to \infty$. Therefore, for each n, conditional on $\mathbf{y}_n$, $\widehat{\mathrm{Cov}}[\boldsymbol{\eta}_n|\mathbf{y}_n] \to \mathrm{Cov}[\boldsymbol{\eta}_n|\mathbf{y}_n]$ in probability as $J_iK \to \infty$ for each i. Furthermore, by the law of large numbers, $N^{-1}\sum_{n=1}^{N}\mathrm{Cov}[\boldsymbol{\eta}_n|\mathbf{y}_n]$ in (51) is a consistent estimator of $\boldsymbol{\Sigma}_{\boldsymbol{\eta\eta}}$ as $N \to \infty$. Thus, $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta\eta}}$ in (57) is a consistent estimator of $\boldsymbol{\Sigma}_{\boldsymbol{\eta\eta}}$ as $N \to \infty$ and $J_iK \to \infty$ for each i. Then, following (Gong and Samaniego 1981), it is straightforward to establish the consistency of $(\widehat{\boldsymbol{\Omega}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta\eta}})$. Similarly, it is relatively easy to show that the limiting distribution of $(\widehat{\boldsymbol{\Omega}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{\eta\eta}})$ is normal.

The fit of a structural equation model can be accessed by using various methods such as the goodness-of-fit chi-square test. Choosing between two nested competing models can be guided by Chi-square difference tests (Amemiya and Anderson 1990). Since a structural equation system is comprised of two parts, a measurement model and a structural model, selecting an appropriate structural equation model can proceed as follows. First, keeping a full structural model, select a measurement model. Second, using the chosen measurement model, simplify the structural model. This procedure is used in analyzing the example in Section 3.6.

3.5 Prediction, detecting hot spots and determining practical values of configuration variables

Once the model (47) is fitted and associated inferences are made, it can be used as a *surrogate model* to study the data center thermal distribution in lieu of expensive CFD based computer experiment. Use of the surrogate model is essential in the present application.

It is well known that there is huge variation in the thermal properties of a data center in relation to different values of configuration variables. Hence, it is necessary to model a data center under a large variety of configuration scenarios. Use of the expensive computer experiment to explore over such a broad design space of the configuration variables is practically infeasible. The built surrogate model can act as a proxy in the exploration without significantly increasing the computational time. Since the surrogate model is built by using the data from the computer experiment (which is more accurate but time-consuming), it should also have reasonable precision. In this section the surrogate model will be used for predicting the data center thermal distribution at untried locations and "what-if" configuration conditions, for detecting hot spots and for determining practical values of configuration variables to meet some physical and usage requirements.

After model (47) is fitted, it is straightforward to predict the temperature at an untired location in the data center under any configuration, either untried or used in the experiment. Under a given configuration scenario $\mathbf{x}_0$, physical parameters $\mathbf{y} = (\mathbf{y}_2, \mathbf{y}_1)$ can be constructed as

$$\begin{aligned} \begin{pmatrix} \mathbf{y}_1(\mathbf{x}_0) \\ \mathbf{y}_2(\mathbf{x}_0) \end{pmatrix} &= \begin{pmatrix} \widehat{\mathbf{B}}_0 \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \widehat{\mathbf{B}} \\ \mathbf{I} \end{pmatrix} \mathbf{f}(\mathbf{x}_0) + \mathbf{u}, \\ \mathbf{f}(\mathbf{x}_0) &= \widehat{\mathbf{a}}_0 + \widehat{\mathbf{A}}\mathbf{w}(\mathbf{x}_0) + \mathbf{e}. \end{aligned} \tag{59}$$

Note that the OLS error $\boldsymbol{\eta}$ is absent, unlike in (50), because our interest is in predicting the true value of $\mathbf{y}$ not its estimator. Let z_{ih} denote the temperature at height h and row R_i. From (42) and (59), z_{ih} is given by

$$z_{ih}(\mathbf{x}_0) = \alpha_i(\mathbf{x}_0) + h\beta_i(\mathbf{x}_0) + \epsilon_i(\mathbf{x}_0), \quad i = 1, \ldots, I, h \in [h_L, h_U], \tag{60}$$

where $\epsilon_i(\mathbf{x}_0) \sim N(0, \sigma_i^2(\mathbf{x}_0))$, h_L and h_U are the lower and upper limits of a data center, $\alpha_i(\mathbf{x}_0), \beta_i(\mathbf{x}_0)$ and $\sigma_i^2(\mathbf{x}_0)$ are part of $\mathbf{y}(\mathbf{x}_0)$ given in (59). The slope β_i is nonnegative because of the increasing temperature trend discussed in Section 3.3.

Locations in a data center where temperatures are extremely high are called *hot spots* (Schmidt 2003). Detecting hot spots is essential in data center temperature management. Data centers are particularly vulnerable at hot spots because of their extreme high heat

density. The systems can easily malfunction at these hot spots. Furthermore, data centers are intricate inter-connected systems. The malfunctions at hot spots may lead to a system-wide collapse. Finding hot spots via physical or computer experiment is rather difficult because it entails modeling the entire temperature profile of a data center system, whereas the surrogate model approach provides a convenient means. Let M_i denote the maximum temperature in rack row R_i. Since $\beta_i \geq 0$, $M_i(\mathbf{x}_0)$ is given by

$$M_i(\mathbf{x}_0) = \alpha_i(\mathbf{x}_0) + h_U \beta_i(\mathbf{x}_0). \tag{61}$$

Note that this model does not consider the temperature fluctuation $\epsilon(\cdot)$ in (43). To take into account both the temperature trend and fluctuation, we introduce a 95% upper bound on M_i, denoted by L_i. It is defined as $P(M_i \leq L_i) = 95\%$. Under the normality assumption of M_i,

$$L_i(\mathbf{x}_0) = \alpha_i(\mathbf{x}_0) + h_U \beta_i(\mathbf{x}_0) + 1.67\sigma_i(\mathbf{x}_0). \tag{62}$$

This bound has an intuitive explanation: if L_i is controled below a specified level, the maximum temperature in row R_i, with probability more than 95%, will stay below this level. Unfortunately, $L_i(\mathbf{x}_0)$ cannot be used for prediction because of uncertainties in $\alpha_i(\mathbf{x}_0)$, $\beta_i(\mathbf{x}_0)$ and $\sigma_i(\mathbf{x}_0)$. To mitigate this difficulty, we consider the 95% upper limit on L_i, denoted by UL_i. Assuming normality for L_i, UL_i is given by

$$E(L_i) + 1.67\sqrt{\text{var}(L_i)}.$$

The normality of L_i can be justified by the central limit theorem because $J_i K$ is typically large. Owing to the complex error structure in (59), it is difficult to derive the analytical form of UL_i. To expedite computation, we use the following approximation

$$\widehat{UL_i} = \widehat{E}(L_i) + 1.67\sqrt{\widehat{\text{var}}(L_i)}. \tag{63}$$

The value of $\widehat{E}(L_i)$ is given by

$$\widehat{E}(L_i) = \dot{\alpha}_i + \dot{\beta}_i + 1.67, \tag{64}$$

where

$$\begin{aligned}
\dot{\mathbf{y}}_1 &= (\dot{\alpha}_2, \dot{\beta}_2, \log\dot{\sigma}_2, \ldots, \dot{\alpha}_I, \dot{\beta}_I, \log\dot{\sigma}_I)^t, \\
\dot{\mathbf{y}}_2 &= (\dot{\alpha}_1, \dot{\beta}_1, \log\dot{\sigma}_1)^t, \\
\begin{pmatrix} \dot{\mathbf{y}}_1(\mathbf{x}_0) \\ \dot{\mathbf{y}}_2(\mathbf{x}_0) \end{pmatrix} &= \begin{pmatrix} \widehat{\mathbf{B}}_0 \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \widehat{\mathbf{B}} \\ \mathbf{I} \end{pmatrix} \dot{\mathbf{f}}(\mathbf{x}_0), \\
\dot{\mathbf{f}}(\mathbf{x}_0) &= \widehat{\mathbf{a}}_0 + \widehat{\mathbf{A}}\mathbf{w}(\mathbf{x}_0). \qquad (65)
\end{aligned}$$

Using a Taylor expansion

$$e^{\log\sigma_i} \approx e^{\log\dot{\sigma}_i} + e^{\log\dot{\sigma}_i}(\log\sigma_i - \log\dot{\sigma}_i), \qquad (66)$$

$\widehat{\text{var}}(L_i)$ is given as

$$(1, h_U, 1.67e^{\log\dot{\sigma}_i})\mathbf{\Sigma}_i(1, h_U, 1.67e^{\log\dot{\sigma}_i})^t,$$

where $\mathbf{\Sigma}_i$ is the covariance of $(\alpha_i, \beta_i, \log\sigma_i)$ in (59). Locations of hot spots in a data center can be found by comparing the values of $\widehat{UL}$ for different rows.

An important objective in designing a reliable data center is to avoid potential hot spots in the system. The proposed surrogate model approach makes it easy to achieve this objective. Let $\widehat{HU}$ denote the temperature upper bound at the hot spots of the system. Under configuration $\mathbf{x}_0$, $\widehat{HU}(\mathbf{x}_0)$ can be predicted by

$$\widehat{HU}(\mathbf{x}_0) = \max_{i=1,\ldots,I} \widehat{UL}_i(\mathbf{x}_0). \qquad (67)$$

In data center design, physical and usage requirements, denoted by C, also need to be considered. Hence, practical values of the configuration variables $\mathbf{x}$ that can optimize the thermal performance of a data center and meet some physical and usage requirements C can be determined by

$$\mathbf{x}^* = \text{argmin}_{\mathbf{x}\in C}\widehat{HU}(\mathbf{x}). \qquad (68)$$

3.6 A non-raised floor example

In this section the proposed method is illustrated with a non-raised floor example. It models an air-cooled cabinet, implemented in the thermal analysis software Flotherm (Flometrics

2005), for predicting the airflow and heat transfer in the electronic equipments. Each run in this experiment takes several days to complete because the computation in Flotherm entails solving complex thermal dynamic equations. The data center modeled in this experiment has four cable rows R_A, R_B, R_C, and R_D with six rack positions for the first two rows and four for the last two. The rack positions are denoted by $A_0, \ldots, A_5$, $B_0, \ldots, B_5$, $C_2, \ldots, C_5$, and $D_2, \ldots, D_5$ for the four racks respectively. In the experiment, monitor points are placed at five evenly spaced heights for each of the 20 rack positions, resulting in 100-dimensional temperature readings for each configuration run. Table 19 lists nine configuration variables and their levels used in the experiment. In the table, x_1, x_2, x_6, x_7 are continuous variables and x_3, x_4, x_5, x_8 are categorical variables. These variables are among the key factors of the air-cooled cabinets discussed in Section 3.2. More details on the engineering background of this type of data center can be found in (Schmidt 2001a).

Table 19: Configure variables for the non-raised floor example

x_1: Rack temperature rise (C)	10	15	20		
x_2: Rack power (KW)	4	12	22	28	36
x_3: Diffuser height	10 ft	Ceiling			
x_4: Diffuser location/config.	Alt1	Alt2			
x_5: Diffuser angle	0	30			
x_6: Diffuser flow rate (%)	100	80	60		
x_7: Ceiling height (ft)	12	17	22		
x_8: Hot air return vent loc.	Bot-Per	Top-Per	Bot-Par	Top-Par	
x_9: Remove/mixed power	Uniform	Alt-Zero	Alt-Half		

A 148-run design is generated by using the model-based scheme in Section 3.2, implemented in SAS/QC (SAS 2005b). Since the continuous variable x_2 has five levels, added to the second-order model are: 1. cubic term x_2^3; 2. cross-interactions between x_2^3 and x_3, x_4, x_5, and x_8; 3. higher order cross-products $x_1x_2^2$, $x_2^2x_6$, $x_2^2x_7$, $x_1^2x_2$, $x_2x_6^2$, $x_2x_7^2$, $x_1x_2x_6$, $x_1x_2x_7$, $x_2x_6x_7$, $x_1x_2^2x_6$, $x_1x_2^2x_7$ and $x_2^2x_6x_7$; 4. cross-interactions between x_1^2, x_2^2, x_6^2, and x_7^2 and x_3, x_4,x_5, x_8, and x_9. Among the 148 runs, 132 runs are required to estimate all model parameters and the 16 runs are included for estimating the error variance and to accommodate potential experimental failure.

Preliminary analysis is conducted to screen out "ill-fated" runs. For each of the 144

runs, the average value of the 100-dimensional temperature readings is computed. Figure 13 presents the boxplot of the resulting 148 temperature means. Four outliers (i.e., run no. 137, 139, 142 and 146) are identified with means larger than 100 and far above the rest. These four runs are removed before the subsequent analysis.

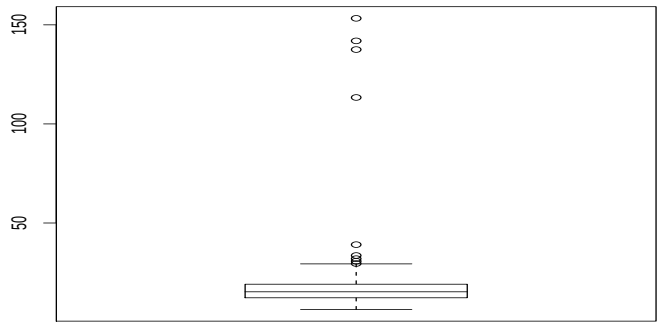

Figure 13: Boxplot of averaged temperatures for the 148 configuration runs, where runs 137, 139, 142 and 146 have temperature above 100.

We now estimate the physical parameters, following stage 1 in the estimation procedure of Section 3.4. Some exploratory analysis is used to aid the selection of an appropriate physical model. Figure 14 presents four 5-dimensional temperature readings taken at randomly selected rack positions A_0, A_1, B_0, and C_5 respectively vs. five different heights attached with fitted linear regression lines. Aimed at exploring temperature trends under different configurations, the temperature readings are selected from four configurations (i.e., run no. 2, 3, 13, 15). It is clear from the figure that this example has a linear temperature trend, which is used for m_i in (42) to produce the physical parameters $\widehat{\mathbf{y}}_n, n = 1, \ldots, 144$. The parameter $\widehat{\mathbf{y}}_{in}$ for the rack row R_i is a 12-dimensional vector $\widehat{\alpha}_i, \widehat{\beta}_i, \log \widehat{\sigma}_i, i = 1, \ldots, 4$.

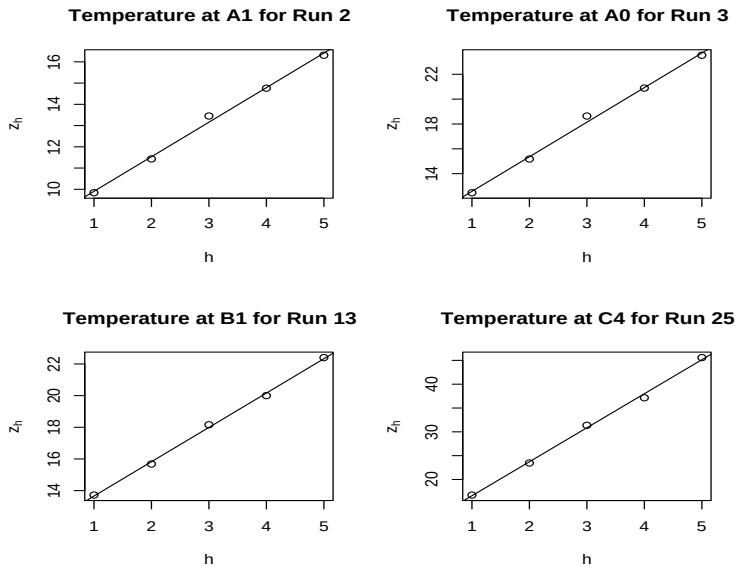

Figure 14: Temperature measurements at five different heights of rack positions A_1, A_0, B_1, and C_4 for runs 2, 3, 13, 15 respectively. Attached straight lines are fitted linear regression functions.

Next we fit a structural equation system to $\widehat{\mathbf{y}}_n$, following stage 2 in the estimation procedure. A three-factor model is chosen on the ground that three factors may be adequate for capturing the relationship among the 12-dimensional physical parameters $\widehat{\mathbf{y}}_n$. The fitted measurement model is given as

$$\begin{pmatrix} \widehat{\alpha}_2 \\ \widehat{\beta}_2 \\ \log\widehat{\sigma}_2 \\ \widehat{\alpha}_3 \\ \widehat{\beta}_3 \\ \log\widehat{\sigma}_3 \\ \widehat{\alpha}_4 \\ \widehat{\beta}_4 \\ \log\widehat{\sigma}_4 \\ \widehat{\alpha}_1 \\ \widehat{\beta}_1 \\ \log\widehat{\sigma}_1 \end{pmatrix} = \begin{pmatrix} b_{01} \\ b_{02} \\ b_{03} \\ b_{04} \\ b_{05} \\ b_{06} \\ b_{07} \\ b_{08} \\ b_{09} \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \\ b_{51} & b_{52} & b_{53} \\ b_{61} & b_{62} & b_{63} \\ b_{71} & b_{72} & b_{73} \\ b_{81} & b_{82} & b_{83} \\ b_{91} & b_{92} & b_{93} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} + \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \\ u_{11} \\ u_{12} \end{pmatrix} + \begin{pmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \\ \eta_5 \\ \eta_6 \\ \eta_7 \\ \eta_8 \\ \eta_9 \\ \eta_{10} \\ \eta_{11} \\ \eta_{12} \end{pmatrix},$$

where

$$\widehat{\mathbf{\Sigma}}_\eta = \begin{pmatrix} 1.638 & -0.447 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.447 & 0.149 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.018 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 7.056 & -1.924 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1.924 & 0.641 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.018 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.629 & -1.535 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1.535 & 0.512 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.028 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3.166 & -0.863 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.863 & 0.288 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.028 \end{pmatrix}$$

is provided by (57).

The three factors have some intuitive explanations: factors f_1 and f_2 are the intercept and the slope of the temperature trend (m_A) respectively and factor f_3 is the logarithm of the standard deviation of the temperature fluctuation (ϵ_A) in R_A. The current structural equation system has a full structure model, consisting of the linear effects of x_1, x_2, x_6 and

x_7, the main effects and interactions of x_3, x_4, x_5, x_8 and x_9, and the cross interactions between x_1, x_2, x_6 and x_7 and x_3, x_4, x_5, x_8 and x_9. The p value associated with the chi-squared goodness-of-fit test obtained by fitting this three-factor system to the 144 observations is 0, suggesting that this model needs to be improved. The χ^2 value is 2526.18 with 740 the degrees of freedom. The factor loading estimates, standard errors, and p values based on the asymptotic normality (suppressing the intercepts) are as follows:

$$
\begin{array}{rcllll}
\widehat{\alpha}_A & \approx & f_1 & & & \\
\widehat{\beta}_A & \approx & f_2 & & & \\
\log \widehat{\sigma}_A & \approx & f_3 & & & \\
\widehat{\alpha}_B & \approx & 0.93f_1 - 0.41f_2 + 0.81f_3 & & & \\
SE & & 0.05 & 0.23 & 0.47 & \\
p \text{ value} & & 0.00 & 0.08 & 0.09 & \\
\widehat{\beta}_B & \approx & 0.01f_1 + 1.06f_2 - 0.20f_3 & & & \\
SE & & 0.01 & 0.07 & 0.15 & \\
p \text{ value} & & 0.40 & 0.00 & 0.15 & \\
\log \widehat{\sigma}_B & \approx & -0.01f_1 - 0.09f_2 + 1.08f_3 & & & \\
SE & & 0.01 & 0.02 & 0.05 & \\
p \text{ value} & & 0.01 & 0.00 & 0.00 & \\
\widehat{\alpha}_C & \approx & 1.02f_1 - 0.46f_2 + 1.27f_3 & & & \\
SE & & 0.04 & 0.22 & 0.453 & \\
p \text{ value} & & 0.00 & 0.04 & 0.01 & \\
\widehat{\beta}_C & \approx & -0.01f_1 + 1.23f_2 - 0.34f_3 & & & \\
SE & & 0.01 & 0.07 & 0.14 & \\
p \text{ value} & & 0.33 & 0.00 & 0.02 & \\
\log \widehat{\sigma}_C & \approx & -0.02f_1 - 0.03f_2 + 0.93f_3 & & & \\
SE & & 0.01 & 0.03 & 0.06 & \\
p \text{ value} & & 0.00 & 0.20 & 0.00 & \\
\widehat{\alpha}_D & \approx & 1.04f_1 - 0.15f_2 + 0.24f_3 & & & \\
SE & & 0.04 & 0.18 & 0.37 & \\
p \text{ value} & & 0.00 & 0.29 & 0.33 & \\
\widehat{\beta}_D & \approx & -0.00f_1 + 1.12f_2 - 0.06f_3 & & & \\
SE & & 0.01 & 0.06 & 0.11 & \\
p \text{ value} & & 0.40 & 0.00 & 0.35 & \\
\log \widehat{\sigma}_D & \approx & -0.01f_1 + 0.02f_2 + 0.97f_3 & & & \\
SE & & 0.01 & 0.03 & 0.05 & \\
p \text{ value} & & 0.01 & 0.33 & 0.00. &
\end{array}
\tag{69}
$$

This model has a clear block-diagonal structure: f_1 in the equations for $\widehat{\alpha}_B$, $\widehat{\alpha}_C$, $\widehat{\alpha}_D$ and f_2 in the equations for $\widehat{\beta}_B$, $\widehat{\beta}_C$, $\widehat{\beta}_D$ and f_3 for the equations of $\widehat{\sigma}_B$, $\widehat{\sigma}_C$, $\widehat{\sigma}_D$ are significant with coefficient close to one. Note that f_1 appears to be insignificant in the three equations for

the slopes $\widehat{\beta}_B$, $\widehat{\beta}_C$, $\widehat{\beta}_D$. Therefore, the three f_1 terms are dropped and the system is refitted using the same procedure, to obtain a simplified model (by suppressing the intercepts)

$$\begin{aligned}
\widehat{\alpha}_A &\approx f_1, \\
\widehat{\beta}_A &\approx f_2, \\
\log \widehat{\sigma}_A &\approx f_3, \\
\widehat{\alpha}_B &\approx 0.97f_1 - 0.54f_2 + 1.06f_3, \\
\widehat{\beta}_B &\approx 1.11f_2 - 0.28f_3, \\
\log \widehat{\sigma}_B &\approx -0.01f_1 - 0.10f_2 + 1.08f_3, \\
\widehat{\alpha}_C &\approx 1.01f_1 - 0.71f_2 + 1.71f_3, \\
\widehat{\beta}_C &\approx 1.33f_2 - 0.50f_3, \\
\log \widehat{\sigma}_C &\approx -0.02f_1 - 0.04f_2 + 0.93f_3, \\
\widehat{\alpha}_D &\approx 1.05f_1 - 0.33f_2 + 0.58f_3, \\
\widehat{\beta}_D &\approx 1.19f_2 - 0.17f_3, \\
\log \widehat{\sigma}_D &\approx -0.02f_1 + 0.02f_2 + 0.95f_3. \qquad (70)
\end{aligned}$$

Fitting this model produces the χ^2 value 2323.64 with 743 degrees of freedom. Comparing with χ^2 value 2526.18 with 740 degrees of freedom for model (69), model (70) is more desirable because it has a better fit (i.e., lower χ^2 value) and simpler structure (i.e., larger degrees of the freedom).

To further simplifying the measurement model, a two-factor measurement model is fitted using the same procedure, which gives the χ^2 value 3311.26 with 828 degrees of freedom. The p value associated with the χ^2 test for choosing between the fitted two-factor model and the three-factor model (70) is zero, leading to rejecting the test; that is, the three-factor model (70) cannot be reduced to a two-factor model.

After a measurement model is selected, we now attempt to simplify the full structural model. We consider the requirements mentioned in Section 3.3 plus additional ones stated as below:

1. The same group of covariates should be chosen for equations for f_1 and f_2 because these

two factors are used in modeling the temperature trend.

2. The equation for f_3 should have more terms than the other two because the temperature fluctuation is known to be more complex than the temperature trend.

The full structural model is simplified by using the cutoff value 1.67 to drop insignificant terms. Terms in the reduced model are listed in Table 20.

Table 20: Terms in the reduced structure model

quadratic	x_2, x_6
cross product	(x_1, x_2), (x_1, x_6), (x_6, x_7)
cross interaction	(x_3, x_2), (x_3, x_6),(x_3, x_7), (x_4, x_1), (x_4, x_2),(x_4, x_6),(x_4, x_7), (x_5, x_1), (x_5, x_2),(x_5, x_6) ,(x_5, x_7),(x_8, x_2), (x_8, x_7),(x_9, x_1),(x_9, x_2),(x_9, x_7)
interaction	(x_3, x_4), (x_3, x_5),(x_4, x_8), (x_4, x_9), (x_5, x_9)

A structural equation system is refitted with the reduced structural model, giving the χ^2 value 1979.62 with 635 degrees of freedom and the measurement model:

$$\begin{aligned}
\widehat{\alpha}_A &\approx f_1, \\
\widehat{\beta}_A &\approx f_2, \\
\log \widehat{\sigma}_A &\approx f_3, \\
\widehat{\alpha}_B &\approx 0.70 + 0.97 f_1 - 0.47 f_2 + 0.78 f_3, \\
\widehat{\beta}_B &\approx 0.00 + 1.08 f_2 - 0.21 f_3, \\
\log \widehat{\sigma}_B &\approx 0.32 - 0.01 f_1 - 0.10 f_2 + 1.07 f_3, \\
\widehat{\alpha}_C &\approx 1.17 + 1.01 f_1 - 0.64 f_2 + 1.50 f_3, \\
\widehat{\beta}_C &\approx 0.03 + 1.31 f_2 - 0.45 f_3, \\
\log \widehat{\sigma}_C &\approx 0.53 - 0.02 f_1 - 0.04 f_2 + 0.94 f_3, \\
\widehat{\alpha}_D &\approx 0.05 + 1.05 f_1 - 0.30 f_2 + 0.45 f_3, \\
\widehat{\beta}_D &\approx 0.16 + 1.18 f_2 - 0.14 f_3, \\
\log \widehat{\sigma}_D &\approx 0.24 - 0.02 f_1 + 0.02 f_2 + 0.96 f_3. \qquad (71)
\end{aligned}$$

To assess the fit of this updated model, it is used to predict physical parameters for the observed 144 configuration runs. Figure 15 plots the physical parameters computed by using OLS vs. their counterparts from the prediction. The points in the figure follow straight lines approximately with no obvious outliers, suggesting a decent model fit. This model will be used as a surrogate for prediction.

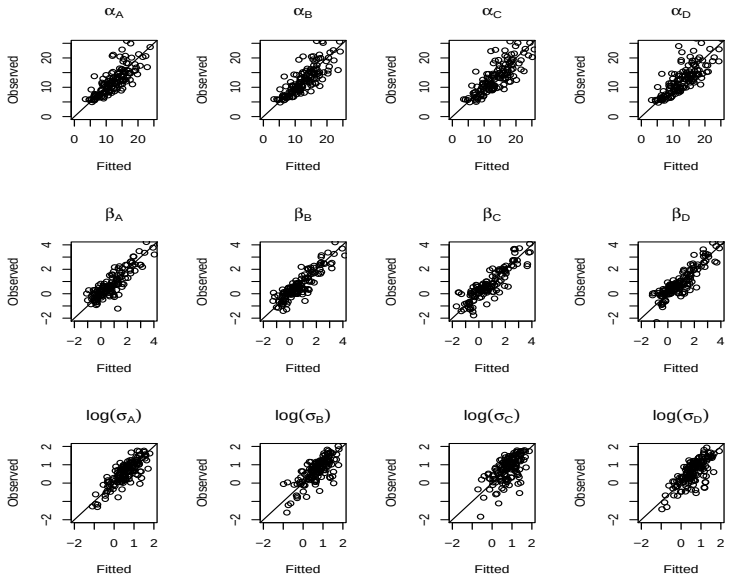

Figure 15: Observed vs. fitted physical parameters for the 144 runs

Finally we predict the temperature of the data center under some untried configuration scenarios. Considering the physical and usage requirements for this example, we choose $x_2 = 28$, $x_4 = 10$ft, $x_7 = 17$ and $x_9 =$ "Uniform". Levels for x_1, x_3, x_5, x_6 and x_8 need to be determined. The values of $\widehat{UL_i}$ in (63) and $\widehat{HU}$ in (67) are computed for every level combination of the five variables. Table 21 lists the scenarios that are associated with the ten lowest values of $\widehat{HU}$. Some interesting observations emerge from this table. First, the row differences in $\widehat{UL}$ are small. Second, the inside rows R_B and R_C tend to have higher UL values than the outside rows R_A and R_D. Third, the values of x_1, x_3 and x_5 for the ten scenarios are, consistently, 10 ft, 30 degree and "Bot-Per". Finally, the effect of x_6 is found to be non-linear with $x_6 = 0.8$ giving the best configuration.

Table 21: The scenarios with ten lowest $\widehat{HU}$ values

x_1 C	x_3 ft	x_5 degree	x_6 %	x_8	$\widehat{UL_A}$	$\widehat{UL_B}$	$\widehat{UL_C}$	$\widehat{UL_D}$	$\widehat{HU}$
10	10	30	0.80	Bot-Per	12.50	12.85	12.04	11.43	**12.85**
10	10	30	0.60	Bot-Per	13.74	14.30	13.23	12.51	14.30
10	10	30	1.00	Bot-Per	14.06	14.45	13.79	13.16	14.45
10	ceiling	30	0.80	Bot-Per	14.05	14.51	13.65	12.96	14.51
10	ceiling	30	1.00	Bot-Per	14.45	14.95	14.26	13.62	14.95
10	10ft	30	0.80	Top-Per	15.46	15.91	15.16	14.45	15.91
10	10ft	30	0.80	Bot-Par	15.53	16.15	15.00	14.19	16.15
10	ceiling	30	0.60	Bot-Per	16.52	17.17	16.03	15.20	17.17
10	10ft	30	0.60	Top-Per	16.73	17.39	16.38	15.58	17.39
10	10ft	30	0.80	Top-Par	16.77	17.49	16.08	15.19	17.49

3.7 Summary

Computer experiment is widely used for studying data center thermal distribution. The high-dimensional responses and configuration variables put forward great challenges in the design and modeling of this type of experiment. In this chapter we have proposed an efficient and informative experimental design for selecting values for the configuration variables, guaranteeing estimability and accommodating potential experimental failure. A three-fold latent variable model is proposed for modeling multivariate temperature responses, incorporating physical structure and various sources of variability. Use of the sufficient statistics makes it feasible to carry out a pseudo-likelihood estimation. A surrogate model has been built to predict practically relevant quantities under a variety of "what-if" conditions in lieu of actually conducting the corresponding expensive experimentation. The surrogate model has been used for determining practical values of the configuration variables of a data center to meet some physical and usage requirements.

Although the proposed method is motivated and developed for the data center application, it is general and can be used in other applications with similar features where the systems produce multivariate measurements and their performance depends on many configuration or design variables.

Extensions of the present work can be made in several directions. First, a latent Gaussian process model (Christensen and Amemiya 2002) can be incorporated into model (47) to better account for the spatial dependence among the temperature responses. Second, instead of using the likelihood-based method proposed here some moment methods similar to those in (Wall and Amemiya 2000) can be developed for the estimation in model (47), which will be more robust to the normality assumptions on $\mathbf{e}_n$ and $\mathbf{u}_n$.

3.8 References

Amemiya, Y. and Anderson, T. W. (1990), "Asymptotic chi-square tests for a large class of factor analysis models," *Annals of Statistics*, **18**, 1453-1463.

Amemiya, Y. and Yalcin, I. (1997), "Model fitting procedures for nonlinear factor analysis using the errors-in-variables parameterization," *Latent Variable Modelling and Applications to Causality*, M. Berkane, ed., Lecture Notes in Statistics 120, Springer Verlag, New York, NY, 195-210.

Apley, D. W. and Shi, J. (2001), "A factor-analysis method for diagnosing variability in multivariate manufacturing processes," *Technometrics*, **43**, 84-95.

Apley, D. W. and Shi, J. (1998), "Diagnosis of multiple fixture faults in panel assembly," *ASME Journal of Manufacturing Science and Engineering*, **120**, 793-801.

Bentler, P. M. (1995), *EQS Strucural Equations Program Manual*, Encino, GA: Multivariate Software.

Bollen, K. A. (1989), *Strucural Equations with Latent Variables*, New York: Wiley.

Basilevsky, A. (1994), *Statistical Factor Analysis and Related Methods*, New York: Wiley.

Ceglarek, D. and Shi, J. (1996), "Fixture failure diagnosis for the autobody assembly using pattern recognition," *ASME Journal of Engineering for Industry*, **118**, 55-66.

Christensen, W. F. and Amemiya, Y. (2002), "Latent variable analysis of multivariate spatial data," *Journal of American Statistical Association*, **97**, 302317.

Ding, Y., Ceglarek, D., Shi, J. (2002), "Fault diagnosis of multistage manufacturing processes by using state space approach," *ASME Journal of Manufacturing Science and Engineering*, **124**, 313-322.

Carrol, R. J., Ruppert, D. and Stefanski, L. A. (1995), *Measurement Error in Nonlinear Models*, London: Chapman and Hall.

Flometrics (2005),"Flotherm: A Fluid Dynamics Based Thermal Analysis Software," http://www.flomerics.com/, data accessed(04/2006).

Fluent (1998), Fluent, Inc., Release 5.5.14 (3d, segregated, laminar).

Fuller, W. A. (1987), *Measurement Error Models*, New York: Wiley.

Gong, G. and Samaniego, F. (1981), "Pseudo maximum likelihood estimation: theory and applications," *Annals of Statistics*, **9**, 861-69.

Lawrence Berkeley National Laboratory (2002), "Data center energy use: truth versus myth," http://www.lbl.gov/Science-Articles/Archive/data-center-energy-myth.html, date accessed (04/2006).

Patel, C., Sharma, R., Bash, C. and Beitelmal, A. (2002), "Thermal considerations in cooling large scale compute density data centers," Itherm Conference, San Diego, California, June 2002, 767 - 776.

Jöreskog, K. G. and Sörbom, D. (1996), *LISREL 8: User's Reference Guide*, Chicago: Scientific Software International.

Johnson, R. A. and Wichern, D. W. (1998), *Applied Multivariate Statistical Analysis*, 4th Edition, Upper Saddle River, NJ: Prentice Hall.

Rolander, N., Rambo, J., Joshi, Y., Allen, J. K., and Mistree, F. (2006), "A survey of robust design with applications to multidisciplinary and multiscale systems," *Journal of Mechanical Design, Special Issue on Risk-Based and Robust Design*, in press.

SAS (2005a), "CALIS: A Covariance Analysis Modular in SAS," http://www.sas.com/, date accessed (04/2006).

SAS (2005b), "QC: A Quality Improvement And Design Of Experiments Modular in SAS," http://www.sas.com/, date accessed (04/2006).

Schmidt, R. (2001), "Effect of data center characteristics on data processing equipment inlet temperatures," *Proceedings of the Pacific Rim/ASME International Electronic Packaging Technical Conference and Exhibition*, Vol 2, Paper IPACK2001-15870, Kauai, Hawaii, 1097-1106.

Schmidt, R. (2001), "Effect of data center characteristics on data processing equipment inlet temperatures," Advances in Electronic Packaging 2001 (Proceedings of IPACK '01, The Pacific Rim/ASME International Electronic Packaging Technical Conference and Exhibition), Vol. 2, Paper IPACK2001-15870, Kauai, Hawaii, July 8-13, 1097-1106.

Schmidt, R. (2003), "Hot spots in data centers," Online Forum of Electrical Cooling, www.electronics-cooling.com.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer.

Fang, K. F., Li, R. Z. and Sudjianto, A. (2005), *Design and Modeling for Computer Experiments*, New York: Chapman & Hall/CRC.

Wall, M. M. and Amemiya, Y. (2000), "Estimation for polynomial structural equation models," *Journal of American Statistical Association*, **95**, 929-940.

Wu, C. F. J., and Hamada, M. (2000), *Experiments, Planning, Analysis and Parameter Design Optimization*, New York: John Wiley & Son.

CHAPTER IV

NESTED SPACE-FILLING DESIGNS FOR MULTIPLE EXPERIMENTS WITH DIFFERENT LEVELS OF ACCURACY

4.1 Introduction

Experimentation to study complex real world systems in engineering and sciences can be conducted at different levels of accuracy or sophistication. Complex mathematical models, implemented in large computer codes, are widely used as a proxy to study the real systems. Doing the corresponding physical experiments would be costly. For example, each physical run of the fluidized bed process in the food industry to coat certain food products with additives discussed in Reese et al. 2004 can take days or even weeks to finish while running the associated computer code only takes minutes per run. Furthermore, a large computer program can often be run at different levels of sophistication with vastly varying computational times. As a result, multiple experiments with various levels of accuracy or fidelity have become popular in practice. These experiments can be physical vs. computer experiments or detailed vs. approximate computer experiments.

Study of such multiple experiments involves two aspects: experimental planning, and analysis and modeling of experimental data. While some headway, e.g., Kennedy and O'Hagan 2000; Qian et al. 2006; Qian and Wu 2005; Reese et al. 2004, has been made to tackle the modeling issue, with exceptions like Qian et al. 2006 little has been done so far to address the planning issue. This problem must be tackled because it is a key to efficiently allocating resources and acquiring information from multiple data sources. It is a new issue in design of experiment because traditional methods (Box, Hunter and Hunter 2005; Wu and Hamada 2000) deal almost exclusively with a single experiment. Hence new methods need to be developed along with new principles.

The purpose of this chapter is to propose a method for constructing nested space-filling designs for multiple experiments with different levels of accuracy. This construction makes use of Galois fields and orthogonal arrays. Multiple design sets generated by the method are guaranteed to have some space-filling property, i.e., each of them is either an orthogonal array-based Latin hypercube design or a randomized orthogonal array. The motivation of this study and construction in a special case are given in Section 4.2. Results for the general case are presented in Section 4.3. Extensions of the proposed procedure to more than two experiments are given in Section 4.4.

4.2 Motivation and construction in a special case

In this section we present the motivation for this study and construction of nested space-filling designs in a special case. Discussed here is a generic situation, where two experiments are available, and one is more accurate but more expensive than the other. The two experiments considered are called *low-accuracy experiment* (LE) and *high-accuracy experiment* (HE). The pair can be physical vs. computer experiments or detailed vs. approximate computer experiments. For the ease of presentation, denote by D_l and D_h the design sets for LE and HE respectively. Without loss of generality, we restrict the design spaces of D_l and D_h to be the unit hypercubes. Construction of D_l and D_h put forward new challenges to design of experiments because conventional methods usually consider a single information source. In the rest of this section we shall discuss new design principles, review an existing method, and propose a new design scheme. Throughout, the construction of D_l and D_h is guided by three new principles:

Principle of economy: The number of points of D_h, denoted by n_h, is smaller than the number of points of D_l, denoted by n_l.

Principle of nested relationship: There is a nested relationship between D_l and D_h, i.e., $D_h \subset D_l$.

Principle of uniformity: The points in D_l and D_h are uniformly distributed over the entire design space.

The principle of economy is concerned with the difference in the computation time of HE and LE. LE is cheaper than HE, so more LE runs can be afforded. The principle of nested relationship makes it easier to model data from HE and LE. This principle implies that, for every point in D_h, results from both LE and HE are available. This part of data can thus be used for modeling the differences between these two experiments, defined as the *adjustment step* in Qian et al. 2006; Qian and Wu 2005, or calibrating the values of unknown parameters in the model of HE (Kennedy and O'Hagan 2001). The principle of uniformity is based on the belief that interesting features of the true models are as likely to be in one part of the design space as the other. Hence, it is desirable to spread the points in D_l and D_h *uniformly* in the design space, which is a reasonable assumption in the absence of any prior idea regarding the models. Such an allocation will be robust to the assumptions on the complex input-and-output relationships HE and LE may exhibit. Throughout this chapter, we refer to the designs that spread the points in a design space uniformly as *space-filling designs*. There are several ways to define the uniformity of the distribution of the points throughout a design space such as distance measures or low-dimensional balances (Santner, Williams and Notz 2003).

These three principles were actually behind a construction method used in (Qian et al. 2006) but the terminology and definitions were not formally given therein. Taking into account of these principles, an immediate approach one may suggest is to optimize some objective function with a total cost constraint. The problem is more complicated than it appears. What objective function should be chosen? It is almost impossible to find a meaningful function to *encompass* the three principles because of their distinctive natures.

Below we shall give a review of *orthogonal array* (OA) and related space-filling designs, serving as a basis for later development. An OA of strength λ, denoted by $OA(m, p, q, \lambda)$, is an $m \times p$ matrix, where each column has q symbols and, for any λ columns, all possible combinations appear equally often in the matrix. Throughout, the q symbols are taken as $1, 2, \ldots, q$ unless otherwise stated. Since OAs with strength 3 or larger tend to have very large run sizes, in this chapter we confine our attention to OAs with strength 2, denoted by the compact notation $OA(m, p, q)$. By definition $m \propto q^2$ for such OAs.

Two space-filling designs (Owen 1992; Tang 1993) using the idea of OAs have been proposed to achieve better uniformity over regular *Latin hypercube designs* (LHDs) (McKay, Beckman and Conover 1979). Tang (1993) proposes *OA-based Latin hypercube designs* (OA-lhds). His construction starts with an $OA(q^2, p, q)$, and then replaces the q positions with symbol t by a random permutation of $(t-1)q+1, \ldots, tq$, for all $t = 1, \ldots, q$. After the replacement procedure is done for all the p columns, denote by $\mathbf{A} = (a_{ij}), i = 1, \ldots, q^2, j = 1, \ldots, p$, the resulting matrix. Suppose that $X_i^j \sim \text{Unif}(0,1]$ or $= 1/2$. The matrix

$$\left(\frac{a_{ij} - X_i^j}{q^2}\right)_{ij}, \ i = 1, \ldots, q^2, \ j = 1, \ldots, p,$$

forms a $q^2 \times p$ OA-lhd with q^2 levels. Owen (1992) introduces *randomized orthogonal arrays* (ROAs). His method can be described as follows. Suppose that $\mathbf{A} = (a_{ij})$ is an $OA(q^2, p, q)$ with its symbols randomized and that $X_i^j \sim \text{Unif}(0,1]$ or $= 1/2$, $i = 1, \ldots, q^2$, $j = 1, \ldots, p$. An ROA is the matrix

$$\left(\frac{a_{ij} - X_i^j}{q}\right)_{ij}, \ i = 1, \ldots, q, \ j = 1, \ldots, p.$$

We now turn to reviewing a two-step procedure used in (Qian et al. 2006) as a motivation for the new method. It proceeds as follows:

Step 1: Construct an OA-lhd for D_l with size n_l;

Step 2: Choose a subset of D_l with size n_h as D_h based on the *maximin distance criterion*, i.e.,

$$D_h = \text{argmax}_D[\min_{\mathbf{x}_1, \mathbf{x}_2 \in D} d(\mathbf{x}_1, \mathbf{x}_2)], \tag{72}$$

where D is any subset of D_l with size n_h.

OA-lhd seems to be a good choice for D_l in this procedure because it can accommodate larger run sizes and enjoys one- and two-dimensional balances (Tang 1993). Guided by the principle of uniformity, step 2 produces a *maximin distance design* (mdd) (Santner, Williams and Notz 2003) for D_h, where no two points are too close to each other.

As an illustration, consider an example with five variables $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5)^t$ taking values in the unit hypercube $(0,1]^5$. This example will be used through the end of this

section. In step 1, $OA(64, 5, 8)$, obtained from Sloane 1994, is used to generate a 64-run OA-lhs for D_l. Figure 16 presents the bivariate projection of the points in D_l. For better exposition, only the projections of the first three variables are plotted. In the figure if we divide domain of each dimension into eight equally spaced intervals (forming 64 reference square bins), for each pair of variables, the 64 points are uniformly distributed in each of the 64 reference square bins. Also, if we divide each of these bins in the figure into 8 equally spaced new bins with smaller size (64 new bins in each dimension), we find that each individual variable in each dimension has a nearly uniform distribution in the 64 bins. In step 2, a 16-run mdd is chosen for D_h. Computing an mdd in general is a difficult combinational optimization problem. For this example, D_h of size 16 is computed by using a *simulated annealing algorithm* (Belisle 1992) with 2000 iterations. Figure 17 presents the bivariate projection of D_h for the first three variables, suggesting that D_h is far from being space-filling. If the domain of each dimension is equally divided into four components (thereby forming 16 reference square bins), in each of the six plots in the figure there are more than one point in some square bins while no point in others. This example indicates two major drawbacks of the two-step method. First, search for an mdd in step 2 entails intensive computation, especially for problems with many variables. Second, the resulting mdd from step 2 may not be space-filling.

To mitigate these drawbacks, we propose a new construction method that can generate nested space-filling designs for D_h and D_l and requires limited computation. In this section the procedure is illustrated with the aforementioned five-dimensional example with the general results deferred to the next section. Recall that, for this example the two-step procedure (72) generates a 64-run OA-lhd for D_l and a 16-run mdd for D_h. By contrast, we now aim at generating

$$D_l : \text{a 64-run OA-lhd and } D_h : \text{a 16-run design with two-dimensional balance with } D_h \subset D_l. \tag{73}$$

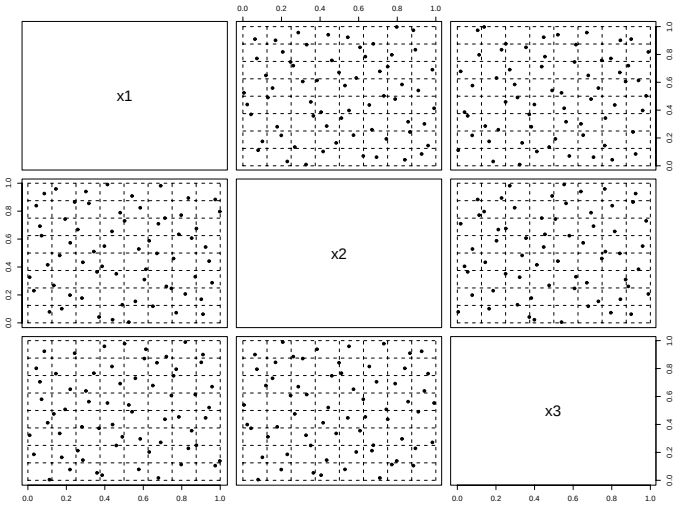

Figure 16: Bivariate projections for x_1, x_2, x_3 of D_l as a 64-run OA-lhs.

Here D_l is chosen the same as before while D_h is different from the previous method. D_h to be generated is space-filling: in its bivariate projections, if the domain of each dimension is equally divided into four components (thereby forming 16 reference square bins) there will be exactly one point of D_h in each of the 16 square bins. Note that D_h and D_l under consideration have *different levels of uniformity* in terms of the lengths of the reference bins in their bivariate projections. This difference is considered reasonable because D_h and D_l have different run sizes.

Two questions associated with (73) need to be answered:

1. Does such a nested structure exist?

2. How to construct?

One possible approach to tackling these questions is through a modified version of the two-step procedure in (72) in which the maximin distance criterion in step 2 is replaced by some criterion aimed at quantifying the two-dimension stratification required for D_h. This approach has several potential problems. First, it is incapable of addressing the existence issue unless an exhaustive search is conducted. Second, finding a criterion to quantify the two-dimension stratification is no easy task. Third, search in the second step often involves

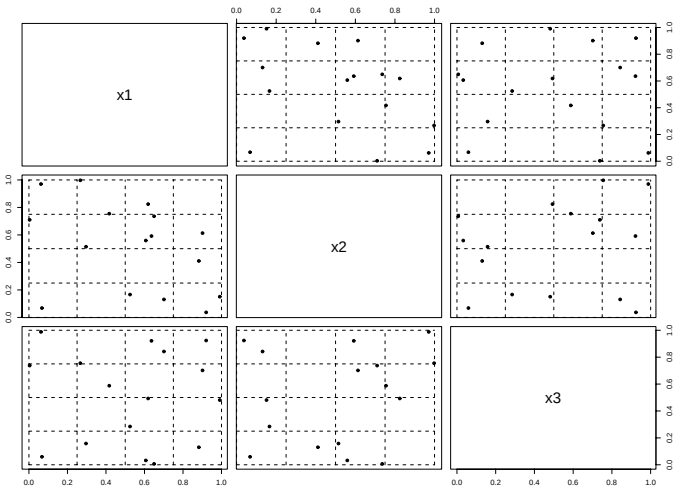

Figure 17: Bivariate projections for x_1, x_2, x_3 of D_h as a 16-run maximin distance design.

intensive computation.

In view of these difficulties, we now discuss a method to better address the existence and construction issues of the designs in (73) by utilizing some algebraical techniques. Note that by definition the existence and construction of D_l and D_h in (73) is mathematically equivalent to that of two underlying OAs,

$$OA(64, 5, 8) \text{ and } OA(16, 5, 4). \tag{74}$$

Hence we shall give a constructive proof to show the existence of these two OAs first, and then use them to construct the required space-filling designs. Construction of the two OAs is complicated by their nested structure, different sizes and numbers of levels. In this chapter we propose an efficient method based on the use of Galois fields. Throughout consider the Galois field $GF(p^w)$, where p is a prime number and w is a positive integer. The elements of the field can be expressed as polynomials of degree $w-1$ with coefficient from $GF(p) = \{0, \ldots, p-1\}$. In the example under consideration the levels of $OA(64, 9, 8)$ and $OA(16, 4, 5)$ are denoted by the elements in $GF(8) = \{0, 1, x, 1{+}x, x^2, 1{+}x^2, x{+}x^2, 1{+}x{+}x^2\}$ and $GF(4) = \{0, 1, u, 1+u\}$ respectively. As a key to achieving the nested structure $D_h \subset D_l$, we establish a correspondence between the elements of $GF(8)$ and $GF(4)$ in the

following manner:

$$\begin{aligned} 0, x^2 &\rightarrow 0, \\ 1, 1+x^2 &\rightarrow 1, \\ x, x+x^2 &\rightarrow u, \\ 1+x, 1+x+x^2 &\rightarrow 1+u. \end{aligned} \tag{75}$$

We start with a larger $OA(64, 9, 8)$, rendering flexibility in choosing *appropriate* columns in the later construction. It is generated by using the *Rao-Hamming construction* (Hedayat, Sloane and Stufken 1999) as follows.

1. Create a 2×9 generator matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & x & 1+x & x^2 & 1+x^2 & x+x^2 & 1+x+x^2 \end{bmatrix},$$

where the columns are all nonzero 2-tuples $(z_1, z_2)^t$ from $GF(8)$ in which the first nonzero z_i is 1.

2. Take all 64 linear combinations of the two rows of this matrix to form an $OA(64, 9, 8)$.

To generate $OA(64, 5, 8)$ and $OA(16, 5, 4)$ based on the constructed $OA(64, 9, 8)$, two issues need to be addressed:

1. How to select columns of $OA(64, 9, 8)$?

2. How to select rows of $OA(64, 9, 8)$?

First, we select columns *1-5* of $OA(64, 9, 8)$ to form a new matrix. By the definition of OA, the resulting matrix is $OA(64, 5, 8)$ as required. Then, select rows *1-4*, *9-12*, *17-20*, *25-28* of $OA(64, 8, 5)$ to form matrix B_h. For B_h, we have the following result.

Lemma 4.1. *With the levels of B_h replaced by the elements of $GF(4)$ according to (75), B_h becomes an $OA(16, 5, 4)$.*

No proof is given here. This lemma is a special case of Lemma 4.3 in Section 4.3, where a complete proof is provided.

Based on the constructed $OA(64, 5, 8)$ and $OA(16, 5, 4)$, it is straightforward to generate the nested designs in (73). First, replace the elements of $OA(64, 5, 8)$ by symbols 1,...,8 according to the following rule between the elements of $GF(8)$ and 1,...,8:

$$\begin{aligned} 0 &\rightarrow 1, \\ 1 &\rightarrow 3, \\ x &\rightarrow 5, \\ 1+x &\rightarrow 7, \\ x^2 &\rightarrow 2, \\ 1+x^2 &\rightarrow 4, \\ x+x^2 &\rightarrow 6, \\ 1+x+x^2 &\rightarrow 8. \end{aligned} \tag{76}$$

Then, replace the eight positions with symbol t of the resultant $OA(64, 5, 8)$ by a random permutation of $(t-1)8+1, \ldots, 8t$, for all $t = 1, \ldots, 8$. Denote by $\mathbf{A} = (a_{ij}), i = 1, \ldots, 64, j = 1, \ldots, 5$, the resulting matrix. Suppose that $X_i^j \sim \text{Unif}(0, 1]$. Let D_l be the matrix $\left(\frac{a_{ij}-X_i^j}{64}\right)_{ij}, i = 1, \ldots, 64, j = 1, \ldots, 5$. Finally, select rows *1-4*, *9-12*, *17-20*, *25-28* of D_l to form D_h. It is evident from the construction that $D_h \subset D_l$ and D_l is a 64-run OA-lhd with 64 levels. For D_h, we have the following result, which follows immediately from Lemma 4.1.

Theorem 4.1. *D_h is a 16-run design with 16 levels and two-dimensional balance.*

Figure 18 depicts the pairwise projections for the first three variables of the points in D_h. It is clear that D_h has the required two-dimensional balance. If the domain of each dimension is equally divided into four components (thereby forming 16 reference square bins), there will be exactly one point of D_h in each of these 16 square bins. Note that this construction can only generate a D_h with two-dimensional balance, not necessarily an OA-lhd with one- and two-dimensional balances. One potential drawback of using an ROA

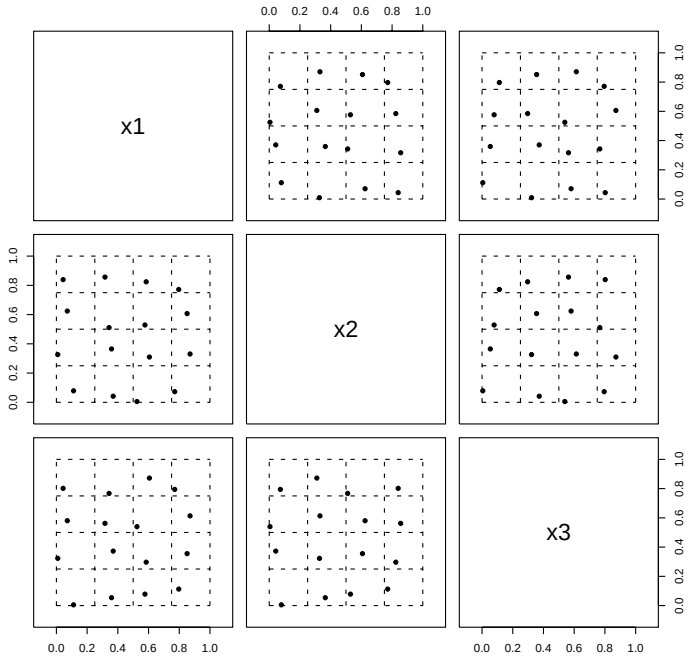

Figure 18: Bivariate projections for x_1, x_2, x_3 of D_h as a 16-run ROA.

is that some of its projected points in one dimension may be identical. This is not the case for the D_h generated by our procedure. As shown in Figure 18, projected points of D_h in one dimension take distinct values.

4.3 General results

In this section we discuss the proposed method for the general case. Without loss of generality, our construction is confined to the class of designs

$$\begin{aligned} D_l: &\quad \text{an } s^{2k} \times (s^{k-l}+1) \text{ OA-lhd}, \\ D_h: &\quad \text{an } s^{2(k-l)} \times (s^{k-l}+1) \text{ design with two-dimensional balance} \end{aligned} \tag{77}$$

where $D_h \subset D_l$ and $k \geq 2$, s is a prime power and $l = 1, 2, 3, \ldots, (k-1)$. Here D_l and D_h have the same number of columns following the assumption that HE and LE use the same group of factors. D_l has more runs than D_h due to the principle of economy. Mathematically, the existence and construction of these nested designs is equivalent to that

of the following two OAs:

$$OA(s^{2k}, s^{k-l}+1, s^k) \text{ and } OA(s^{2(k-l)}, s^{k-l}+1, s^{k-l}). \tag{78}$$

In the rest of this section, we shall first discuss constructing these two OAs and then describe how to use them to generate the designs in (77).

Throughout let $s = p^m$, where p is a prime number and m is a positive integer. The levels of $OA(s^{2k}, s^{(k-l)}+1, s^k)$ and $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$ are denoted by the elements of two Galois fields, $GF(p^{km})$ consisting of

$$a_0 + a_1 x + \ldots + a_{km-1} x^{km-1}, \;\; a_i \in GF(p), \;\; i = 0, \ldots, km-1,$$

and $GF(p^{(k-l)m})$ consisting of

$$a_0 + a_1 u + \ldots, a_{(k-l)m-1} u^{(k-l)m-1}, \;\; a_i \in GF(p), \;\; i = 0, \ldots, (k-l)m-1.$$

Note that distinct symbols x and u are used for the polynomials in the two fields because they are associated with two different OAs. Assume that the elements of the two fields are arranged in *lexicographical* order. Let $\alpha_i, i = 1, \ldots, p^{km}$, denote the elements of $GF(p^{km})$ and $\beta_j, j = 1, \ldots, p^{(k-l)m}$, denote the elements of $GF(p^{(k-l)m})$ in this order. For example if $k = 2$, $p = 3$, $l = 1$, $m = 1$ we consider the Galois fields $GF(3^2)$ and $GF(3^1)$. The elements of $GF(3^2)$ may be exhibited as

$$\begin{aligned}
\alpha_0 &= 0, \\
\alpha_1 &= 1, \\
\alpha_2 &= 2, \\
\alpha_3 &= x, \\
\alpha_4 &= 1 + x, \\
\alpha_5 &= 2 + x, \\
\alpha_6 &= 2x, \\
\alpha_7 &= 1 + 2x, \\
\alpha_8 &= 2 + 2x,
\end{aligned} \tag{79}$$

and the elements of $GF(3^1)$ as

$$\begin{aligned} \beta_0 &= 0, \\ \beta_1 &= 1, \\ \beta_2 &= 2. \end{aligned} \tag{80}$$

As a key to achieving the nested structure $D_h \subset D_l$, we now establish a correspondence between the elements of $GF(p^{km})$ and $GF(p^{(k-l)m})$ in the following manner. The element α_i of $GF(p^{km})$ corresponds to the element

$$\beta_j = a_0 + a_1 u + \cdots + a_{(k-l)m-1} u^{(k-l)m-1} \tag{81}$$

of $GF(p^{(k-l)m})$, where the coefficients of $u^{(k-l)m-1}$ and lower powers of u are the same as the coefficients of the corresponding powers of x in α_i. A similar correspondence is used in Bose and Bush 1952 for constructing completely resolvable arrays. In this correspondence β_j is uniquely determined by α_i as

$$j = i \pmod{p^{(k-l)m}}, \quad 0 \le j < p^{(k-l)m}. \tag{82}$$

Suppose β_j is given. For α_i, each of the coefficients $a_{(k-l)m}, \ldots, a_{km-1}$ can take p possible values. Hence each β_j of $GF(p^{(k-l)m})$ is associated with p^{lm} elements of $GF(p^{km})$. In the example with $k = 2$, $p = 3$, $l = 1$, $m = 1$ the correspondence between the elements of $GF(3^2)$ and $GF(3^1)$ is given by

$$\begin{aligned} \alpha_0, \alpha_3, \alpha_6 &\rightarrow \beta_0, \\ \alpha_1, \alpha_4, \alpha_7 &\rightarrow \beta_1, \\ \alpha_2, \alpha_5, \alpha_8 &\rightarrow \beta_2. \end{aligned} \tag{83}$$

The construction of $OA(s^{2k}, s^{(k-l)}+1, s^k)$ and $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$ proceeds in four steps:

Step 1: Create a $2 \times (s^k + 1)$ generator matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & x & \cdots & 1 + (p-1)x + \cdots + (p-1)x^{km-1} \end{bmatrix},$$

whose columns are all nonzero 2-tuples $(z_1, z_2)^t$ from $GF(p^{km})$ in which the first nonzero z_i is 1.

Step 2: Take all s^{2k} linear combinations of the two rows of this matrix to form an $OA(s^{2k}, s^k + 1, s^k)$.

Step 3: Select the first $s^{(k-l)}+1$ columns of the $OA(s^{2k}, s^k+1, s^k)$ to form an $OA(s^{2k}, s^{(k-l)}+1, s^k)$.

Step 4: Select rows $(i-1)s^k+1, \ldots, (i-1)s^k+s^{k-l}, i = 1, \ldots, s^{k-l}$, of the $OA(s^{2k}, s^{(k-l)}+1, s^k)$, and replace the entries in these rows by using the elements of $GF(p^{(k-l)m})$ according to (81) to form an $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$.

For the ease of verification, denote by B_l and B_h the resulting matrices generated from steps 3 and 4 respectively. The following lemma, taken from Section 3.4 of Hedayat, Sloane and Stufken 1999, is used throughout.

Lemma 4.2. *For prime q, an $OA(q^n, (q^n-1)/(q-1), q)$ exists whenever $n \geq 2$.*

A proof of this lemma is given in (Hedayat, Sloane and Stufken) through the Rao-Hamming constructions. For B_l and B_h, we have the following result.

Lemma 4.3. *(i) B_l is an $OA(s^{2k}, s^{(k-l)}+1, s^k)$.*

(ii) B_h is an $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$.

Proof. (i). Let $q = s^k$ and $n = 2$. Then the design parameters of $OA(s^{2k}, s^{(k-l)}+1, s^k)$ satisfy the condition of Lemma 4.2. Furthermore, steps 1 and 2 comprise exactly the third Rao-Hamming construction used in (Hedayat, Sloane and Stufken) to prove Lemma 4.2. Therefore, the resultant matrix from Step 2 is an $OA(s^{2k}, s^k+1, s^k)$. Hence B_l is an $OA(s^{2k}, s^{(k-l)}+1, s^k)$ by the definition of OA.

(ii). Note that the first $s^{(k-l)}+1$ columns of the generator for $OA(s^{2k}, s^k+1, s^k)$ are

$$\begin{bmatrix} 0 & 1 & 1 & 1 & \cdots & & 1 \\ 1 & 0 & 1 & x & \cdots & 1+(p-1)x+\cdots+(p-1)x^{(k-l)m-1} \end{bmatrix},$$

and the selected $s^{2(k-l)}$ rows in step 4 are linear combinations of the two rows of this matrix with coefficients taking values $0, 1, x, x+1, \cdots, 1+(p-1)x+\cdots+(p-1)x^{(k-l)m-1}$

of $GF(p^{km})$. Thus, according to the correspondence in (81), the construction of B_h is equivalent to the following two steps:

1. Create a $2 \times s^{(k-l)} + 1$ generator matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 & \cdots & & 1 \\ 1 & 0 & 1 & u & \cdots & 1+(p-1)u+\cdots+(p-1)u^{(k-l)m-1} \end{bmatrix},$$

whose columns are all nonzero 2-tuples $(z_1, z_2)^t$ from $GF(p^{(k-l)m})$ in which the first nonzero z_i is 1.

2. Take all linear combinations of the rows of this matrix with respect to $GF(p^{(k-l)m})$ to form B_h.

Let $q = s^{k-l}$ and $n = 2$. Then the design parameters of $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$ satisfy the condition of Lemma 4.2. Moreover, this construction is exactly the third Rao-Hamming construction. It then follows that B_h is an $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$. □

Based on the constructed $OA(s^{2k}, s^{(k-l)}+1, s^k)$ and $OA(s^{2(k-l)}, s^{(k-l)}+1, s^{(k-l)})$, we now proceed to generate the required designs in (77). For $i = 1, \ldots, s^k$, let $\tau_i = (i-1) (\text{mod } s^l)$ and $\gamma_i = [(i-1) - \tau_i]/s^l$. Then arrange $i = 1, \ldots, s^k$ according to the dictionary order of (γ_i, τ_i). Let $c_1, \ldots, c_{s^k}$ denote the i's in this order. For the example with $s = 3$, $k = 2$, $s^k = 9$ and symbols 1,...,9, $c_1, \ldots, c_9$ are given by

$$\begin{aligned} c_1 &= 1 \text{ with } (\gamma_1, \tau_1) = (0,0), \\ c_2 &= 4 \text{ with } (\gamma_2, \tau_2) = (0,1), \\ c_3 &= 7 \text{ with } (\gamma_3, \tau_3) = (0,2), \\ c_4 &= 2 \text{ with } (\gamma_4, \tau_4) = (1,0), \\ c_5 &= 5 \text{ with } (\gamma_5, \tau_5) = (1,1), \\ c_6 &= 8 \text{ with } (\gamma_6, \tau_6) = (1,2), \\ c_7 &= 3 \text{ with } (\gamma_7, \tau_7) = (2,0), \\ c_8 &= 6 \text{ with } (\gamma_8, \tau_8) = (2,1), \\ c_9 &= 9 \text{ with } (\gamma_9, \tau_9) = (2,2). \end{aligned} \tag{84}$$

Next replace the entry α_i of $OA(s^{2k}, s^{(k-l)}+1, s^k)$ according to the following rule

$$\alpha_i = c_{i+1}, \;\; i = 0, \ldots, s^k - 1. \tag{85}$$

After the above replacement is done, further replace the s^k positions with symbol t in $OA(s^{2k}, s^{(k-l)}+1, s^k)$ by a random permutation of $(t-1)s^k+1, \ldots, s^k t$, for all $t = 1, \ldots, s^k$. Denote by $\mathbf{A} = (a_{ij}), i = 1, \ldots, s^k, j = 1, \ldots, s^{k-l}+1$, the resultant matrix. Suppose that $X_i^j \sim \text{Unif}(0, 1]$. Let D_l be the matrix $\left(\frac{a_{ij}-X_i^j}{s^{2k}}\right), i = 1, \ldots, s^k, j = 1, \ldots, s^{k-l}+1$. Finally select rows $(i-1)s^k+1, \ldots, (i-1)s^k+s^{k-l}, i = 1, \ldots, s^{k-l}$, of D_l as suggested by step 4 to form D_h. For the resultant D_h and D_l, we have the following theorem.

Theorem 4.2. *For $k \geq 2$, a prime power s and $l = 1, 2, 3, \ldots, (k-1)$,*

(i) $D_h \subset D_l$;

(ii) D_l is an $s^{2k} \times (s^{k-l}+1)$ OA-lhd with s^{2k} levels;

(iii) D_h is an $s^{2(k-l)} \times (s^{k-l}+1)$ design with $s^{2(k-l)}$ levels and two-dimensional balance.

Proof. (i) and (ii) are evident from the construction. (iii) follows from Lemma 4.3. □

4.4 Extensions to nested space-filling designs for more-than-two experiments

In this section the proposed method is extended to the case of more than two experiments. These experiments can be a combination of physical experiment, detailed computer experiment and approximate computer experiment or Finite Element Analysis (FEA) based computer experiments with different mesh sizes. Suppose that there are $u+1$ such experiments $T_l, T_{h_1} \ldots, T_{h_u}$, arranged in the order of increasing accuracy (i.e., T_{h_u} is the most accurate and expensive). Assume that these experiments use the same group of factors. Let $D_l, D_{h_1}, \ldots, D_{h_u}$ denote the design sets for $T_l, T_{h_1} \ldots, T_{h_u}$, respectively. Denote by $n_l, n_{h_1}, \ldots, n_{h_u}$ the sizes of $D_l, D_{h_1}, \ldots, D_{h_u}$, respectively.

The construction of $D_l, D_{h_1}, \ldots, D_{l_u}$ is guided by the following modified version of the three principles in Section 4.2.

Principle of economy: $n_{h_u} < n_{h_{u-1}} < \cdots < n_l$.

Principle of nested principle: $D_{h_u} \subset D_{h_{u-1}} \cdots \subset D_{h_1} \subset D_l$.

Principle of uniformity: The points in $D_{h_u}, D_{h_{u-1}}, \ldots, D_{h_1}, D_l$ are uniformly distributed over the entire design space.

Without loss of generality, the construction in this section is confined to the following class of designs:

$$\begin{aligned}
D_l: &\quad \text{an } s^{2k}\text{-run OA-lhd with } s^{k-l_u}+1 \text{ columns,} \\
D_{h_1}: &\quad \text{an } s^{2(k-l_1)}\text{-run design with } s^{k-l_u}+1 \text{ columns and two-dimensional balance,} \\
D_{h_2}: &\quad \text{an } s^{2(k-l_2)}\text{-run design with } s^{k-l_u}+1 \text{ columns and two-dimensional balance,} \\
&\quad \vdots \qquad\qquad\qquad\qquad \vdots \\
D_{h_u}: &\quad \text{an } s^{2(k-l_u)}\text{-run design with } s^{k-l_u}+1 \text{ columns and two-dimensional balance,}
\end{aligned} \tag{86}$$

where $k \geq 2$, $s = p^m$ with prime p and positive integer m, $0 < l_1 < l_2 \cdots l_u < k$ and $D_{h_u} \subset D_{h_{u-1}} \cdots \subset D_{h_1} \subset D_l$. Note that the same number of columns are used for all the designs based on the assumption that the same group of factors are chosen for $T_l, T_{h_1} \ldots, T_{h_u}$. The construction of the designs in (86) consists of two stages:

Stage 1: Construct a set of OAs.

Stage 2: Use the constructed OAs to generate the required designs.

Details of these two stages are given below.

By definition, the underlying OAs of the designs in (86) are

$$\begin{gathered}
OA(s^{2(k-l_0)}, s^{k-l_u}+1, s^{k-l_0}), \\
OA(s^{2(k-l_1)}, s^{k-l_u}+1, s^{k-l_1}), \\
OA(s^{2(k-l_2)}, s^{k-l_u}+1, s^{k-l_2}), \\
\vdots \\
OA(s^{2(k-l_u)}, s^{k-l_u}+1, s^{k-l_u}),
\end{gathered} \tag{87}$$

where $l_0 = 0$. For $j = 0, 1, \ldots, u$, the levels of $OA(s^{2(k-l_j)}, s^{k-l_u}+1, s^{k-l_j})$ are denoted by the elements of the Galois field $GF(p^{(k-l_j)m})$, which consists of

$$a_0 + a_1 x + \ldots, a_{(k-l_j)m-1} x^{(k-l_j)m-1}, \;\; a_i \in GF(p), \;\; i = 0, \ldots, (k-l_j)m-1.$$

Further assume that the elements of the field are arranged in lexicographical order. Denote by $\alpha_{j+1,i},\ldots,\alpha_{j+1,p^{(k-l_j)m}}$ the elements of $GF(p^{(k-l_j)m})$ in this order.

For example if $k=3$, $p=2$, $m=1$, $l_1=1$, $l_2=2$ we consider the Galois fields $GF(2^6)$, $GF(2^2)$ and $GF(2^1)$. The elements of $GF(2^6)$ may be exhibited as

$$
\begin{aligned}
\alpha_{1,0}&=0, & \alpha_{1,1}&=1,\\
\alpha_{1,2}&=x, & \alpha_{1,3}&=x+1,\\
\alpha_{1,4}&=x^2, & \alpha_{1,5}&=x^2+1,\\
\alpha_{1,6}&=x^2+x, & \alpha_{1,7}&=x^2+x+1,\\
\alpha_{1,8}&=x^3, & \alpha_{1,9}&=x^3+1,\\
\alpha_{1,10}&=x^3+x, & \alpha_{1,11}&=x^3+x+1,\\
\alpha_{1,12}&=x^3+x^2, & \alpha_{1,13}&=x^3+x^2+1,\\
\alpha_{1,14}&=x^3+x^2+x, & \alpha_{1,15}&=x^3+x^2+x+1,\\
\alpha_{1,16}&=x^4, & \alpha_{1,17}&=x^4+1,\\
\alpha_{1,18}&=x^4+x, & \alpha_{1,19}&=x^4+x+1,\\
\alpha_{1,20}&=x^4+x^2, & \alpha_{1,21}&=x^4+x^2+1,\\
\alpha_{1,22}&=x^4+x^2+x, & \alpha_{1,23}&=x^4+x^2+x+1,\\
\alpha_{1,24}&=x^4+x^3, & \alpha_{1,25}&=x^4+x^3+1,\\
\alpha_{1,26}&=x^4+x^3+x, & \alpha_{1,27}&=x^4+x^3+x+1,\\
\alpha_{1,28}&=x^4+x^3+x^2, & \alpha_{1,29}&=x^4+x^3+x^2+1,\\
\alpha_{1,30}&=x^4+x^3+x^2+x, & \alpha_{1,31}&=x^4+x^3+x^2+x+1,\\
\alpha_{1,32}&=x^5, & \alpha_{1,33}&=x^5+1,\\
\alpha_{1,34}&=x^5+x, & \alpha_{1,35}&=x^5+x+1,\\
\alpha_{1,36}&=x^5+x^2, & \alpha_{1,37}&=x^5+x^2+1,\\
\alpha_{1,38}&=x^5+x^2+x, & \alpha_{1,39}&=x^5+x^2+x+1,\\
\alpha_{1,40}&=x^5+x^3, & \alpha_{1,41}&=x^5+x^3+1,\\
\alpha_{1,42}&=x^5+x^3+x, & \alpha_{1,43}&=x^5+x^3+x+1,\\
\alpha_{1,44}&=x^5+x^3+x^2, & \alpha_{1,45}&=x^5+x^3+x^2+1,\\
\alpha_{1,46}&=x^5+x^3+x^2+x, & \alpha_{1,47}&=x^5+x^3+x^2+x+1,\\
\alpha_{1,48}&=x^5+x^4, & \alpha_{1,49}&=x^5+x^4+1,\\
\alpha_{1,50}&=x^5+x^4+x, & \alpha_{1,51}&=x^5+x^4+x+1,\\
\alpha_{1,52}&=x^5+x^4+x^2, & \alpha_{1,53}&=x^5+x^4+x^2+1,\\
\alpha_{1,54}&=x^5+x^4+x^2+x, & \alpha_{1,55}&=x^5+x^4+x^2+x+1,\\
\alpha_{1,56}&=x^5+x^4+x^3, & \alpha_{1,57}&=x^5+x^4+x^3+1,\\
\alpha_{1,58}&=x^5+x^4+x^3+x, & \alpha_{1,59}&=x^5+x^4+x^3+x+1,\\
\alpha_{1,60}&=x^5+x^4+x^3+x^2, & \alpha_{1,61}&=x^5+x^4+x^3+x^2+1,\\
\alpha_{1,62}&=x^5+x^4+x^3+x^2+x, & \alpha_{1,63}&=x^5+x^4+x^3+x^2+x+1,
\end{aligned}
\tag{88}
$$

the elements of $GF(2^4)$ as

$$\begin{aligned}
\alpha_{2,0} &= 0, & \alpha_{2,1} &= 1, \\
\alpha_{2,2} &= x, & \alpha_{2,3} &= x+1, \\
\alpha_{2,4} &= x^2, & \alpha_{2,5} &= x^2+1, \\
\alpha_{2,6} &= x^2+x, & \alpha_{2,7} &= x^2+x+1, \\
\alpha_{2,8} &= x^3, & \alpha_{2,9} &= x^3+1, \\
\alpha_{2,10} &= x^3+x, & \alpha_{2,11} &= x^3+x+1, \\
\alpha_{2,12} &= x^3+x^2, & \alpha_{2,13} &= x^3+x^2+1, \\
\alpha_{2,14} &= x^3+x^2+x, & \alpha_{2,15} &= x^3+x^2+x+1,
\end{aligned} \tag{89}$$

and the elements of $GF(2^2)$ as

$$\begin{aligned}
\alpha_{3,0} &= 0, \\
\alpha_{3,1} &= 1, \\
\alpha_{3,2} &= x, \\
\alpha_{3,3} &= x+1.
\end{aligned} \tag{90}$$

To achieve the nested structure $D_{h_u} \subset D_{h_{u-1}} \cdots \subset D_{h_1} \subset D_l$, for $j = 0, 1 \ldots, u-1$, a correspondence between the elements of $GF(p^{(k-l_{j+1})m})$ and $GF(p^{(k-l_j)m})$ is established in the following manner. The element α_{j,i_1} of $GF(p^{(k-l_j)m})$ corresponds to the element

$$\alpha_{j+1,i_2} = a_0 + a_1 x + \cdots + a_{(k-l_{j+1})m-1} x^{(k-l_{j+1})m-1} \tag{91}$$

of $GF(p^{(k-l_{j+1})m})$, where the coefficients of $x^{(k-l_{j+1})m}$ and lower powers of x are the same as the coefficients of the corresponding powers of x in α_{j,i_1}. In this correspondence α_{j+1,i_2} is uniquely determined by α_{j,i_1} as

$$i_2 = i_1 \pmod{p^{(k-l_{j+1})m}}, \quad 0 \le i_2 < p^{(k-l_{j+1})m}. \tag{92}$$

This relationship implies that each α_{j,i_2} of $GF(p^{(k-l_{j+1})m})$ is associated with $p^{(l_{j+1}-l_j)m}$ elements of $GF(p^{(k-l_j)m})$. In the example with $k = 3$, $p = 2$, $m = 1$, $l_1 = 1$, $l_2 = 2$ the

correspondence among the elements of $GF(2^6)$, $GF(2^4)$ and $GF(2^2)$ is given by

$$\begin{aligned}
\alpha_{1,0}, \alpha_{1,16}, \alpha_{1,32}, \alpha_{1,48} &\rightarrow \alpha_{2,0} \rightarrow \alpha_{3,0}, \\
\alpha_{1,1}, \alpha_{1,17}, \alpha_{1,33}, \alpha_{1,49} &\rightarrow \alpha_{2,1} \rightarrow \alpha_{3,1}, \\
\alpha_{1,2}, \alpha_{1,18}, \alpha_{1,34}, \alpha_{1,50} &\rightarrow \alpha_{2,2} \rightarrow \alpha_{3,2}, \\
\alpha_{1,3}, \alpha_{1,19}, \alpha_{1,35}, \alpha_{1,51} &\rightarrow \alpha_{2,3} \rightarrow \alpha_{3,3}, \\
\alpha_{1,4}, \alpha_{1,20}, \alpha_{1,36}, \alpha_{1,52} &\rightarrow \alpha_{2,4} \rightarrow \alpha_{3,0}, \\
\alpha_{1,5}, \alpha_{1,21}, \alpha_{1,37}, \alpha_{1,53} &\rightarrow \alpha_{2,5} \rightarrow \alpha_{3,1}, \\
\alpha_{1,6}, \alpha_{1,22}, \alpha_{1,38}, \alpha_{1,54} &\rightarrow \alpha_{2,6} \rightarrow \alpha_{3,2}, \\
\alpha_{1,7}, \alpha_{1,23}, \alpha_{1,39}, \alpha_{1,55} &\rightarrow \alpha_{2,7} \rightarrow \alpha_{3,3}, \\
\alpha_{1,8}, \alpha_{1,24}, \alpha_{1,40}, \alpha_{1,56} &\rightarrow \alpha_{2,8} \rightarrow \alpha_{3,0}, \\
\alpha_{1,9}, \alpha_{1,25}, \alpha_{1,41}, \alpha_{1,57} &\rightarrow \alpha_{2,9} \rightarrow \alpha_{3,1}, \\
\alpha_{1,10}, \alpha_{1,26}, \alpha_{1,42}, \alpha_{1,58} &\rightarrow \alpha_{2,10} \rightarrow \alpha_{3,2}, \\
\alpha_{1,11}, \alpha_{1,27}, \alpha_{1,43}, \alpha_{1,59} &\rightarrow \alpha_{2,11} \rightarrow \alpha_{3,3}, \\
\alpha_{1,12}, \alpha_{1,28}, \alpha_{1,44}, \alpha_{1,60} &\rightarrow \alpha_{2,12} \rightarrow \alpha_{3,0}, \\
\alpha_{1,13}, \alpha_{1,29}, \alpha_{1,45}, \alpha_{1,61} &\rightarrow \alpha_{2,13} \rightarrow \alpha_{3,1}, \\
\alpha_{1,14}, \alpha_{1,30}, \alpha_{1,46}, \alpha_{1,62} &\rightarrow \alpha_{2,14} \rightarrow \alpha_{3,2}, \\
\alpha_{1,15}, \alpha_{1,31}, \alpha_{1,47}, \alpha_{1,63} &\rightarrow \alpha_{2,15} \rightarrow \alpha_{3,3}, \qquad (93)
\end{aligned}$$

The construction of $OA(s^{2(k-l_j)}, s^{k-l_m}+1, s^{k-l_j}), j = 0, 1, \ldots, u$, consists of four steps:

Step 1: Create a $2 \times (s^k + 1)$ generator matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 & \cdots & & 1 \\ 1 & 0 & 1 & x & \cdots & 1+(p-1)x+\cdots+(p-1)x^{km-1} \end{bmatrix},$$

whose columns are all nonzero 2-tuples $(z_1, z_2)^t$ from $GF(p^{km})$ in which the first nonzero z_i is 1.

Step 2: Take all s^{2k} linear combinations of the two rows of this matrix to form an $OA(s^{2k}, s^k + 1, s^k)$.

Step 3: Select the first $s^{k-l_u}+1$ columns of $OA(s^{2k}, s^k+1, s^k)$ to form an $OA(s^{2k}, s^{k-l_u}+1, s^k)$.

Step 4: For $j = 1, \ldots, u$, (i) select rows $(i-1)s^{k-l_{j-1}}+1, \ldots, (i-1)s^{k-l_{j-1}}+s^{k-l_j}, i = 1, \ldots, s^{k-l_j}$, of the $OA(s^{2(k-l_{j-1})}, s^{k-l_u}+1, s^{k-l_{j-1}})$; (ii) replace the entries in these rows by using the elements of $GF(p^{(k-l_j)m})$ according to (91) to form an $OA(s^{2(k-l_j)}, s^{k-l_u}+1, s^{k-l_j})$.

For the ease of verification, denote by B_j the resulting matrix generated from the jth iteration of step 4. Applying similar arguments used in establishing Lemma 4.3, we have the following result for B_j.

Lemma 4.4. *For $j = 1, \ldots, u$, B_j is an $OA(s^{2(k-l_j)}, s^{k-l_u}+1, s^{k-l_j})$.*

With the constructed $OA(s^{2(k-l_j)}, s^{k-l_u}+1, s^{k-l_j}), j = 0, 1, \ldots, u$, we now proceed to generate the designs in (86). It is infeasible to use a simple scheme like (84) for the present case because of its *hierarchical* nested structure $D_{h_u} \subset D_{h_{u-1}} \cdots \subset D_{h_1} \subset D_l$. Below is a more sophisticated procedure used for replacing the elements $\alpha_{1,i}$'s of $GF(s^k)$ by $1, \ldots, s^k$. For the ease of presentation, denote by $\mathrm{sub}(\alpha_{1,i})$ the second subscript of $\alpha_{1,i}$, i.e., $\mathrm{sub}(\alpha_{1,i}) = i$.

Step 1: Assign $\alpha_{1,i}$'s into s^{k-l_u} groups $g_1, \ldots, g_{s^{k-l_u}}$ in the following manner. The elements $\alpha_{1,i}$'s in group g_j satisfy the condition that

$$\mathrm{sub}(\alpha_{1,i}) (\mathrm{mod}\ s^{k-l_u}) = j-1, \quad j = 1, \ldots, s^{k-l_u}. \tag{94}$$

Table 22 lists the elements in the s^{k-l_u} groups.

Step 2: For group g_j, assign its elements into $s^{l_u-l_{u-1}}$ subgroups $g_{j,1}, \ldots, g_{j,s^{l_u-l_{u-1}}}$ according to $s^{l_u-l_{u-1}}$ distinct values of $\mathrm{sub}(\alpha_{1,i}) (\mathrm{mod}\ s^{k-l_{u-1}})$.

Steps 3 to (u-1): Applying similar procedures as in step 2 to further generate subgroups.

Step u: For each group generated in step u-1, assign the elements $\alpha_{1,i}$'s into $s^{l_1-l_0}$ subgroups according to $s^{l_1-l_0}$ distinct values of $\mathrm{sub}(\alpha_{1,i}) (\mathrm{mod}\ s^{k-l_1})$. In this step, s^{k-l_1} subgroups are generated.

Table 23 lists the subgroups generated in steps 1 to u.

Step u+1: Arrange $\alpha_{1,i}$'s in each subgroup generated in step u in increasing order of sub($\alpha_{1,i}$). Let $c_1, \ldots, c_{s^k}$ denote the resultant $\alpha_{1,i}$'s. Then, replace the entry $\alpha_{1,i}$ of $OA(s^{2k}, s^{k-l}+1, s^k)$ according to the following rule

$$c_i = i + 1, \quad i = 0, \ldots, s^k - 1. \tag{95}$$

Table 22: α_i's of $GF(s^k)$ in groups $g_1, \ldots, g_{s^{k-l_u}}$

g_1	$\alpha_{1,0}$ $\alpha_{1,s^{k-l_u}}$ $\alpha_{1,2s^{k-l_u}}$ $\vdots$ $\alpha_{1,(s^{lu}-1)s^{k-l_u}}$
g_2	$\alpha_{1,1}$ $\alpha_{1,s^{k-l_u}+1}$ $\alpha_{1,2s^{k-l_u}+1}$ $\vdots$ $\alpha_{1,(s^{lu}-1)s^{k-l_u}+1}$
$\vdots$ $\vdots$ $\vdots$	$\vdots$ $\vdots$ $\vdots$
$g_{s^{k-l_u}}$	$\alpha_{1,s^{k-l_u}-1}$ $\alpha_{1,s^{k-l_u}+s^{k-l_u}-1}$ $\alpha_{1,2s^{k-l_u}+s^{k-l_u}-1}$ $\vdots$ $\alpha_{1,(s^{lu}-1)s^{k-l_u}+s^{k-l_u}-1}$.

As an illustration, consider an example with $k = 3$, $p = 2$, $m = 1$, $l_1 = 1$, $l_2 = 2$ and Galois field $GF(2^6)$. Table 24 lists the replacement of the elements $GF(2^6)$ by $1, \ldots, 64$.

Table 23: The groups generated in steps 1 to u

Step 1	Step 2	Step 3	$\vdots$	Step u
				$g_{1,1,1,\ldots,1}$
		$g_{1,1,1}$	$\vdots$	$\vdots$
				$g_{1,1,1,\ldots,s^{l_1-l_0}}$
	$g_{1,1}$	$\vdots$	$\vdots$	$\vdots$
				$g_{1,1,s^{l_{u-1}-l_{u-2}},\ldots,1}$
		$g_{1,1,s^{l_{u-1}-l_{u-2}}}$	$\vdots$	$\vdots$
				$g_{1,1,s^{l_{u-1}-l_{u-2}},\ldots,s^{l_1-l_0}}$
g_1	$\vdots$	$\vdots$	$\vdots$	$\vdots$
				$g_{1,s^{l_u-l_{u-1}},1,\ldots,1}$
		$g_{1,s^{l_u-l_{u-1}},1}$	$\vdots$	$\vdots$
				$g_{1,s^{l_u-l_{u-1}},1,\ldots,s^{l_1-l_0}}$
	$g_{1,s^{l_u-l_{u-1}}}$	$\vdots$	$\vdots$	$\vdots$
				$g_{1,s^{l_u-l_{u-1}},s^{l_{u-1}-l_{u-2}},\ldots,1}$
		$g_{1,s^{l_u-l_{u-1}},s^{l_{u-1}-l_{u-2}}}$	$\vdots$	$\vdots$
				$g_{1,s^{l_u-l_{u-1}},s^{l_{u-1}-l_{u-2}},\ldots,s^{l_1-l_0}}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},1,1,\ldots,1}$
		$g_{s^{k-l_u},1,1}$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},1,1,\ldots,s^{l_1-l_0}}$
	$g_{s^{k-l_u},1}$	$\vdots$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},1,s^{l_{u-1}-l_{u-2}},\ldots,1}$
		$g_{s^{k-l_u},1,s^{l_{u-1}-l_{u-2}}}$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},1,s^{l_{u-1}-l_{u-2}},\ldots,s^{l_1-l_0}}$
$g_{s^{k-l_u}}$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},s^{l_u-l_{u-1}},1,\ldots,1}$
		$g_{s^{k-l_u},s^{l_u-l_{u-1}},1}$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},s^{l_u-l_{u-1}},1,\ldots,s^{l_1-l_0}}$
	$g_{s^{k-l_u},s^{l_u-l_{u-1}}}$	$\vdots$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},s^{l_u-l_{u-1}},s^{l_{u-1}-l_{u-2}},\ldots,1}$
		$g_{s^{k-l_u},s^{l_u-l_{u-1}},s^{l_{u-1}-l_{u-2}}}$	$\vdots$	$\vdots$
				$g_{s^{k-l_u},s^{l_u-l_{u-1}},s^{l_{u-1}-l_{u-2}},\ldots,s^{l_1-l_0}}$

Table 24: The replacement of the elements $\alpha_{1,i}$'s in $GF(2^6)$ by $1, \ldots, 64$

Step 1	Step 2	$\alpha_{1,i}$	i
g_1	$g_{1,1}$	$\alpha_{1,0}$	1
		$\alpha_{1,16}$	2
		$\alpha_{1,32}$	3
		$\alpha_{1,48}$	4
	$g_{1,2}$	$\alpha_{1,4}$	5
		$\alpha_{1,20}$	6
		$\alpha_{1,36}$	7
		$\alpha_{1,52}$	8
	$g_{1,3}$	$\alpha_{1,8}$	9
		$\alpha_{1,24}$	10
		$\alpha_{1,40}$	11
		$\alpha_{1,56}$	12
	$g_{1,4}$	$\alpha_{1,12}$	13
		$\alpha_{1,28}$	14
		$\alpha_{1,44}$	15
		$\alpha_{1,60}$	16
g_2	$g_{2,1}$	$\alpha_{1,1}$	17
		$\alpha_{1,17}$	18
		$\alpha_{1,33}$	19
		$\alpha_{1,49}$	20
	$g_{2,2}$	$\alpha_{1,5}$	21
		$\alpha_{1,21}$	22
		$\alpha_{1,37}$	23
		$\alpha_{1,53}$	24
	$g_{2,3}$	$\alpha_{1,9}$	25
		$\alpha_{1,25}$	26
		$\alpha_{1,41}$	27
		$\alpha_{1,57}$	28
	$g_{2,4}$	$\alpha_{1,13}$	29
		$\alpha_{1,29}$	30
		$\alpha_{1,45}$	31
		$\alpha_{1,61}$	32
g_3	$g_{3,1}$	$\alpha_{1,2}$	33
		$\alpha_{1,18}$	34
		$\alpha_{1,34}$	35
		$\alpha_{1,50}$	36
	$g_{3,2}$	$\alpha_{1,6}$	37
		$\alpha_{1,22}$	38
		$\alpha_{1,38}$	39
		$\alpha_{1,54}$	40
	$g_{3,3}$	$\alpha_{1,10}$	41
		$\alpha_{1,26}$	42
		$\alpha_{1,42}$	43
		$\alpha_{1,58}$	44
	$g_{3,4}$	$\alpha_{1,14}$	45
		$\alpha_{1,30}$	46
		$\alpha_{1,46}$	47
		$\alpha_{1,62}$	48
g_4	$g_{4,1}$	$\alpha_{1,3}$	49
		$\alpha_{1,19}$	50
		$\alpha_{1,35}$	51
		$\alpha_{1,51}$	52
	$g_{4,2}$	$\alpha_{1,7}$	53
		$\alpha_{1,23}$	54
		$\alpha_{1,39}$	55
		$\alpha_{1,55}$	56
	$g_{4,3}$	$\alpha_{1,11}$	57
		$\alpha_{1,27}$	58
		$\alpha_{1,43}$	59
		$\alpha_{1,59}$	60
	$g_{4,4}$	$\alpha_{1,15}$	61
		$\alpha_{1,31}$	62
		$\alpha_{1,47}$	63
		$\alpha_{1,63}$	64

After the above replacement is done, further replace the s^k positions with symbol t in $OA(s^{2k}, s^{k-l_u}+1, s^k)$ by a random permutation of $(t-1)s^k+1, \ldots, s^k t$, for all $t = 1, \ldots, s^k$. Denote by $\mathbf{A} = (a_{ij}), i = 1, \ldots, s^k, j = 1, \ldots, s^{k-l_u}+1$, the resultant matrix. Suppose that $X_i^j \sim \text{Unif}(0,1]$. Let D_l be the matrix $\left(\frac{a_{ij}-X_i^j}{s^{2k}}\right), i = 1, \ldots, s^k, j = 1, \ldots, s^{k-l_u}+1$. Finally, for $j = 1, \ldots, u$, select rows of D_l as suggested by step 4 to form D_{h_j}. For the resultant $D_{h_j}, j = 1, \ldots, u$, and D_l, we have the following result.

Theorem 4.3. *For $k \geq 2$, a prime power s and $0 < l_1 < l_2 \cdots l_u < k$,*

(i) $D_{h_u} \subset D_{h_{u-1}} \cdots \subset D_{h_1} \subset D_l$;

(ii) D_l is an $s^{2k} \times (s^{k-l_u}+1)$ OA-lhd with s^{2k} levels;

(iii) D_{h_j} is an $s^{2(k-l_j)} \times (s^{k-l_u}+1)$ design with $s^{2(k-l_j)}$ levels and two-dimensional balance, for $j = 1, \ldots, u$.

Proof. (i) and (ii) are evident from the construction. (iii) follows from Lemma 4.4. □

4.5 References

Box, G. E. P., Hunter, J. S., and Hunter, W. G. (2005), *Statistics for Experimenters: Design, Innovation, and Discovery*, 2nd Edition, New York: John Wiley & Son.

Bose, R. C., and Bush, K. A. (1952), "Orthogonal arrays of strength two and three," *Annals of Mathematical Statistics*, **23**, 508-524.

Belisle, C. J. P. (1992), "Convergence theorems for a class of simulated annealing algorithms on R^d," *Journal of Applied Probability*, **29**, 885-895.

Hedayat, A. S., Sloane, N. J. A., and Stufken, J. (1999), *Orthogonal Arrays: Theory and Applications*, New York: Springer.

Kennedy, M. C., and O'Hagan (2000), "Predicting the output from a complex computer code when fast approximations are available," *Biometrika*, **87**, 1133-1152.

Kennedy, M. C., and O'Hagan (2001), "Bayesian calibration of computer models," *Journal of the Royal Statistical Society: Series B*, **63**, 425-2001.

McKay, D., Beckman, R. J., and Conover, W. J. (1979), "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, **21**, 239-245.

Owen, A. (1992), "Orthogonal arrays for computer experiments, integration and visualization," *Statistica Sinica*, **2**, 439-452.

Qian, Z., Seepersad, C. C., Joseph, V. R., Allen, J. K., and Wu, C. F. J. (2006), "Building surrogate models based on detailed and approximate simulations," *ASME Journal of Mechanical Design, to appear.*

Qian, Z., and Wu, C. F. J. (2005), "Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments," tentatively accepted by *Technometrics.*

Reese, C. S., Wilson, A. G, Hamada, M. and Martz, H. F., and Ryan, K. J. (2004), "Intergated analysis of computer and physical experiments," *Technometrics*, **46**, 153-164.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer.

Sloane, N. J. A. (2004), *A Library of Orthogonal Arrays*, available at http://www.research.att.com/∼njas/oadir, date accessed(04/2006).

Tang, B. (1993), "Orthogonal array-based latin hypercubes," *Jounral of the American Statistical Association*, **88**, 1392-1397.

Wu, C. F. J., and Hamada, M. (2000), *Experiments, Planning, Analysis and Parameter Design Optimization*, New York: John Wiley & Son.

APPENDIX A

BAYESIAN HIERARCHICAL MODELING FOR INTEGRATING LOW-ACCURACY AND HIGH-ACCURACY EXPERIMENTS

A.1 Proof of (34)

Recall from (33) that

$$p(\boldsymbol{\theta}_3|\mathbf{y}_l,\mathbf{y}_h) \propto \int_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2} p(\boldsymbol{\theta}_3)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\mathbf{y}_l,\mathbf{y}_h|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)d\boldsymbol{\theta}_1 d\boldsymbol{\theta}_2. \tag{A1}$$

Perform the integration in (A1) in the following two steps:

1. Integrate out $\boldsymbol{\beta}_l$, ρ_0 and δ_0;

2. Integrate out σ_l^2 and σ_ρ^2.

After perform the two steps, (A1) can be simplified to an integral involving τ_1 and τ_2 only.

Step 1: Integrate out $\boldsymbol{\beta}_l$, ρ_0 and δ_0.

Perform the integration

$$\begin{aligned}\int_{\boldsymbol{\beta_l},\rho_0,\delta_0} & p(\boldsymbol{\theta}_3)p(\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\mathbf{y}_l,\mathbf{y}_h|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)d\boldsymbol{\beta}_l d\rho_0 d\delta_0 \\ &= p(\boldsymbol{\theta}_3)p(\boldsymbol{\theta}_2)\int_{\boldsymbol{\beta}_l,\rho_0,\delta_0} p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\mathbf{y}_l,\mathbf{y}_h|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)d\boldsymbol{\beta}_l d\rho_0 d\delta_0\end{aligned} \tag{A2}$$

by integrating out $\boldsymbol{\beta}_l$, ρ_0 and δ_0 one by one.

(a) Integrate out $\boldsymbol{\beta}_l$.

$$
\begin{aligned}
&\int_{\boldsymbol{\beta}_l} p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2)p(\mathbf{y}_l,\mathbf{y}_h|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)d\boldsymbol{\beta}_l \\
&= \int_{\boldsymbol{\beta}_l} (\sigma_l^2)^{-\frac{n+k+1}{2}}(\sigma_\rho^2)^{-\frac{n_1+2}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{(\rho_0-u_\rho)^2}{2v_\rho\sigma_\rho^2}-\frac{(\delta_0-u_\delta)^2}{2v_\delta\tau_1\sigma_\rho^2}\} \\
&\quad \cdot\exp\{-\frac{(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^t\mathbf{M}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})}{2\sigma_\rho^2}\} \\
&\quad \cdot\exp\{-\frac{1}{2\sigma_l^2}[\frac{(\boldsymbol{\beta}_l-\mathbf{u}_l)^t(\boldsymbol{\beta}_l-\mathbf{u}_l)}{v_l}+(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)^t\mathbf{R}_l^{-1}(\mathbf{y}_l-\mathbf{F}_l\boldsymbol{\beta}_l)]\}d\boldsymbol{\beta}_l, \\
&= (\sigma_l^2)^{-\frac{n+k+1}{2}}(\sigma_\rho^2)^{-\frac{n_1+2}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{(\rho_0-u_\rho)^2}{2v_\rho\sigma_\rho^2}-\frac{(\delta_0-u_\delta)^2}{2v_\delta\tau_1\sigma_\rho^2}\} \\
&\quad \cdot\exp\{-\frac{(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^t\mathbf{M}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})}{2\sigma_\rho^2}\} \\
&\quad \cdot\int_{\boldsymbol{\beta}_l}\exp\{-\frac{1}{2\sigma_l^2}(\boldsymbol{\beta}_l^t\mathbf{a}_1\boldsymbol{\beta}_l+\boldsymbol{\beta}_l^t\mathbf{b}_1+c_1)\}d\boldsymbol{\beta}_l,
\end{aligned}
$$

where $\mathbf{a}_1 = {v_l}^{-1}\mathbf{I}_{(k+1)\times(k+1)} + \mathbf{F}_l^t\mathbf{R}_l^{-1}\mathbf{F}_l, \mathbf{b}_1 = -2v_l^{-1}\mathbf{u}_l - 2\mathbf{F}_l^t\mathbf{R}_l^{-1}\mathbf{y}_l$ and

$$
\begin{aligned}
&c_1 = {v_l}^{-1}(\mathbf{u}_l^t\mathbf{u}_l)+\mathbf{y}_l^t\mathbf{R}_l^{-1}\mathbf{y}_l \\
\propto\ &(\sigma_l^2)^{-\frac{n}{2}}(\sigma_\rho^2)^{-\frac{n_1+2}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{(\rho_0-u_\rho)^2}{2v_\rho\sigma_\rho^2}-\frac{(\delta_0-u_\delta)^2}{2v_\delta\tau_1\sigma_\rho^2}\} \\
&\cdot\exp\{-\frac{(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})^t\mathbf{M}^{-1}(\mathbf{y}_h-\rho_0\mathbf{y}_{l_1}-\delta_0\mathbf{1}_{n_1})}{2\sigma_\rho^2}-\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8\sigma_l^2}\}.
\end{aligned}
\tag{A3}
$$

(b) Integrate out ρ_0.

$$\begin{aligned}
&\int_{\rho_0} (\sigma_l^2)^{-\frac{n}{2}} (\sigma_\rho^2)^{-\frac{n_1+2}{2}} {\tau_1}^{-\frac{1}{2}} |\mathbf{a}_1 \mathbf{R}_l \mathbf{M}|^{-\frac{1}{2}} \exp\{-\frac{(\rho_0 - u_\rho)^2}{2 v_\rho \sigma_\rho^2} - \frac{(\delta_0 - u_\delta)^2}{2 v_\delta \tau_1 \sigma_\rho^2}\} \\
&\qquad \cdot \exp\{-\frac{(\mathbf{y}_h - \rho_0 \mathbf{y}_{l_1} - \delta_0 \mathbf{1}_{n_1})^t \mathbf{M}^{-1} (\mathbf{y}_h - \rho_0 \mathbf{y}_{l_1} - \delta_0 \mathbf{1}_{n_1})}{2\sigma_\rho^2} - \frac{4c_1 - \mathbf{b}_1^t \mathbf{a}_1^{-1} \mathbf{b}_1}{8\sigma_l^2}\} d\rho_0 \\
= \quad & (\sigma_l^2)^{-\frac{n}{2}} (\sigma_\rho^2)^{-\frac{n_1+2}{2}} {\tau_1}^{-\frac{1}{2}} |\mathbf{a}_1 \mathbf{R}_l \mathbf{M}|^{-\frac{1}{2}} \exp\{-\frac{(\delta_0 - u_\delta)^2}{2 v_\delta \tau_1 \sigma_\rho^2} - \frac{4c_1 - \mathbf{b}_1^t \mathbf{a}_1^{-1} \mathbf{b}_1}{8\sigma_l^2}\} \\
&\qquad \cdot \int_{\rho_0} \exp\{-\frac{(\rho_0 - u_\rho)^2}{2 v_\rho \sigma_\rho^2} - \frac{(\mathbf{y}_h - \rho_0 \mathbf{y}_{l_1} - \delta_0 \mathbf{1}_{n_1})^t \mathbf{M}^{-1} (\mathbf{y}_h - \rho_0 \mathbf{y}_{l_1} - \delta_0 \mathbf{1}_{n_1})}{2\sigma_\rho^2}\} d\rho_0, \\
= \quad & (\sigma_l^2)^{-\frac{n}{2}} (\sigma_\rho^2)^{-\frac{n_1+2}{2}} {\tau_1}^{-\frac{1}{2}} |\mathbf{a}_1 \mathbf{R}_l \mathbf{M}|^{-\frac{1}{2}} \exp\{-\frac{(\delta_0 - u_\delta)^2}{2 v_\delta \tau_1 \sigma_\rho^2} - \frac{4c_1 - \mathbf{b}_1^t \mathbf{a}_1^{-1} \mathbf{b}_1}{8\sigma_l^2}\} \\
&\qquad \cdot \int_{\rho_0} \exp\{-\frac{a_2 \rho_0^2 + b_2 \rho_0 + c_2}{2\sigma_\rho^2}\} d\rho_0,
\end{aligned}$$

where $a_2 = {v_\rho}^{-1} + \mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{y}_{l_1}, b_2 = -2u_\rho {v_\rho}^{-1} - 2\mathbf{y}_{l_1}^t \mathbf{M}^{-1} (\mathbf{y}_h - \delta_0 \mathbf{1}_{n_1})$ and

$$\begin{aligned}
& c_2 = u_\rho^2 {v_\rho}^{-1} + (\mathbf{y}_h - \delta_0 \mathbf{1}_{n_1})^t \mathbf{M}^{-1} (\mathbf{y}_h - \delta_0 \mathbf{1}_{n_1}) \\
\propto \quad & (a_2)^{-\frac{1}{2}} (\sigma_l^2)^{-\frac{n}{2}} (\sigma_\rho^2)^{-\frac{n_1+1}{2}} {\tau_1}^{-\frac{1}{2}} |\mathbf{a}_1 \mathbf{R}_l \mathbf{M}|^{-\frac{1}{2}} \exp\{-\frac{(\delta_0 - u_\delta)^2}{2 v_\delta \tau_1 \sigma_\rho^2} - \frac{4c_1 - \mathbf{b}_1^t \mathbf{a}_1^{-1} \mathbf{b}_1}{8\sigma_l^2}\} \\
& \cdot \exp\{-\frac{t_1 \delta_0^2 + t_2 \delta_0 + t_3}{2 a_2 \sigma_\rho^2}\}, \qquad \text{(A4)}
\end{aligned}$$

where $t_1 = ({v_\rho}^{-1} + \mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{y}_{l_1})(\mathbf{1}_{n_1}^t \mathbf{M}^{-1} \mathbf{1}_{n_1}) - (\mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{1}_{n_1})^2$, $t_2 = -2[({v_\rho}^{-1} + \mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{y}_{l_1})(\mathbf{1}_{n_1}^t \mathbf{M}^{-1} \mathbf{y}_h) - ({u_\rho}{v_\rho}^{-1} + \mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{y}_h)(\mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{1}_{n_1})]$ and

$$t_3 \quad = \quad ({v_\rho}^{-1} + \mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{y}_{l_1})(u_\rho^2 {v_\rho}^{-1} + \mathbf{y}_h^t \mathbf{M}^{-1} \mathbf{y}_h) - ({u_\rho}{v_\rho}^{-1} + \mathbf{y}_{l_1}^t \mathbf{M}^{-1} \mathbf{y}_h)^2.$$

(c) Integrate out δ_0.

$$\begin{aligned}
&\int_{\delta_0} (a_2)^{-\frac{1}{2}}(\sigma_l^2)^{-\frac{n}{2}}(\sigma_\rho^2)^{-\frac{n_1+1}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{(\delta_0-u_\delta)^2}{2v_\delta\tau_1\sigma_\rho^2}-\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8\sigma_l^2}\}\\
&\quad\cdot\exp\{-\frac{t_1\delta_0^2+t_2\delta_0+t_3}{2a_2\sigma_l^2}\}d\delta_0\\
=\;&(a_2)^{-\frac{1}{2}}(\sigma_l^2)^{-\frac{n}{2}}(\sigma_\rho^2)^{-\frac{n_1+1}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8\sigma_l^2}\}\\
&\quad\cdot\int_{\delta_0}\exp\{-\frac{(\delta_0-u_\delta)^2}{2v_\delta\tau_1\sigma_\rho^2}-\frac{t_1\delta_0^2+t_2\delta_0+t_3}{2a_2\sigma_l^2}\}d\delta_0,\\
=\;&(a_2)^{-\frac{1}{2}}(\sigma_l^2)^{-\frac{n}{2}}(\sigma_\rho^2)^{-\frac{n_1+1}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8\sigma_l^2}\}\\
&\quad\cdot\int_{\delta_0}\exp\{-\frac{a_3\delta_0^2+b_3\delta_0+c_3}{2\sigma_\rho^2}\}d\delta_0,
\end{aligned}$$

where $a_3=(v_\delta\tau_1)^{-1}+t_1a_2{}^{-1}, b_3=-2u_\delta(v_\delta\tau_1)^{-1}+t_2a_2{}^{-1}$ and

$$\begin{aligned}
&c_3=u_\delta^2(v_\delta\tau_1)^{-1}+t_3a_2{}^{-1}\\
\propto\;&(a_2a_3)^{-\frac{1}{2}}(\sigma_l^2)^{-\frac{n}{2}}(\sigma_\rho^2)^{-\frac{n_1}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8\sigma_l^2}-\frac{4a_3c_3-b_3^2}{8\sigma_\rho^2a_3}\}.
\end{aligned}\tag{A5}$$

Step 2: Integrate out σ_l^2 and σ_ρ^2.

$$\begin{aligned}
&\int_{\sigma_l^2,\sigma_\rho^2} p(\boldsymbol{\theta}_2)(a_2a_3)^{-\frac{1}{2}}(\sigma_l^2)^{-\frac{n}{2}}(\sigma_\rho^2)^{-\frac{n_1}{2}}\tau_1^{-\frac{1}{2}}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}\exp\{-\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8\sigma_l^2}-\frac{4a_3c_3-b_3^2}{8\sigma_\rho^2a_3}\}d\sigma_l^2d\sigma_\rho^2\\
\propto\;&\tau_1^{-(\alpha_\delta+\frac{3}{2})}\tau_2^{-(\alpha_\epsilon+1)}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}(a_2a_3)^{-\frac{1}{2}}\\
&\quad\cdot\int_{\sigma_l^2}(\sigma_l^2)^{-(\alpha_l+\frac{n}{2}+1)}\exp\{-\frac{\gamma_l+\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8}}{\sigma_l^2}\}d\sigma_l^2\\
&\quad\cdot\int_{\sigma_\rho^2}(\sigma_\rho^2)^{-(\alpha_\rho+\alpha_\delta+\alpha_\epsilon+\frac{n_1}{2}+1)}\exp\{-\frac{\gamma_\rho+\frac{\gamma_\delta}{\tau_1}+\frac{\gamma_\epsilon}{\tau_2}+\frac{4a_3c_3-b_3^2}{8a_3}}{\sigma_\rho^2}\}d\sigma_\rho^2,\\
\propto\;&\tau_1^{-(\alpha_\delta+\frac{3}{2})}\tau_2^{-(\alpha_\epsilon+1)}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}(a_2a_3)^{-\frac{1}{2}}(\gamma_l+\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8})^{-(\alpha_l+\frac{n}{2})}\\
&\quad\cdot(\gamma_\rho+\frac{\gamma_\delta}{\tau_1}+\frac{\gamma_\epsilon}{\tau_2}+\frac{4a_3c_3-b_3^2}{8a_3})^{-(\alpha_\rho+\alpha_\delta+\alpha_\epsilon+\frac{n_1}{2})}.
\end{aligned}\tag{A6}$$

Therefore, (A1) is proportional to

$$\begin{aligned}
&p(\boldsymbol{\theta}_3)\int_{\tau_1,\tau_2}\tau_1^{-(\alpha_\delta+\frac{3}{2})}\tau_2^{-(\alpha_\epsilon+1)}|\mathbf{a}_1\mathbf{R}_l\mathbf{M}|^{-\frac{1}{2}}(a_2a_3)^{-\frac{1}{2}}(\gamma_l+\frac{4c_1-\mathbf{b}_1^t\mathbf{a}_1^{-1}\mathbf{b}_1}{8})^{-(\alpha_l+\frac{n}{2})}\\
&\quad\cdot(\gamma_\rho+\frac{\gamma_\delta}{\tau_1}+\frac{\gamma_\epsilon}{\tau_2}+\frac{4a_3c_3-b_3^2}{8a_3})^{-(\alpha_\rho+\alpha_\delta+\alpha_\epsilon+\frac{n_1}{2})}d\tau_1d\tau_2.
\end{aligned}\tag{A7}$$

VITA

Zhiguang Qian was born in Nantong, a coastal city in the Yangtze Delta of China. He received his B.S. degree in Statistics from East China Normal University in 2000. He obtained a M.S. degree in Statistics from the University of Michigan in 2003. He will receive a Ph.D. degree in 2006 from Georgia Institute of Technology. In August 2006, he will join the Department of Statistics at the University of Wisconsin-Madison as an assistant professor.