Optimal Design of Experiments for Functional Responses

by

Moein Saleh

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2015 by the
Graduate Supervisory Committee:

Rong Pan, Chair
Douglas C. Montgomery
George Runger
Ming-Hung Kao

ARIZONA STATE UNIVERSITY

December 2015

ABSTRACT

Functional or dynamic responses are prevalent in experiments in the fields of engineering, medicine, and the sciences, but proposals for optimal designs are still sparse for this type of response. Experiments with dynamic responses result in multiple responses taken over a spectrum variable, so the design matrix for a dynamic response have more complicated structures. In the literature, the optimal design problem for some functional responses has been solved using genetic algorithm (GA) and approximate design methods. The goal of this dissertation is to develop fast computer algorithms for calculating exact D-optimal designs.

First, we demonstrated how the traditional exchange methods could be improved to generate a computationally efficient algorithm for finding G-optimal designs. The proposed two-stage algorithm, which is called the cCEA, uses a clustering-based approach to restrict the set of possible candidates for PEA, and then improves the G-efficiency using CEA.

The second major contribution of this dissertation is the development of fast algorithms for constructing D-optimal designs that determine the optimal sequence of stimuli in fMRI studies. The update formula for the determinant of the information matrix was improved by exploiting the sparseness of the information matrix, leading to faster computation times. The proposed algorithm outperforms genetic algorithm with respect to computational efficiency and D-efficiency.

The third contribution is a study of optimal experimental designs for more general functional response models. First, the B-spline system is proposed to be used as the non-parametric smoother of response function and an algorithm is developed to determine D-optimal sampling points of a spectrum variable. Second, we proposed a two-step algorithm for finding the optimal design for both sampling points and experimental settings. In the first step, the matrix of experimental settings is held

fixed while the algorithm optimizes the determinant of the information matrix for a mixed effects model to find the optimal sampling times. In the second step, the optimal sampling times obtained from the first step is held fixed while the algorithm iterates on the information matrix to find the optimal experimental settings. The designs constructed by this approach yield superior performance over other designs found in literature.

To my parents - Mrs. Fatemeh Shariati and Mr. Jaafar Saleh,

who missed us dearly in their empty nest these passed years,

and to all my teachers.

ACKNOWLEDGEMENTS

First, I would like to extend my deepest gratitude to my advisor, Dr. Rong Pan, for his guidance and support during my PhD. His vision, knowledge, and encouragement always inspired me and helped me stay focused and pursue my objectives. I am forever grateful for his generosity.

I would like to extend a very special thanks to Dr. Ming-Hung Kao for providing me with the opportunity to work on a research under his supervision. Without him, the research would not have been as successful or as enjoyable. He was the constant source of support throughout and provided me with immeasurable encouragement and guidence.

Dr. Douglas C. Montgomery is also highly deserving of my gratitude for his extensive support during my PhD. The time and feedback he generously shared with me during the research made it possible to reach the conclusion of the study.

I would like to thank Dr. George Runger for partaking his time and effort as my committee member to help me fulfill the degree requirements.

I am heartily thankful to Dr. Bingrong Jiang, Dr. Glade Erikson, Jon Ruterman, Vincent He, and Anand Narasimhan who provided me with the opportunity to learn and grow in the Discover Financial Services and also continue to work on this research work.

This journey would not have been the same without the immeasurable help and support of my close friends: Tina Pourshams, Mickey Mancenido, Petek Yontay, Nima Tajbakhsh, Hossein Vashani. Thank you all for being there every step of the way.

Last, but not the least, I want to thank my loving family: my mother, Fatemeh Shariati, my father, Jaafar Salehm and my big brother, Mehdi Saleh, for their unconditional love, support, and care. Without them, I would never have been able to complete this, or any other endeavor in life.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

The popularity of optimal experimental designs has been attributed to many non-conventional experimental design problems. In contrast with standard experimental designs, which require standard cubic or spherical design regions and normal/linear models, optimal experiments allow for irregular design regions and non-normal responses. The idea of optimal experimental design can be found in, e.g., Kiefer (1961); Kiefer and Wolfowitz (1959). Kiefer (1959) proposed to optimize functionals of the design matrix, X, which contains the experimental runs or factor settings. These functionals are aptly called alphabetic optimality criteria, an example of which is D-optimality.

So far, research on optimal designs have been largely focused on the static response system, where one experimental run generates one response value. Examples of these experiments are abound in any literature on experimental designs.

Recently, experiments that generate multiple response values that form a response curve or a response profile are beginning to garner more attention. Improvement in measurement technology and data collection practices have ushered interest in designing experiments with functional responses taken over different points of a continuum variable. Functional response experiments are different from multivariate response experiments. In multivariate response experiments, several attributes are measured simultaneously; while in functional response experiments, the same attribute is measured at different points of a spectrum variable. For example, the spectrum variable could be *time* in a longitudinal study.

Functional data can be found in a variety of applications in engineering, medicine, and the sciences. Some examples that motivated this research are presented and briefly discussed.

1. **Designing fMRI experiments**. Functional Magnetic Resonance Imaging (fMRI) is a brain imaging technique for studying human brain functions. In a typical fMRI experiment, a sequence of mental stimuli are presented to a subject. At the same time, an MRI scanner scans the subject's brain to collect brain activity data by measuring Blood-Oxygen-Level Dependent (BOLD). The measurements produced by the scanner produce a response curve over time. This information is used by scientists to detect and monitor the activated region of a subject's brain over the testing period. fMRI procedures are costly and invoke human testing ethical considerations. In this regard, it is necessary to design highly efficient experiments that produce valid and precise statistical inferences about brain activities.

2. **Designing experiments for emission studies.** Advancements in engine design technology target minimizing vehicle fuel consumption and pollutant emissions. For example, Binde *et al.* (2012) studied soot and NO_x emissions of diesel engines through the design of a spatially separated pilot injection. The objective of these experiments usually include determining the optimal position of pilot injector, the start of injection (SOI), and the number of pilot injections. Soot temperature and concentration are recorded over the entire engine cycle using pyrometrics. In this case, the engine cycle is the spectrum variable and the soot temperature and concentration are the response curves. From a modeling standpoint, a challenging aspect of this problem is the dynamic behavior of

the response curves towards the end of the engine cycle. A flexible model incorporating these dynamics is a required input in the design and analysis of engine experiments.

3. **Determining the optimal measurement points of photovoltaic systems.** Reliability engineers are usually interested in studying the degradation behavior of photovoltaic systems when exposed to environmental elements. Modeling the change in system degradation rate helps predict the inflection point in the degradation path, an indication of a hard failure. This degradation path can be practically modeled as functional data, with time as the continuum. Measuring photovoltaic systems, however, is complicated and time-consuming so taking frequent measurements is impractical. The experimental design problem for this case is in determining the sampling frequency that will still yield good models for prediction.

This dissertation aims to address two major problems raised by these three examples. First, it is necessary to develop general and flexible modeling methods for analyzing functional responses in the context of experimental studies. Many studies involving functional data result from observational studies, such as longitudinal studies on prediction of success in College by Harackiewicz *et al.* (2002), literacy development in children by Juel (1988), and health quality of adolescents Swallen *et al.* (2005). Modeling functional responses with experimental covariates is not yet a well-tapped research area, so there is a need to determine which FDA models are most suitable as practical inputs in the search for optimal experimental designs. As will be discussed later, a critical assumption in the search for optimal designs is the model form.

Second, the functional nature of the responses presents challenging issues in the design of optimal experiments. In static-response experiments, standard experimental designs (such as the 2^k designs) and algorithms for optimal designs have been developed to estimate and predict common statistical measures, such as the process mean and variance. Comparing to models used in static-response experiments, models for functional data are made more complex by the number and type of model parameters of interest, and the presence of inherent correlation with respect to the spectrum variable. Optimal design methods should be adapted to account for this complexity. In doing so, the designs constructed could be guaranteed to be optimal or near-optimal with regard to a certain criterion.

Very few publications have dealt with the problem of the design and analysis of experimental data for functional responses. In modeling, existing researches have focused on the single-subject scenario (Del Castillo et al., 2012). In experimental designs, researches have primarily focused on the specific problem of designing experiments for fMRI studies using genetic algorithm. For example, see Kao *et al.* (2009); Kubilius *et al.* (2011); Eck *et al.* (2013); Mijović *et al.* (2014). Genetic algorithm is an optimization technique that has been widely used in finding optimal designs for linear and nonlinear models (Romeijn and Pardalos, 2002), but it has received criticism for its computational inefficiency.

This research aims to address the two major problems previously mentioned. Chapter 2 focuses on comparing different functional data analysis (FDA) models. The main purpose of this chapter is to introduce the reader to different FDA models, and to recommend a general and flexible modeling approach for experimental design. The mixed effects model, and the varying coefficients model are explored in Chapter 2. These models are extended to accommodate multiple experimental test units, which is the first significant contribution of this research.

Chapter 3 presents a short review of basic optimal design concepts. In this section, the widely used optimal design criteria, such as D-optimality, I-optimality and G-optimality, are described. Optimal experimental designs are typically constructed to fulfill one of two major objectives namely, to estimate model parameters or to predict within the design space. Design criteria that are focused on estimation aim to improve the quality of parameter estimates by minimizing the variances of model coefficients. D-optimality is the most popular criterion due to its efficient update formula and invariance to linear transformations, among others (see Pukelsheim (1993), Fedorov (1971)). A D-optimal design maximizes the determinant of the expected Fisher information matrix of parameter estimates, which is equivalent to minimizing the volume of the confidence region of regression coefficients. G-optimality, another popular prediction-based criterion, minimizes the maximum average prediction over the design space.

Optimal designs are categorized in literature as *exact* designs and *approximate* designs. Exact designs are constructed using computer algorithms such as the point-exchange (Fedorov, 1969; Fedorov, 1971) and coordinate exchange algorithms (CEA) (Meyer and Nachtsheim, 1995). These algorithms are described in detail in Chapter 3. Further, Chapter 3 presents the second significant contribution of this research namely, an exchange algorithm for finding G-optimal designs called the cCEA. The demonstration of the development of a coordinate exchange approach in this chapter lays the groundwork for the development of exchange algorithms for finding optimal designs for functional responses

Chapter 4 introduces an efficient exchange algorithm for finding D-optimal designs in fMRI experiments. In an fMRI study, there are two pertinent sampling times – the interstimulus interval (τ_{isi}) and the time between MR scans of the same brain voxel or region. A stimulus, $q, q = 1, 2, ...Q$, is shown to the patient every τ_{isi} time unit.

In this chapter, the sampling times are fixed or prespecified but the optimal stimulus sequence is designed into the experiment. The optimal design problem now becomes determining the sequence of the Q stimuli that maximizes a functional of the model matrix.

Two problems are of interest in fMRI studies. The first problem is the estimation of the Hemodynamic Response Function (HRF) curve, and the second is the detection of the activated region in the brain. Each problem requires a different functional data model. In the estimation problem, the shape of the response function is of interest in estimation while in the detection problem, there is a presumed shape to this function but the amplitude of the response function that indicates the level of response to a specific stimulus is unknown. The proposed algorithm derives D-optimal designs for both cases. The algorithm takes advantage of the sparseness of the design matrix to improve computational efficiency. More specifically, updating formulas are proposed to simplify the calculation of the determinant at each iteration.

Chapter 5 provides an extension of the D-optimal design algorithm to more general FDA problems. The contributions in this chapter are two-fold. We first consider the problem of determining the optimal sampling points of the spectrum variable. This problem is motivated by cases where there are limited resources to take measurements on a process or a system, such as in the case of degradation data in reliability systems. Among different modeling techniques, we choose the basis splines (B-splines) model introduced by De Boor *et al.* (1978). B-splines are non-parametric models that introduce modeling flexibility while require limited assumptions about the functional form of the data. An efficient algorithm similar to PEA is proposed to find the optimal sampling times. In this research, the key contribution is in improving the quality of the candidate space so that only highly-likely candidates are iterated in the search. This is made possible by taking advantage of the sparsity of the design matrix.

The second half of Chapter 5 considers the additional problem of estimating covariate effects, in addition to determining optimal sampling times. We introduce a two-step algorithm for constructing the D-optimal design that simultaneously optimizes both parameters of the experiment. The proposed algorithm is compared with hypothetical and existing designs in the literature to prove the efficacy of the algorithm.

Finally, Chapter 6 recapitulates the important points in the previous chapters, as well as possible extensions of this research.

Chapter 2

FUNCTIONAL DATA ANALYSIS MODELS

Notational Conventions

$\mathbf{Y}$	**Response Vector**
N	**Number of Experiments**
$\mathbf{B(x)}$	**Basis Matrix**
$\mathbf{t}$	**Vector of Sampling Time**
$\boldsymbol{\theta}$	**Vector of Unknown Model Parameters for Smoothing**
$\mathbf{W}$	**Variance-Covariance Matrix of Observations**
$\mathbf{m}$	**Order of B-spline Bases**
L	**Number of Interior Knots for B-spline system**
$\mathbf{D^m}$	**m-th derivative of a function**
$\mathbf{U}$	**Design Matrix for Fixed Effects**
$\mathbf{Z}$	**Design Matrix for Random Effects**
$\mathbf{\Sigma}$	**Covariance Matrix of Mixed Effects Model**
$\mathbf{L}$	**Covariance Matrix of the Random effects in Mixed Effects Model**
$\boldsymbol{\beta}$	**Vector of Unknown Model Parameters in the Mixed Effects Model**
$\boldsymbol{\omega}$	**Vector of Random Effects in the Mixed Effects Model**
$\mathbf{V}$	**Covariance Matrix of the Observations in Mixed Effects Model**
$\mathbf{G}$	**Design Matrix in Split-Plot Model**
$\mathbf{K}$	**Assignment Matrix in Split-Plot Model**
Q	**Number of Stimuli in fMRI Experiment**
$\mathbf{d}$	**Stimulus Sequence**
d_i	**i-th Element of Stimulus Sequence**
τ_{ISI}	**Interval Time Between Two Consecutive Stimuli Demonstration**
τ_{TR}	**Interval Time Between Two Consecutive fMRI Scans**
$\mathbf{X_q}$	**Design Matrix for q-th Stimulus**
$\mathbf{X}$	**Design Matrix**
$\mathbf{F(x)}$	**Design Matrix for Independent Factors**
$\mathbf{h_q}$	**Unknown Parameter Vector for q-th Stimulus**
$\mathbf{h}$	**Concatenation of All $\mathbf{h_q}$**
ΔT	**the largest denominator that makes both $\tau_{ISI}/\Delta T$ and $\tau_{TR}/\Delta T$ integers**
$\mathbf{h}^*$	**The Basis Function for Modeling the fMRI Experiment for Detection Problem**

$\theta_{\mathbf{q}}$	**Unknown Magnitude in Detection Problem for q-th Stimulus**
$\mathbf{M_h}$	**Moment Matrix for fMRI Estimation Problem**
$\mathbf{M}_\theta$	**Moment Matrix for fMRI Detection Problem**
$\mathbf{M}_\theta$	**Moment Matrix for fMRI Detection Problem**
$\mathbf{H}$	**Matrix of the Unknown Model Parameters in Second Stage of the Hierarchical Modeling**
$\mathbf{f(x_i)}$	**Column Vector That Contains the Values of i-th Experimental Factors**
$\mathbf{P}$	**Design Matrix in Varying Coefficient Model**
$\beta(\mathbf{t})$	**Varying Coefficient**
λ	**Non-negative Smoothing Parameters Which Penalize the Roughness of βs**
$\mathbf{B_T}$	**Basis Matrix for Modeling the Varying Coefficients Over a Continuum**
D_γ	**The D-optimal Criterion for Varying Coefficient Model**

Functional data is a category of observational or experimental data that is collected over a spectrum variable. For every single response variable, a series of data y_{it} are measured at discrete points of the spectrum variable, t. Processes and systems that exhibit inherent dynamic behavior may be naturally modeled as functional data, where typically, information about the dynamic behavior of systems can not be aptured by static-response models.

In this research, the first natural question is how functional data is analyzed using statistical models. The construction of optimal experimental designs require the complete specification of the model form. The focus of this chapter is on the model-based analysis of functional data. First, a review of these models and other approaches in literature is presented. Next, the mixed effects and varying coefficients models, two popular techniques for analyzing functional data, are adapted and generalized for the purpose of modeling experimental data with multiple experimental units or subjects. In addition, the varying coefficient model is implemented in a hierarchal form, where the B-spline bases are fitted to the derived varying coefficients. The steps of this generalization are provided in this chapter. Finally, the varying coefficients

and mixed effects models are compared and contrasted based on predictive error and the practicality as a model choice for designing optimal experiments.

2.1 Review of FDA Approaches

Techniques for modeling functional data are best understood based on two conditions, namely, the presence or absence of co-factors or covariates. In the absence of covariates, analyzing functional data can be perceived as a generalization of smoothing time series data. Some widely-used time series smoothing techniques include the simple moving average and EWMA. Stone (1974), Silverman (1985), and Rice and Silverman (1991) proposed cross-validation methods for determining the level of smoothing in functional data analysis. However, these methods may under-smooth the data when the error terms are correlated. In such case, Diggle and Hutchinson (1989), Altman (1990) and Hart (1991) proposed a model-free approach to smooth individual response curve. Non-parametric smoothers are also studied in, e.g., Hastie and Tibshirani (1990), Silverman and Ramsay (2005), and Eubank (1999). These non-parametric smoothers use the least square estimation method to fit smoothing functions, such as B-splines, to noisy observations.

In the presence of covariates, the general approach is to smooth functional data on the spectrum variable and then regress the coefficients of smoothing function on the covariates. In this regard, approaches for analyzing functional data may be categorized as either parametric or non-parametric approach. Besse and Ramsay (1986) explored principal-components analysis for analyzing functional data. Royston and Altman (1994), and Lesaffre *et al.* (1999) used fractional polynomial models for modeling functional data against time. Pan and Goldstein (1998) applied non-parametric spline functions with the same intent. Spline smoothing estimators of model param-

eters was introduced by Hastie and Tibshirani (1993). This estimator was further studied by Hoover *et al.* (1998). A generalized version of varying coefficient model where factors and parameters of the response model are functions time was proposed by West *et al.* (1985). Fan and Zhang (2000) used a two-step approach to estimate the varying coefficients of this generalized model. Eubank *et al.* (2004) demonstrated an efficient way of calculating spline smoothing estimators. Chiang *et al.* (2001) discussed the use of ordinary univariate smoothers in the situation where experimental factors are constant over time.

Furthermore, hierarchal modeling technique has been discussed in literature as a powerful technique for modeling functional data. Wu and Hamada (2011a), Tsui (1999), Del Castillo *et al.* (2012), Verbeke and Molenberghs (2009a) adopted the hierarchal or 2-stage modeling technique for modeling functional data using parametric methods such as polynomial models. However, simple low-degree polynomial models might require extension to handle the nonlinear and complicated shape of functional data. However, at the experimental design stage, response curves are still unobserved and, sometimes, unpredictable. We need to either increasine the degrees of polynomial or introduce flexible piecewise regression methods such as splines. The B-spline system is employed in this dissertation due to its model compactness and computational stability.

2.1.1 Smoothing Techniques for Analyzing Functional Data

Smoothing techniques are widely used to separate desired signals from noise and the earliest applications of these methods are in time-series analysis. For example, the exponentially weighted moving average (EWMA) has been used in smoothing

health surveillance data by Zhou and Lawson (2008), and semiconductor manufacturing processes by Fan *et al.* (2002). Smoothing techniques are typically used for finding explanatory patterns in response curves.

Any non-linear, flexible function can be used to smooth functional data. A popular method is to use a series of B-splines to make up a basis system, and then model the response curve as a linear combination of these B-spline basis functions. Suppose that the basis system is stored in a matrix, $\mathbf{B}(\mathbf{t})$, then the response is given by the following linear model of basis functions:

$$\mathbf{y} = \mathbf{B}(\mathbf{t})\boldsymbol{\theta} + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim N(0, \boldsymbol{\Sigma}). \tag{2.1}$$

In the simplest case, the $M \times 1$ column vector $\mathbf{y}$ includes M consecutive observations for one subject, the $M \times 1$ column vector $\mathbf{t}$ has the values of the spectrum variable (e.g. time), and $\boldsymbol{\theta}$ is the model parameter vector. A basis system has three tuning parameters that need to be specified beforehand and they are the number of bases, the location of the knots, and the order of the spline system. The (i, k)-th element in matrix $\mathbf{B}(\mathbf{t})$ indicates the value of the k^{th} basis value at t_i. Consequently, the ordinary least squares estimator of the model parameters in Equation (2.1) is given by

$$\hat{\boldsymbol{\theta}} = (\mathbf{B}(\mathbf{t})'\mathbf{B}(\mathbf{t}))^{-1}\mathbf{B}(\mathbf{t})'\mathbf{y}. \tag{2.2}$$

If the assumptions of independence and equal variance among observations are not satisfied, a weighted least squares estimator is more appropriate, i.e.,

$$\hat{\boldsymbol{\theta}} = (\mathbf{B}(\mathbf{t})'\mathbf{W}^{-1}\mathbf{B}(\mathbf{t}))^{-1}\mathbf{B}(\mathbf{t})'\mathbf{W}^{-1}\mathbf{y}, \tag{2.3}$$

where the square matrix **W** is the variance-covariance matrix of observations.

In fact, any basis system can be used for constructing B(t). For example, the Fourier basis system is given by:

$$[1, \sin \omega t', \cos \omega t, \sin \omega t', \cos \omega t', ...] \tag{2.4}$$

Figure 2.1 shows a set of five Fourier bases defined on the range [0,1] with the basic cycling period equals to 0.5 .

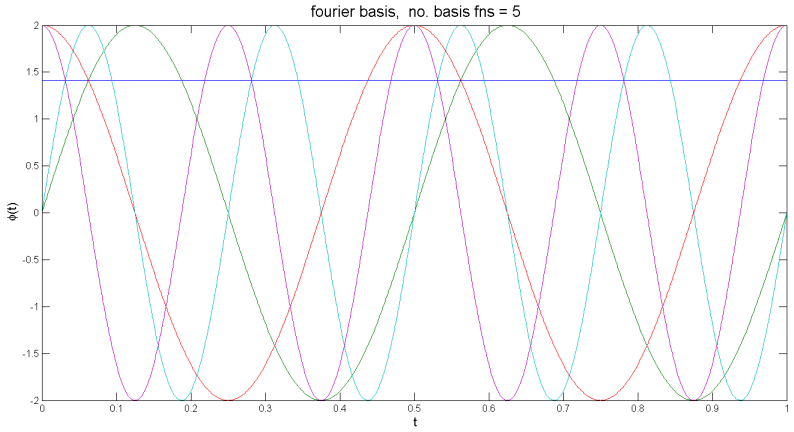


Figure 2.1: A Set of Five Fourier Bases Defined on the Range [0,1] with Period Equals to 0.5

The Fourier basis system is accurate at modeling stationary data that do not exhibit extreme infimums, supremums, or lack of local behaviors. The B-spline basis system, on the other hand, is more appropriate for the case are many known local behaviors on the response curve. Proposed by De Boor *et al.* (1978), B-spline basis functions are generated in a recursive way. The range of spectrum variable is partitioned into several sub-intervals by prespecified interior knots or break points and the number of bases needed for an order-m spline function equals to the number of interior knots plus m. These order-m basis functions can be recursively formed by

their lower order basis functions. The $\mathbf{B}(\mathbf{t})$ matrix are then obtained by evaluating these basis functions at $\mathbf{t}$. For example, consider an order-three B-spline basis system, $m = 3$. Scale the range of spectrum variable to be [0, 1] with knot points located at $\boldsymbol{\tau} = \{0, 0.3, 0.6, 0.9, 1\}$. Figure 2.2 provides the basis functions of this B-spline system. If this system is evaluated at $\mathbf{t} = \{0, 0.2, 0.3, 0.4, 0.8, 0.9, 1\}$,then the basis matrix is given by:

$$\mathbf{B}(\mathbf{t}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0.11 & 0.66 & 0.23 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0.22 & 0.72 & 0.06 & 0 & 0 \\ 0 & 0 & 0.05 & 0.62 & 0.33 & 0 \\ 0 & 0 & 0 & 0.25 & 0.75 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\mathbf{B_k}(\mathbf{t})$ indicates the value of k^{th} basis value at $\mathbf{t}$.

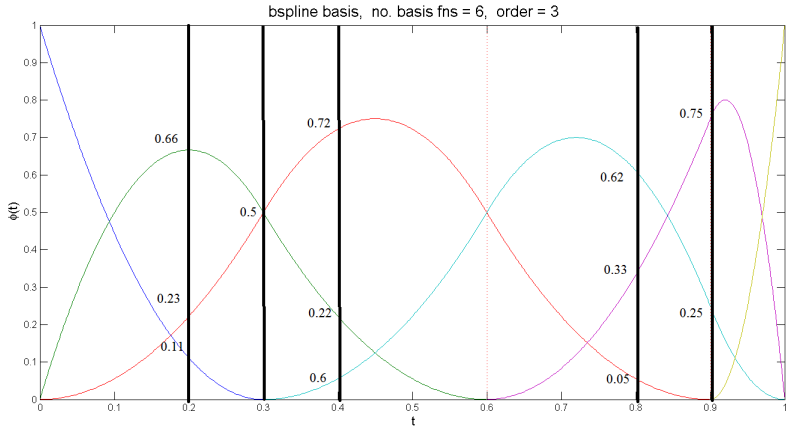
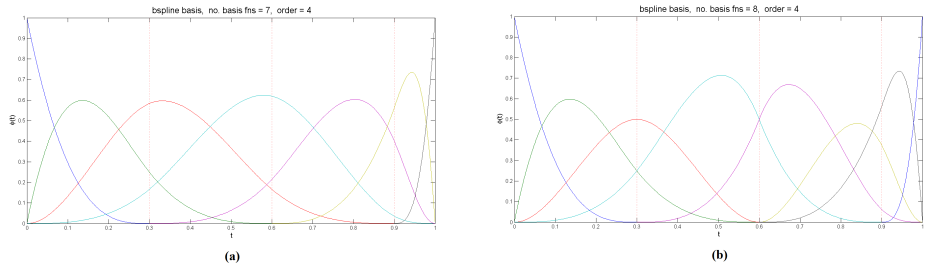


Figure 2.2: A Set of Six B-Spline Bases Defined on the Range [0,1] with the Knots Located at $\boldsymbol{\tau} = \{0, 0.3, 0.6, 0.9, 1\}$.

B-spline systems enjoy the properties such as compact support or the ability to control differentiability of the function at arbitrary points. The compact support

property implies that an order-m B-spline function is only non-zero in at-most m adjacent intervals on the spectrum variable. This property results in a relatively sparse design matrix. As an additional consequence of the compact support property, the number of differentiations of a spline function can be controlled by adding duplicate knots. For example, an order-m B-spline function has $m-2$ continuous derivatives at each interior knot. If it is desired to reduce the number of continuous derivatives at a specific knot location, a duplicate knot can be added to that specific location.

The number of non-zero B-spline basis functions is calculated by adding the order of B-spline functions to the total number of interior knots or $p = m + L$, where L is the number of interior knots. Figure 2.3 shows a set of order-4 B-spline functions with knots at $\boldsymbol{\tau} = \{0, 0.3, 0.6, 0.9, 1\}$. Compared to Figure 2.3 (a), 1, 2, and 3 duplication of knot 0.6 are added to the basis systems in Figures 2.3 (b), (c), and (d), respectively. Therefore, the B-splines using the basis systems in Figures 2.3 (a), (b), and (c) have 2, 1, and 0 continuous derivatives at 0.6, respectively, and the B-spline using the basis system in Figure 2.3 (d) has discontinuity at this point.



(a) (b)

The unknown parameter vector, $\boldsymbol{\theta}$ in Equation (2.1) can be calculated by Equation (2.2) or (2.3). Statistics packages such as R and MATLAB feature estimation suites or libraries for functional data analysis.

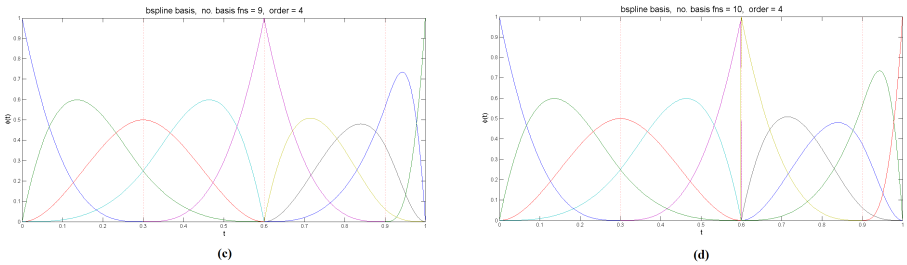


Figure 2.3: Order Four B-Spline Bases System with 7,8,9 and 10 Bases Functions Which Are Derived from the Knot Points That Are Located at {0,0.3,0.6,0.9,1}, {0,0.3,0.6,0.6,0.9,1}, {0,0.3,0.6,0.6,0.6,0.9,1} and {0,0.3,0.6,0.6,0.6,0.6,0.9,1}, Respectively from Left to Right.

Aside from using least squares methods, using roughness penalty functions is another powerful technique for smoothing and estimating functional data. Smoothing by a roughness penalty requires the quantification of roughness. Roughness penalty is defined as:

$$PEN_m(g(\mathbf{t})) = \int [D^m g(s)]^2 ds, \tag{2.5}$$

where D^m indicates m-th derivative of the function. The penalty function of Equation (2.5) is used to penalize the residual sum of squares,

$$PENSSE_\lambda(g(\mathbf{t})|\mathbf{y}) = [\mathbf{y} - g(\mathbf{t})]'\mathbf{W}[\mathbf{y} - g(\mathbf{t})] + \lambda \times PEN_m(g(\mathbf{t})), \tag{2.6}$$

In Equation (2.6), $g(\mathbf{t})$ is estimated by minimizing $PENSSE_\lambda(g(\mathbf{t})|\mathbf{y})$, while λ is a tuning parameter that controls the roughness of the function.

Smoothing is a technique that is only effective when the interest is in modeling the response function against the spectrum variable, such as time. In the case where the effects of other covariates are of interest, some paramaterized models, such as the mixed effects model and varying coefficients model are more appropriate. These two models are discussed in the next two subsections.

2.1.2 Mixed Effects Model

The *mixed effects model* has a wide variety of applications in the field of experimental design and analysis, such as the split-plots experiment, the repeated measurement experiments, the fMRI experiment, etc. Using matrix notation, the mixed effects model is written as:

$$\mathbf{y} = \mathbf{U}\beta + \mathbf{Z}\gamma + \epsilon, \tag{2.7}$$

where

$$\epsilon \sim N(0, \mathbf{\Sigma}), \quad \gamma \sim N(0, \mathbf{L}), \quad Cov(\epsilon, \gamma) = \mathbf{0}. \tag{2.8}$$

To simplify analysis, $\mathbf{\Sigma}$ is assumed to be equal to $\sigma^2\mathbf{I}$. On the other hand, the structure of the covariance matrix of random effects, $\mathbf{L}$, is assumed beforehand or determined by the problem under study. Another important assumption in the mixed effects model is that β and γ are independently distributed, implying that:

$$E(\mathbf{y}) = \mathbf{U}\beta \quad \mathbf{Cov}(\mathbf{y}) = \mathbf{V} = \mathbf{\Sigma} + \mathbf{ZLZ}' \tag{2.9}$$

Maximum likelihood estimation (MLE) can be used for estimating the unknown model parameters of the mixed effects model, which is

$$\hat{\beta} = (\mathbf{U}'\mathbf{V}^{-1}\mathbf{U})\mathbf{U}'\mathbf{V}^{-1}\mathbf{y} \tag{2.10}$$

with the covariance matrix as

$$\mathbf{Cov}(\beta) = (\mathbf{U}'\mathbf{V}^{-1}\mathbf{U})^{-1}. \tag{2.11}$$

Equation (2.7) was used in Goos and Jones (2011) for modeling split-plot industrial experiments, in Laird and Ware (1982) for modeling repeated measures, in Liu and

Frank (2004) and Kao *et al.* (2009) for modeling fMRI experimental data, and in Wu and Hamada (2011a) and Tsui (1999), Del Castillo *et al.* (2012) and Verbeke and Molenberghs (2009a) for modeling functional data. In subsequent discussions, we will present two applications of mixed effects models.

2.1.2.1 Mixed Effects Model in fMRI Experiments

Functional Magnetic Resonance Imaging (fMRI) is a brain imaging technique that aims to study human brain response to stimuli in an organized experiment. In this section, we discuss how the mixed-effects model is utilized in analyzing data from fMRI experiments.

In the design of an fMRI experiment, the main concern is in determining the sequence in which mental stimuli are presented to an experimental subject. Suppose that there Q types of stimuli and that each stimulus type appears in the experiment at multiple time points. When $Q = 2$, a possible fMRI experimental design is a stimulus sequence such as $\mathbf{d} = \{121020...1\}$. In this example, the i-th element, d_i, $i = 1, 2, ..., N$, represents the stimulus type presented at the i-th time point. A value of 0 indicates a rest period where no stimulus was presented. The number of elements in a design, denoted by N, can be any integer so consequently, there is a large number of possible designs ($(Q + 1)^N$). Further, the interval between two consecutive time points is denoted by τ_{ISI} (e.g., 4 seconds). If the experiment starts at time 0, then d_i is presented at time $(i - 1)\tau_{ISI}$. Each stimulus is presented briefly to the subject (e.g., 1 second).

The primary goal of fMRI experiments is to collect data on brain activity. The different regions of the brain react differently to a stimulus type. These brain activities are analyzed by collecting and studying fMRI time series produced by brain voxels (three dimensional imaging units) that cover the region of interest. Inferences on brain

activity at each voxel are made by studying the characteristics of the hemodynamic response function (HRF). The HRF models the change in the ratio of oxy- to deoxy-blood evoked by the stimulus (see, e.g., Lazar (2008) for details on the HRF). We show in Figure 2.4 an example of HRFs evoked by a simple one-type stimulus sequence with $\tau_{ISI} = 3$ seconds and $\tau_{TR} = 2$ seconds. τ_{TR} is the time between two consecutive scans of the fMRI. Overlapping HRFs are a consequence of short τ_{ISI}. Under the widely-used linear time invariant (LTI) system (Lindquist et al., 2008), the heights of these HRFs accumulate linearly to form an "accumulated HRF" as in Figure 2.4. We note that, in most cases, the assumption of linear accumulation holds unless τ_{ISI} is short, e.g. $\tau_{ISI} < 2$ seconds (see Dale and Buckner (1997)).

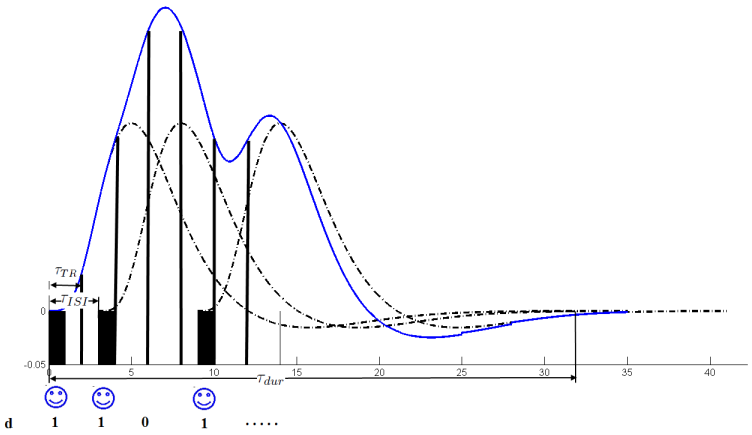


Figure 2.4: The Accumulated HRF of Three HRFs That Correspond to a Design Sequence $\mathbf{D} = \{1101...\}$ with $\tau_{ISI} = 3$ and $\tau_{TR} = 2$. The Accumulated HRF, Drawn by a Blue Solid Curve, Is Equal to the Summation of HRFs, Drawn by Black Dotted Curves.

One common objective of an fMRI study is to estimate the HRF of each stimulus type. This is called the estimation problem in fMRI design literature. For this specific objective, the following model is proposed (Liu and Frank, 2004):

$$\mathbf{y} = \sum_{q=1}^{Q} \mathbf{X}_q \mathbf{h}_q + \mathbf{S}\gamma + \mathbf{e}. \tag{2.12}$$

The elements of this model are subsequently described.

The response vector, $\mathbf{y} = (y_1, ..., y_T)'$, is the fMRI time series of a voxel obtained by an fMRI scanner. The length of this time series, T, depends on the experimental duration and the scanning frequency of the fMRI machine. This model aims to separate the effects of the different stimuli by estimating the unknown parameter vector $\mathbf{h}_q$, which are heights of the discretized HRF for the q-th stimulus type so that $\mathbf{h}_q = (h_{q,1}, ..., h_{q,K})'$ for $q = 1, ..., Q$. Further, in Equation (2.12), $K = 1 + \lfloor \tau_{dur}/\Delta T \rfloor$, $\lfloor . \rfloor$ is a flooring function, ΔT is the largest denominator that makes both $\tau_{ISI}/\Delta T$ and $\tau_{TR}/\Delta T$ integers, and τ_{dur} is the effective period of a single stimulus (i.e. the duration of the HRF). To demonstrate with a simple example, suppose that the effect of one stimulus lasts $\tau_{dur} = 32$ seconds. In this case, if $\tau_{ISI} = 4$ and $\tau_{TR} = 2$, then $\Delta T = 2$ and $K = 17$. We note that ΔT may or may not be an integer.

The matrix $\mathbf{X}_q$ in Equation (2.12) is a T-by-K matrix populated by 1's and 0's. Its $(t, k)^{th}$ element denotes whether the k-th height of the HRF of the q-th stimulus type contributes to the t-th fMRI signal, y_t.

The nuisance term $\mathbf{S}\boldsymbol{\gamma}$ models the frequently observed trend/drift of fMRI time series. Here, $\mathbf{S}$ is a specific matrix and $\boldsymbol{\gamma}$ is the corresponding parameter vector.

Vector $\mathbf{e}$ in Equation (2.12) is a T-by-1 noise vector with mean $\mathbf{0}$ and $cov(\mathbf{e}) = \sigma^2\mathbf{R}$ for some $\sigma^2 > 0$ and a positive definite matrix $\mathbf{R}$.

Because the heights of the discretized HRF of each stimulus type are of interest in the estimation problem, an experimental design that minimizes the general variance of the least-square estimate of $\mathbf{h} = (\mathbf{h}_1', ..., \mathbf{h}_Q')'$ is desired.

Another common fMRI study objective is to identify the brain voxels activated by a stimulus type. In fMRI literature, this is called the detection problem, and the following linear model often used for this objective:

$$\mathbf{y} = \sum_{q=1}^{Q} \mathbf{X}_q \mathbf{h}^* \theta_q + \mathbf{S}\gamma + \mathbf{e}, \tag{2.13}$$

where $\mathbf{h}^*$ in Equation (2.13) is a basis function that contains information about the heights of the discretized HRF function. In the detection problem, these heights are assumed to be known and equivalent for different stimuli. Hence, the HRF of the q-th stimulus type is modeled using information about $\mathbf{h}^*$ with an unknown magnitude, θ_q. The other terms in Equation (2.13) are defined as in Equation (2.12). For the detection problem, the parameter vector $\boldsymbol{\theta} = (\theta_1, ..., \theta_Q)'$ is of interest, and it indicates the strength of brain activity due to each stimulus type.

In likelihood estimation, a statistic of interest is the *information matrix*, a metric that indicates the amount of information the data set holds about the unknown parameters in a statistical model. This matrix is calculated by taking the negative expectation of the second derivative of the log-likelihood function with respect to the unknown parameters of the model. The information matrix can be written as:

$$\mathbf{I}(\beta) = -\mathbf{E}[\frac{d^2}{d\beta^2} \log f(x; \beta)|\beta]. \tag{2.14}$$

Generally, the asymptotic variance of the maximum likelihood estimator is equal to the inverse of the Fisher information. The information matrix is pertinent in the design of optimal experiments, and we defer further discussions about this matrix for Chapter 3. In the meantime, we define the information matrices for the two fMRI models just described. For the estimation model in (2.12), the information matrix is derived as:

$$\mathbf{M}_h = \mathbf{X}'\mathbf{V}'(\mathbf{I}_T - \mathbf{P}_{\mathbf{VS}})\mathbf{V}\mathbf{X}, \tag{2.15}$$

where $\mathbf{X}$ is the concatenation of all $\mathbf{X}_q$'s (i.e., $\mathbf{X} = [\mathbf{X}_1 ... \mathbf{X}_Q]$), $\mathbf{V}$ is a matrix such that $\mathbf{VRV}' = \mathbf{I}_T$ with $\mathbf{I}_T$ being the T-by-T identity matrix, and $\mathbf{P_{VS}} = \mathbf{VS}(\mathbf{S}'\mathbf{V}'\mathbf{VS})^{-1}\mathbf{S}'\mathbf{V}'$ is the orthogonal projection matrix onto the column space of $\mathbf{VS}$.

For the detection model in (2.13), the information matrix is given by:

$$\mathbf{M}_\theta = (\mathbf{I}_Q \otimes \mathbf{h}^*)'\mathbf{X}'\mathbf{V}'(\mathbf{I}_T - \mathbf{P_{VS}})\mathbf{VX}(\mathbf{I}_Q \otimes \mathbf{h}^*), \tag{2.16}$$

where $\otimes$ denotes the Kronecker product.

2.1.2.2 Hierarchical Modeling for Functional Data

The mixed effects model for modeling functional data can also be perceived by using a hierarchical modeling approach (Wu and Hamada (2011a), Tsui (1999), Del Castillo *et al.* (2012), and Verbeke and Molenberghs (2009a)).

Consider an experiment in which each experimental unit is subject to a treatment and where the response is measured over time. The hierarchical modeling technique consists of two consecutive stages. In the first stage, the response curve over time is modeled by a non-parametric function, and in the second stage, the coefficients of the first stage model is modeled as functions of treatments.

In mathematical notation, the two stage hierarchical model is represented as:

$$\mathbf{y}_j = \mathbf{B}(\mathbf{t})\boldsymbol{\theta}_j + \boldsymbol{\epsilon}_j \quad \boldsymbol{\epsilon}_j \sim N_M(0, \boldsymbol{\Sigma}), \tag{2.17}$$

and

$$\boldsymbol{\theta}_j = \mathbf{Hf}(\mathbf{x}_j) + \boldsymbol{\omega}_j \quad \boldsymbol{\omega}_j \sim N_M(0, \boldsymbol{\Sigma}_\omega), \tag{2.18}$$

where $\mathbf{y}_j^T = [y_{1j}, y_{2j}, ..., y_{Mj}]$ is a $1 \times M$ column vector containing the repeated measures over the continuum (e.g., time) for the j^{th} profile, $\mathbf{B}(\mathbf{t})$ is a $M \times p$ basis matrix that consists of the values of the basis function at specified time points, $\boldsymbol{\theta}_j$ is a $p \times 1$ column vector of regression coefficients, $\mathbf{f}(\mathbf{x}_j)$ is a $q \times 1$ column vector that contains the values of experimental factors, and $\mathbf{H}$ is a $p \times q$ matrix of the unknown model parameters.

Intuitively, the first-stage model individually smooths out the actual observed data profile $\mathbf{y}_j$, while the second-stage model is used to relate the smoothing parameters to the experimental factors. As a consequence of the multi-stage modeling, a relationship between the response function and the treatments is derived.

Del Castillo *et al.* (2012) and Verbeke and Molenberghs (2009a) proposed combining Equations (2.17) and (2.18) to derive the mixed effects model so that:

$$\mathbf{y}_j = \mathbf{B}(\mathbf{t})[\mathbf{H}\mathbf{f}(\mathbf{x}_j) + \boldsymbol{\omega}_j] + \boldsymbol{\epsilon}_j = \mathbf{B}(\mathbf{t})\mathbf{H}\mathbf{f}(\mathbf{x}_j) + \mathbf{B}(\mathbf{t})\boldsymbol{\omega}_j + \boldsymbol{\epsilon}_j, \tag{2.19}$$

Using Kronecker product of two matrices, the first term of the right hand side of the equation can be rewritten as $(\mathbf{f}(\mathbf{x}_j)^T \otimes \mathbf{B})\mathbf{vec}(\mathbf{H})$, where $\mathbf{vec}()$ operator stacks columns of $\mathbf{H}$ to one column. Hence, the mixed effects model of Equation (2.19) can be rewritten as:

$$\mathbf{y_j} = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{B}\boldsymbol{\omega}_j + \boldsymbol{\epsilon}_j, \tag{2.20}$$

where $\mathbf{X}_j = \mathbf{f}(\mathbf{x}_j)^T \otimes \mathbf{B}$ and $\boldsymbol{\beta} = \mathbf{vec}(\mathbf{H})$. It is easy to show the variance of $\mathbf{y}_j$ is given by $\mathbf{V}_j = \boldsymbol{\Sigma} + \mathbf{B}(\mathbf{t})\boldsymbol{\Sigma}_w\mathbf{B}(\mathbf{t})^T$.

Now consider an experiment with multiple experimental units and their response curves. The mixed effects model becomes

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + (\mathbf{I_N} \otimes \mathbf{B})\boldsymbol{\omega} + \epsilon, \tag{2.21}$$

where $NM \times 1$ vector $\mathbf{Y}$ is equal to $[\mathbf{y_1}, \mathbf{y_2}, ..., \mathbf{y_N}]$, $NM \times pq$ matrix $\mathbf{X}$ is $(\mathbf{I_N} \otimes \mathbf{B})\mathbf{F}(\mathbf{x})$, where $Np \times pq$ matrix $\mathbf{F}(\mathbf{x}) = [\mathbf{I_p} \otimes f(\mathbf{x_1}), \mathbf{I_p} \otimes f(\mathbf{x_2}), ..., \mathbf{I_p} \otimes f(\mathbf{x_N})]^T$. $\boldsymbol{\beta}$, the fixed unknown parameters of the model, is equal to $[\boldsymbol{\beta}_1^T, \boldsymbol{\beta}_2^T, ..., \boldsymbol{\beta}_p^T]^T$, where $\boldsymbol{\beta}_k^T = [\beta_{k1}, \beta_{k2}, ..., \beta_{kq}]$. Finally, the random unknown parameters of the model, $\boldsymbol{\omega}$, is equal to $[\boldsymbol{\omega}_1^T, \boldsymbol{\omega}_2^T, ..., \boldsymbol{\omega}_N^T]^T$ where $\boldsymbol{\omega}_j^T = [w_{j1}, w_{j2}, ..., w_{jp}]$. It can be shown that the covariance matrix of model parameter in Equation (2.21) is equal to

$$\mathbf{COV}(\boldsymbol{\beta}) = (\mathbf{X'V^{-1}X})^{-1}, \tag{2.22}$$

where $\mathbf{V} = \Sigma + (\mathbf{I_N} \otimes \mathbf{B})\Sigma_{\boldsymbol{\omega}}(\mathbf{I_N} \otimes \mathbf{B})'$.

Therefore, the D-optimal design is to minimize the determinant of $\mathbf{COV}(\boldsymbol{\beta})$, and the D-optimal criterion is given by

$$D_{\boldsymbol{\beta}} = |(\mathbf{X'V^{-1}X})^{-1}|. \tag{2.23}$$

The derivation of this D-optimality criterion is a critical for optimal experimental designs in Chapter 5.

2.1.3 Varying Coefficient Model

The varying coefficient model is a powerful and flexible non-parametric regression technique for modeling functional data. In contrast with the mixed-effects model, this model takes a one-step modeling approach and allows the coefficients of the experimental factors to vary over the spectrum variable. Hart (1991), and Hastie and Tibshirani (1993) proposed several forms of the varying coefficient model with *time* as the spectrum variable.

1. The case when the settings of the experimental factors, P_i, have no time-dependency:

$$\mathbf{Y} = \beta_1(t)P_1 + \beta_2(t)P_2 + \beta_3(t)P_3 + ... + \beta_k(t)P_k + \epsilon. \tag{2.24}$$

2. The case when the model parameters β are dependent on the settings of the experimental factors:

$$\mathbf{y} = \mathbf{P}\beta(\mathbf{P}) + \epsilon. \tag{2.25}$$

3. The case when the model parameters vary linearly with time so that $\beta_{\cdot} \times t$:

$$\mathbf{y} = \mathbf{P}\beta t + \epsilon. \tag{2.26}$$

4. The case when both factor settings and model parameters are functions of time:

$$\mathbf{y} = \mathbf{P}(t)\beta(\mathbf{t}) + \epsilon. \tag{2.27}$$

The last model form was proposed by West *et al.* (1985) and is called the generalized/dynamic varying coefficient model. This model was comprehensively studied by Cleveland *et al.* (1992) and discussed in Eubank *et al.* (2004). We further expound on the dynamic varying coefficient model in the discussion below.

2.1.3.1 Dynamic Varying Coefficient Model

In contrast with the mixed-effects model, the dynamic varying coefficient model in Equation (2.27) can accommodate both cases when the experimental settings of the factors and the parameters vary over time. Here, the estimation method for this model is outlined. Expanding Equation (2.27) provides the following form of the varying coefficient model:

$$Y = \beta_1(t)P_1(t) + \beta_2(t)P_2(t) + \beta_3(t)P_3(t) + \ldots + \beta_k(t)P_k(t) + \epsilon : \tag{2.28}$$

Due to the challenges in estimating $\beta_i(t)$ by ordinary univariate smoothers, Eubank *et al.* (2004) proposed an efficient computational method for simultaneously smoothing the response and estimating the coefficient curves. The model for this particular technique is as shown:

$$y_{ij} = \sum_{r=1}^{k} \beta_r(t_i) p_{rij} + \epsilon_{ij} \qquad i = 1, ..., n, \quad j = 1, ..., n_i, \tag{2.29}$$

where subscripts i and j are indicators of subjects and repeated measures over time and $\epsilon_{ij} \sim N(0, \sigma^2)$. The parameter $\beta(.)$ in Equation (2.29) is estimated by minimizing the penalized least squares criterion:

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} \{y_{ij} - \sum_{r=1}^{k} p_{rij} g_r(t_i)\}^2 + \sum_{r=1}^{k} \lambda_r \int_0^1 (g_r^{(m)})^2(t) dt, \tag{2.30}$$

over all the functions $g_r \quad r = 1, 2, ..., k$, which has $m - 1$ absolutely continuous derivatives and is square integrable at the m-th derivative for $m \geq 1$. The smoothing parameters λ_1 to λ_k are non-negative numbers which penalize the roughness of $\beta_1, \beta_2, ..., \beta_k$. A unique estimator that minimizes Equation (2.30) was derived by Eubank *et al.* (2004) as:

$$\hat{\beta}_\lambda = \mathbf{A}(\lambda)\mathbf{y}, \tag{2.31}$$

where $\mathbf{A}(\lambda) = \mathbf{Q}\pi^T\mathbf{M}^{-1} - (\mathbf{Q}\pi^T\mathbf{M}^{-1}\mathbf{V} - \mathbf{T})(\mathbf{V}^T\mathbf{M}^{-1}\mathbf{V})^{-1}\mathbf{V}^T\mathbf{M}^{-\mathbf{1}}$ and $y = (y_1^T, ..., y_n^T)^T$ and $y_i = (y_{i1}, ..., y_{in_i})^T$. Matrix π in $\mathbf{A}(\lambda)$ is defined as:

$$\pi = \begin{bmatrix} \mathbf{P_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{P_n} \end{bmatrix} \tag{2.32}$$

with

$$\mathbf{P_i} = \begin{bmatrix} \mathbf{p}_{i1}^T \\ \vdots \\ \mathbf{p}_{in_i}^T \end{bmatrix} \tag{2.33}$$

and $\mathbf{p}_{ij} = (p_{1ij}, ..., p_{kij})$, $j = 1, ..., n_i$. The matrix $\tilde{\mathbf{Q}}$ is calculated by taking the following integral for any two sampling times:

$$\tilde{\mathbf{Q}} = \{\int_0^{\min(t_i,t_j)} \frac{(t_i - s)^{m-1}(t_j - s)^{m-1}}{\{(m-1)!\}^2} ds\}_{i,j=1,...,n}. \tag{2.34}$$

$\mathbf{T}$ is defined as:

$$\mathbf{T} = \begin{bmatrix} \tilde{\mathbf{t_1}}^T \otimes \mathbf{I_k} \\ \vdots \\ \tilde{\mathbf{t_n}}^T \otimes \mathbf{I_k} \end{bmatrix} \tag{2.35}$$

where $\tilde{t_i} = (1, t_i, ..., t_i^{m-1})^T, i = 1, ..., n$, and $\mathbf{Q} = \tilde{\mathbf{Q}} \otimes diag(\lambda_1^{-1}, ..., \lambda_k^{-1})$, and finally Eubank *et al.* (2004) defines $\mathbf{V} = \pi\mathbf{T}$ and $\mathbf{M} = \pi\mathbf{Q}\pi^T + \mathbf{I}$. The fitted values $\hat{\mathbf{y}_\lambda}$ can be calculated by $\mathbf{R}(\lambda)\mathbf{y}$, where

$$\mathbf{R}(\lambda) = \mathbf{I} - (\mathbf{M^{-1}} - \mathbf{M^{-1}}\mathbf{V}(\mathbf{V}^T\mathbf{M}^{-1}\mathbf{V})^{-1}\mathbf{V}^T\mathbf{M}^{-1}). \tag{2.36}$$

Therefore, the covariance matrix of the model parameters can be derived from Equation (2.31) as:

$$Cov(\hat{\beta}_\lambda) = \mathbf{A}(\lambda)Cov(\mathbf{y})\mathbf{A}^T(\lambda), \tag{2.37}$$

Further, when $Cov(\mathbf{y}) = \sigma^2\mathbf{I}$,

$$Cov(\hat{\beta}_\lambda) = \sigma^2\mathbf{A}(\lambda)\mathbf{A}^T(\lambda). \tag{2.38}$$

Finding the D-optimal design is equivalent to maximizing the determinant of the information matrix,$|\mathbf{A}(\lambda)\mathbf{A}^T(\lambda)|$.

Eubank *et al.* (2004) proposed replacing the discrete points estimates of $\beta_1, ..., \beta_k$ with k smooth and continuous curves. This now becomes a two-stage problem where $\beta_{\lambda l}^{O}$ are fitted using basis systems.

The specification of this model is as follows:

$$\beta_{\lambda l}^{O} = \mathbf{B_l}\gamma_\mathbf{l} + \epsilon_l \quad l = 0, 1, ..., k \tag{2.39}$$

where $\beta_{\lambda l}^{O} = (\beta_{\lambda l}(t_1), ..., \beta_{\lambda l}(t_n))^T$, the basis system $\mathbf{B_l}$ an $n \times p$ matrix, and $\gamma_\mathbf{l}$ the vector of p unknown parameters. Ordinary least squares estimation is used to estimate $\gamma_\mathbf{l}$ in the model.

A more general form of Equation (2.39) is:

$$\beta_\lambda = \mathbf{B_T}\gamma + \epsilon, \tag{2.40}$$

where $\mathbf{B_T}$ is

$$\mathbf{B_T} = \begin{bmatrix} \mathbf{I_k} \otimes \mathbf{B_{1.}} \\ \mathbf{I_k} \otimes \mathbf{B_{2.}} \\ \vdots \\ \mathbf{I_k} \otimes \mathbf{B_{n.}} \\ . \end{bmatrix} \tag{2.41}$$

The parameter γ in Equation (2.40) is a $(p \times k) \times 1$ column vector with the ordinary least squares estimator:

$$\hat{\gamma} = (\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T\beta_\lambda. \tag{2.42}$$

The covariance matrix of the $\hat{\gamma}$ becomes

$$cov(\gamma) = [(\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T]cov(\beta_\lambda)[(\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T]^T \tag{2.43}$$

By replacing $cov(\beta_\lambda)$ with Equation (2.37), $cov(\gamma)$ can also be written as:

$$cov(\gamma) = \sigma^2[(\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T]\mathbf{A}(\lambda)\mathbf{A}^T(\lambda)[(\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T]^T \tag{2.44}$$

As a result, the D-optimal criterion becomes:

$$D_\gamma = |[(\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T]\mathbf{A}(\lambda)\mathbf{A}^T(\lambda)[(\mathbf{B_T}^T\mathbf{B_T})^{-1}\mathbf{B_T}^T]^T| \tag{2.45}$$

where λ is the smoothing parameter previously defined for basis systems.

2.2 Model Comparison

In this section, we compare the performance of the mixed-effects model and the varying coefficients model by applying them on the data sets found in Nair *et al.* (2002) (the design of an electric alternator) and Grove *et al.* (2004) (engine mapping problem).

In Nair et al.'s experiment, electric current is recorded over various RPM (revolutions per minute) settings ($x_1 = 1375$, $x_2 = 1500$, $x_3 = 1750$, $x_4 = 2000$, $x_5 = 2500$, $x_6 = 3500$ and $x_7 = 5000$). This was accomplished for 108 designed alternators (see Figure 5.11). Eight controllable factors and two noise factors were considered by Nair *et al.* (2002) for the design problem.

For this problem, we consider the mixed effects model and the varying coefficient models with a fourth-order B-spline basis system and three internal knots. The knots are located at $\{0, 0.3, 0.5, 0.6, 1\}$. In this case, the number of bases is equal to seven. Figure 2.6 shows the basis system used for modeling the data.

Figures 2.7 and 2.8 show the predicted profiles of 10 randomly chosen response curves generated by these two models. Figure 2.9 further shows the coefficient curves

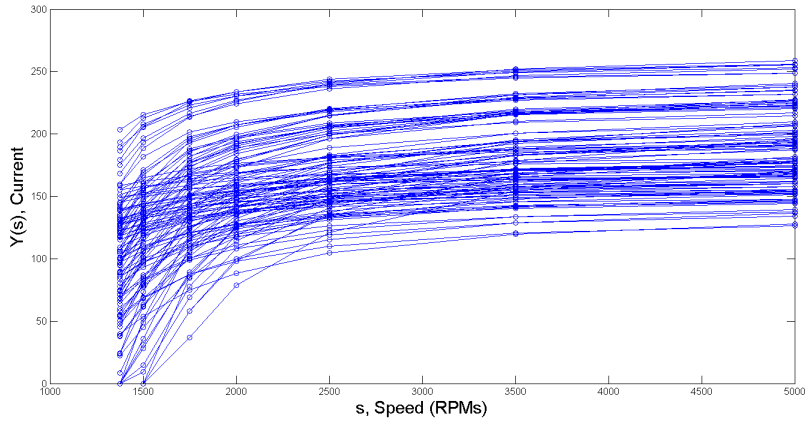


Figure 2.5: 108 Profile Curves Derived from the Experiments Conducted for Designing the Electrical Alternator. The Data Is Available for $x_1 = 1375$, $x_2 = 1500$, $x_3 = 1750$, $x_4 = 2000$, $x_5 = 2500$, $x_6 = 3500$ and $x_7 = 5000$ Which Are Shown by Circles. These Points Are Connected by Straight Lines.

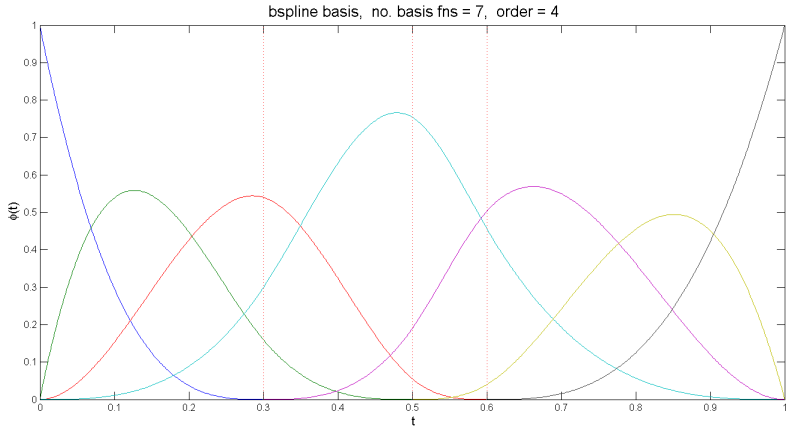


Figure 2.6: B-Spline Bases Used for the Mixed Effects and the Varying Coefficient Models. Seven Bases Functions Are Derived from the Knot Points That Are Located at 0.3, 0.5 and 0.6 and 1 with Order 4.

fitted using the B-spline basis system for the varying coefficient model. The two models perform almost equivalently with respect to the sum of squares of error. For this data set, the mixed effects model performed just a little bit better than the varying coefficients model.

Grove *et al.* (2004) discussed an engine mapping experiment where the response, brake torque, is measured with respect to varying spark advances per sweep. Fifty-five

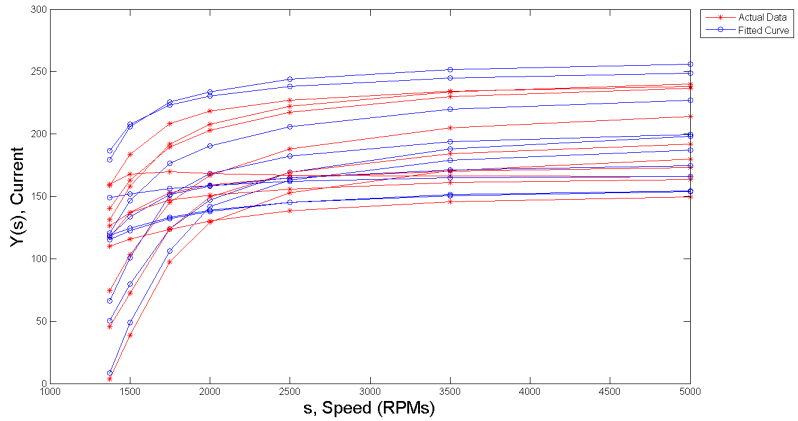


Figure 2.7: Fitted Profiles Calculated by the Mixed Effects Model for 10 out of 108 Samples. These Profile Are Chosen Randomly to Show the Model's Accuracy over All the Profiles. The Sum of Squares of the Error for This Model over All 108 Profiles Is Equal to $1.8876e+05$.

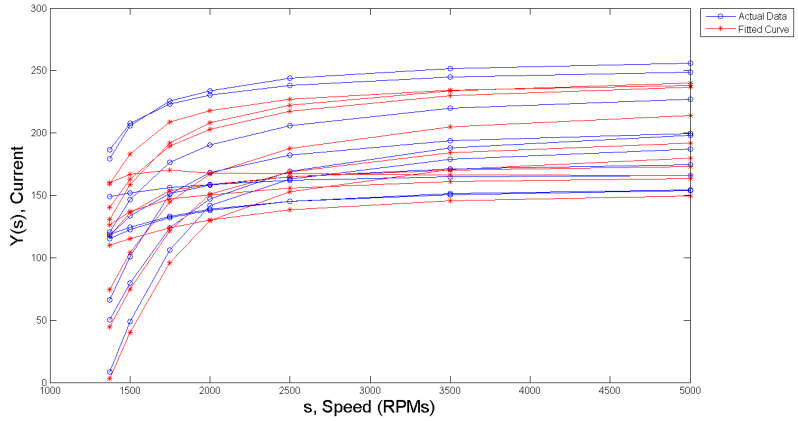


Figure 2.8: Fitted Profiles Calculated by the Varying Coefficient Model for 10 out of 108 Samples. These Profiles Are Chosen Randomly to Show the Model's Accuracy over All the Profiles. The Sum of Squares of the Error for This Model over All 108 Profiles Is Equal to $1.8911e+05$

spark sweeps were conducted in the study. The data is functional in nature because torque is measured repeatedly at different spark advances. Figure 2.10) shows the data set. In this example, the measurement points of the spark advances may be different at each sweep. Three controllable factors namely, speed (in RPM), load and AFR were of interest the original study.

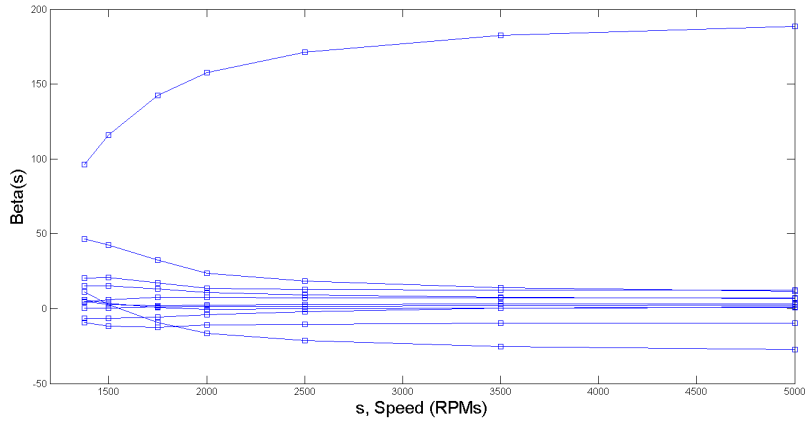


Figure 2.9: Eleven Estimated Predictor Curves Calculated by the Varying Coefficient Model for Model's Predictors and Intercept. These Curves Are Modeled in the Second Stage by the B-Spline System Which Is Shown in Figure 2.6.

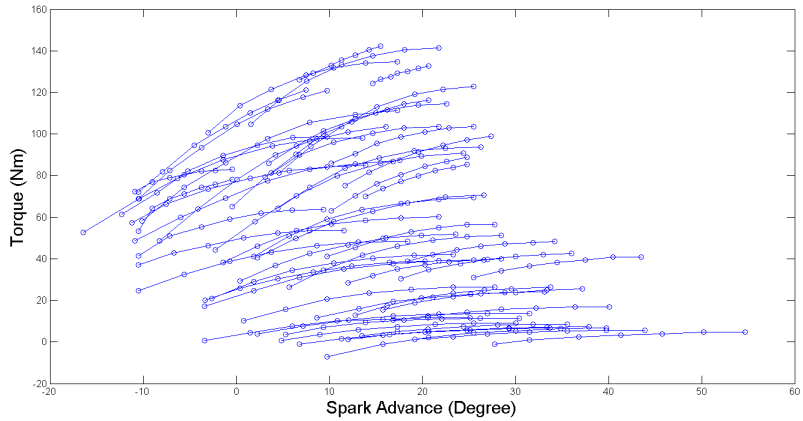


Figure 2.10: 55 Profile Curves Derived from the Experiments Conducted for Designing the Experiments for Engine Mapping. The Data Is Available for 164 Unique Degrees Which Are Shown by Circles. These Points Are Connected by Straight Lines.

In our analysis, we used an order-4 B-spline basis system with one internal knot at 0.5. Figures 2.11 and 2.12 show the predicted sweeps for 10 out of 55 randomly selected sweeps to demonstrate the predictive performance of the two models. Figure 2.13 also shows the curves fitted to the discrete points from the varying coefficient model.

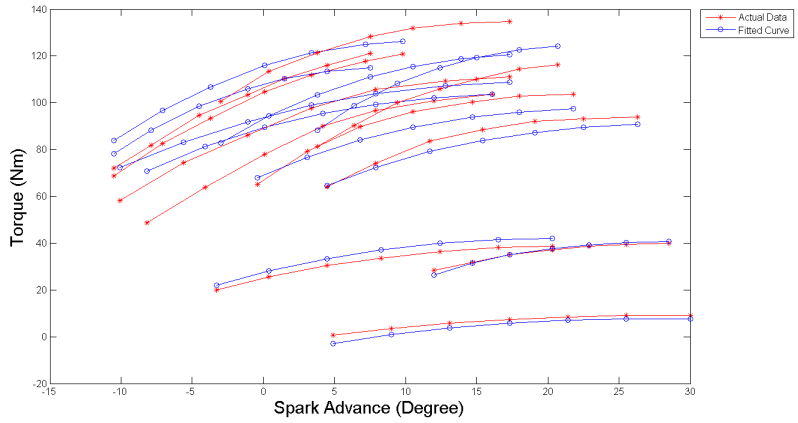


Figure 2.11: Fitted Profiles Calculated by the Mixed Effects Model for 10 out of 55 Samples. These Profiles Are Chosen Randomly to Show the Model's Accuracy over All the Profiles. The Sum of Squares of the Error for This Model over All 55 Profiles Is Equal to $1.96e+04$

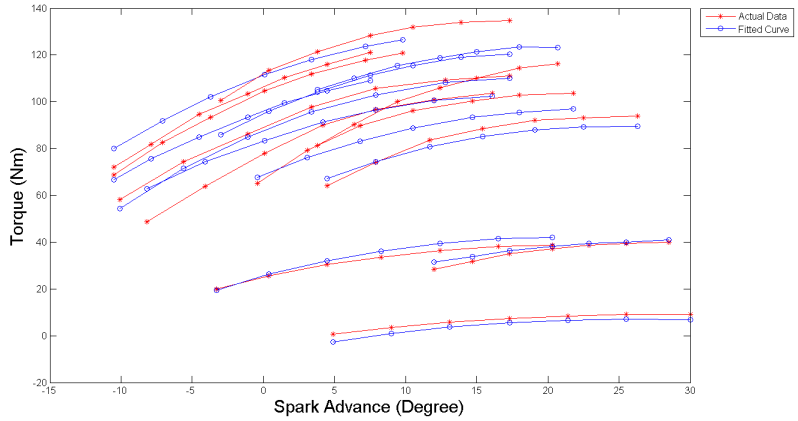


Figure 2.12: Fitted Profiles Calculated by the Varying Coefficient Model for 10 out of 55 Samples. These Profiles Are Chosen Randomly to Show the Model's Accuracy over All the Profiles. The Sum of Squares of the Error for This Model over All 55 Profiles Is Equal to $1.74e+04$

In addition to the training data, Grove *et al.* (2004) also provided a validation set of data with 6 sweeps and 7 observations along different spark advances. Both the mixed effects model and varying coefficient modeled previously are validated by this data set. Figure 2.14 and 2.15 show the actual observations and the predicted values of the mixed effects model and the varying coefficient model, respectively.

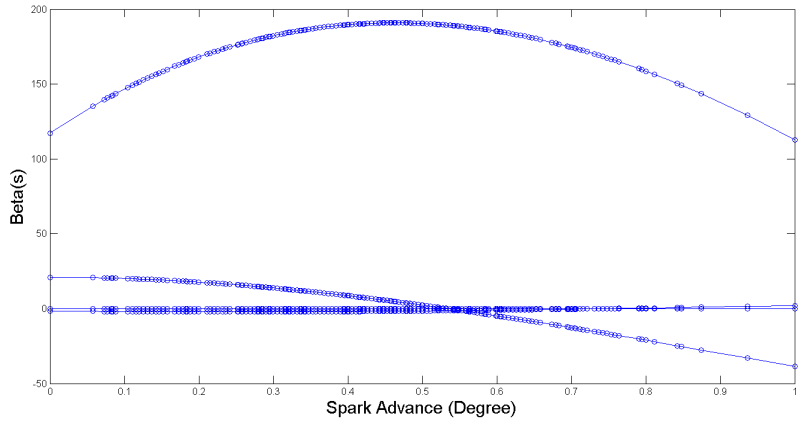


Figure 2.13: Four Estimated Predictor Curves Calculated by the Varying Coefficient Model for Model's Predictors and Intercept. These Curves Are Modeled in the Second Stage by the B-Spline System with Order 4 and Knots at 0, 0.5 and 1.

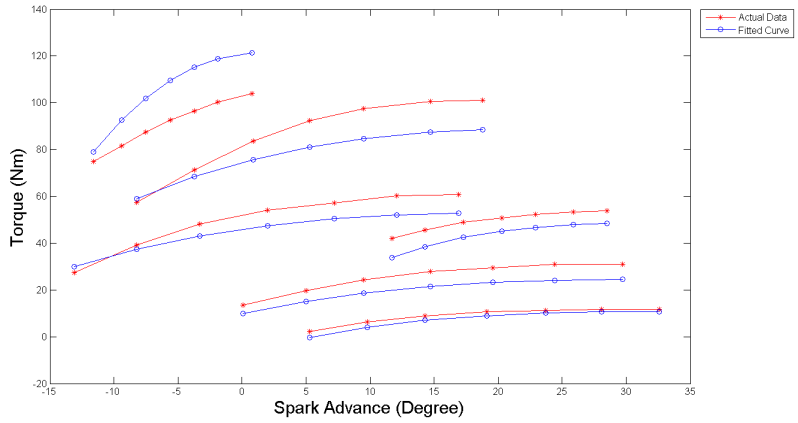


Figure 2.14: Fitted Profiles Calculated by the Mixed Effects Model for 6 Validation Sweeps. These Profiles Were Provided by Grove *et al.* (2004) to Assess the Accuracy of the Model Developed in Training Stage. The Sum of Squares of the Error for This Model over All 6 Validation Profiles Is Equal to $3.11e+03$

In our analysis, the validation and training errors of the varying coefficient model are slightly smaller than the mixed effects model, but this difference is not statistically significant.

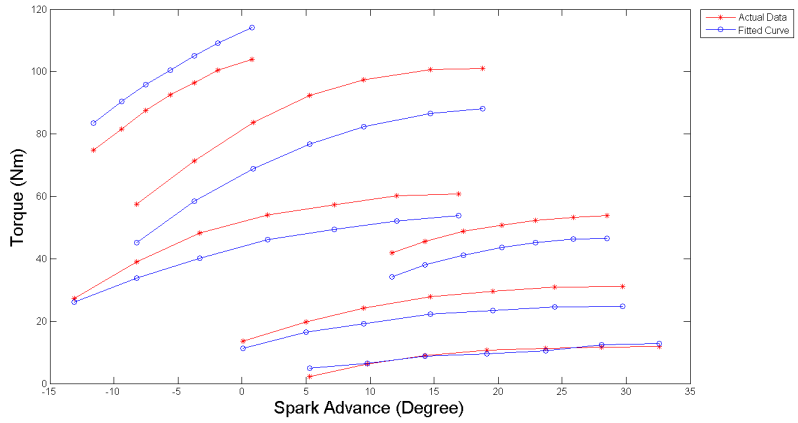


Figure 2.15: Fitted Profiles Calculated by the Varying Coefficient Model for 6 Validation Sweeps. These Profiles Were Provided by Grove *et al.* (2004) to Assess the Accuracy of the Model Developed in Training Stage. The Sum of Squares of the Error for This Model over All 6 Validation Profiles Is Equal to $2.82e+03$

2.3 Conclusion

This chapter provides an overview of the techniques for modeling functional data. Smoothing techniques, such as the use of basis systems, are flexible in modeling different shapes and local behaviors of response profiles. For designing experiments with functional responses, it is necessary to combine the smoothing model with other parametric regression models on experimental factors (covariates). We discussed two of such models, the mixed-effects model and varying coefficient model in this chapter.

These modeling techniques were demonstrated by applying them on the data sets from Nair *et al.* (2002) and Grove *et al.* (2004). Both models are equally flexible in fitting dynamic response data. The flexibility of these models depends on the chosen order of the bases and the location of knots. In the mixed effects model, the bases are fitted to the observations, $y(t)$, while in the varying coefficient model, the bases are fitted to the models' coefficients (parameters),$\beta(t)$.

An advantage of the mixed effects model is in its relative simplicity in estimation. On the other hand, the varying coefficient model provides somewhat smoother

response curves. The primary issue with the varying coefficient model is that an assumption must be made about the structure of the coefficient function with respect to the spectrum variable. In practice, it is difficult to make reasonable assumptions about these unknown coefficients beforehand, which makes this model impractical for constructing optimal experimental designs.

Chapter 3

OPTIMAL EXPERIMENTAL DESIGNS FOR LINEAR MODELS

Notational Conventions

$\mathbf{X}$	**Design Matrix**
$\mathbf{x_i}$	**i-th Row of the Design Matrix**
N	**Number of Design Points**
$\mathbf{C}$	**Candidate Matrix**
O	**Number of Candidate Points in the Candidate Matrix**
$\mathbf{c}$	**Candidate Point in the Candidate Matrix**
$\tilde{\mathbf{x}}_{i,j}$	**changing Coordinates in Row $\mathbf{x_i}$**
$\tilde{\mathbf{x}}_{i,-j}$	**Fixed Coordinates in Row $\mathbf{x_i}$**
$\mathbf{f}(\mathbf{x}_i)$	**Re-arranged Transpose of Row Vector $\mathbf{x}_i$**
$\mathbf{x^{(m)}}$	**Row Vector of a Design Point Expanded to the Model Form**
p	**Number of Model Parameters**
K	**Number of Clusters**
MX	**Cluster With the Largest Maximum Prediction Variance**
MU	**Cluster With the Smallest Maximum Prediction Variance**
$\mathbf{E}$	**Matrix of Evaluation Points Within the Design Space**
$\mathbf{A_l}$	**Possible Values to Replace the Current Coordinate Values of a Design Point That is Located in Cluster l**
t	**Number of Iterations**
d	**A Small Number for Expanding the Cluster Search**

The construction of experimental designs that are optimal with respect to some statistical criterion was proposed by Wald (1943). This topic was further developed by Kiefer in a series of papers that distinguished among these criteria using alphabets (Kiefer, 1959; Kiefer, 1959II; Kiefer, 1961). The alphabetic optimality criteria have since become the standard nomenclature in the field of optimal designs.

Optimal designs fall under two general categories namely, exact (discrete) and continuous (approximate) designs. Continuous designs consider the design region as a probability measure. The goals of the continuous optimal design problem are to

determine the minimum number of support points, the location of these points, and the weights at each support point. While methods for finding continuous designs are mathematically elegant, criticisms abound regarding this approach when applied to real cases. For example, suppose that the optimal design for a one-factor experiment is $\xi = y_1, y_2; w_1, w_2 = (0,0), (-1,1); 0.285, 0.715$, where (y_1, y_2) and (w_1, w_2) are respectively the support points and weights on these points. The weights are the relative proportion of replications on each of the two design points. It is sometimes impossible to find an economic sample size N so that these relative proportions hold exactly true. For this example, the optimal allocations would be $57, 143$ for a total sample size, $N = 200$. The key is to find an integer number of runs at each support point, which could be very large in magnitude. An alternative solution is to approximate the allocations for a specified number of runs, N. If the experimenter could only afford $N = 20$ runs, then the approximate allocation becomes $5.7, 14.3$. In practice, these numbers are rounded up or down. This approach does not guarantee that the approximate allocation is the optimal design for the specified number of runs.

On the other hand, exact designs begin by specifying the number of runs, N. Heuristic methods iterate on the objective function until no improvements could be made for this particular number of runs. Hence, in practical applications, exact designs are more popular. In this research, the focus is on the construction of exact optimal designs.

This chapter presents an introduction to algorithms and methods for finding exact designs for linear models. First, a short discussion about alphabetic optimality and a description of various optimality criteria are presented in Section 3.1. Exchange algorithms (EA), such as the Point Exchange Algorithm (PEA) and Coordinate Exchange Algorithm (CEA), are widely-used and will be discussed in the context of the more popular optimality criteria in Section 3.2. Section 3.3 presents one of the significant

contributions of this dissertation – a new method for finding G-optimal designs (Saleh and Pan (2014a)). The algorithms for finding optimal designs for functional data in subsequent chapters are inspired by the methods discussed in this chapter.

3.1 Alphabetic Optimality Criteria

The most popular optimality criteria, in theory and practice, is the D-optimality criterion proposed by Wald (1943). In the classical design theory, D-criterion aims to maximize the determinant of the information matrix. For linear models, the D-criterion becomes:

$$\max |X'X|$$

There are several reasons for its popularity. The D-criterion is useful in the process of screening out factors, especially in experiments where not much is known about factor effects. Aside from its usefulness in practice, D-optimal designs have also been shown to perform well for other design criteria. Finally, the D-criterion is more computationally efficient than most other design criteria because of its simple update equations resulting from the special characteristics of the determinant of the information matrix (Goos, 2002).

Some of the other widely-used design criteria include the A-criterion for screening and the G- and I-criteria for prediction. The A-criterion optimizes a simpler functional than the D-criterion namely, the trace of the inverse of the information matrix or:

$$\min Tr\{M^{-1}\}$$

A-optimality is favored because of its computational efficiency, but A-optimal designs usually do not perform well with respect to other design criteria. This is

because it only optimizes with respect to the diagonal elements of the information matrix, which are proportional to the variances of the coefficient estimates. However, it ignores the covariance structure among their estimates.

The I-criterion is widely used in experiments where the experimental response prediction within the design region is the primary objective. The I-criterion minimizes the average prediction variance in the design region χ:

$$\min \int_{\chi} f'(x) M^{-1} f(x) dx$$

With standard experimental regions (i.e., cubic, spherical), the I-criterion is easy to calculate. However, for non-standard regions in higher-dimensional space, evaluation of the integral becomes a non-trivial task.

3.2 Exchange Algorithms for Finding **D**, **A** and **I** Optimality

Fedorov (1969), and Fedorov (1971) introduced the Point Exchange Algorithm (PEA) for constructing exact D-optimals design for linear models. PEA is favored for its algorithmic simplicity, but its use in more complex optimality functions is deterred by its heavy computational load. To reduce the computational expense of PEA, enhancements were made on the original procedure by Mitchell and Miller Jr (1970), Wynn (1970), Mitchell (1974), Cook and Nachtrheim (1980), Atkinson and Donev (1989), Welch (1984), Nguyen and Miller (1992), Vahl and Milliken (2011), and Nguyen (1993). Some of these algorithms have been implemented in statistical software, such as Design Expert.

To explain how PEA works, suppose that a D-optimal design needs to be constructed in N runs. PEA starts with an initial design matrix and then iterates to substitute some or all of its rows (design points) with rows from a candidate matrix. To implement the algorithm, a set of K candidate points is enumerated to form a

candidate set or candidate matrix $\mathbf{C}$. The algorithm randomly selects N candidate points to construct the initial design matrix, $\mathbf{X}$. Next, a candidate point that maximizes $\mathbf{c}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{c}'$ is selected from the candidate set. Maximization of this update function results in maximizing the increase in the determinant of the information matrix through the addition of the candidate point. This point is added into the starting design matrix, expanding the number of rows to $N+1$. Every row vector from the initial design, $\mathbf{x}_i, i = 1, \ldots$ is a candidate for deletion. The algorithm iteratively deletes each of the row vector and recalculates $\mathbf{x}_i(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i'$ with the new candidate point. The design point deletion that results in the minimum value of $\mathbf{x}_i(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i'$ results in that point being removed from the design matrix. The algorithm continues to iterate until the value of Fedorov's delta function, $(1 - \mathbf{x}_i(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i')(1 + \mathbf{c}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{c}') - 1$, becomes very small.

The PEA above is a modified version of the proposals by Wynn (1970) and Mitchell and Miller Jr (1970). Mitchell (1974) proposed further computational improvements by allowing more than one row to be added or deleted from the design matrix at each iteration.

PEA obviously has some intrinsic drawbacks:

1. The size of the candidate matrix becomes enormous even for a moderate number of experimental factors. For instance, in an experiment with 4 factors with each factor having 100 possible values, the candidate set contains $O = 100^4$ candidate points. As a result, the computation time exponentially increases with the size of the candidate set.

2. The construction of an optimal design requires relatively expensive matrix computations. In the previous example, $10^6 + N$ matrix multiplications are required to find a candidate replacement for a current point in the design matrix.

The computational complexity of the PEA is NP-hard, as its run time increases exponentially with the increase in the number of model terms. For high-dimensional problems, it could also happen that the candidate matrix may become so large that it cannot be stored in a regular office computing machine. These notable disadvantages of the PEA motivated work on a more efficient algorithm that could run in polynomial time. In 1995, the Coordinate Exchange Algorithm (CEA) was proposed by Meyer and Nachtsheim (1995) for linear models. This algorithm starts with a feasible random design matrix. The variance functions of all rows (design points) in the design matrix are then calculated and sorted. The variance function of a single design point is calculated by:

$$\mathbf{v}(\mathbf{x_i}) = \mathbf{x}_i(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i', \tag{3.1}$$

where $\mathbf{X}$ is the design matrix of a linear model and $\mathbf{x}_i$ is the i^{th} row of $\mathbf{X}$. The rows with the smallest values of the variance function are selected for exchange. Similar to PEA, the variance function updates the determinant of the information matrix, and a small values implies that the point has a minimal impact on the value of the determinant. Following Meyer and Nachtsheim's notation, let $\tilde{\mathbf{x}}_{i,j}$ be the coordinates in $\mathbf{x}_i$ that will be modified and $\tilde{\mathbf{x}}_{i,-j}$ be the unchanged coordinates at each iteration. The changing coordinates will be replaced by $\tilde{\mathbf{x}}_j$, where $\tilde{\mathbf{x}}_j \in \mathbf{C}$ and $\mathbf{C}$ is the set of all possible values for these coordinates. Denote $\mathbf{f}(\mathbf{x}_i)$ as a vector that is a re-arranged transpose of the row vector $\mathbf{x}_i$ so that

$$\mathbf{f}(\mathbf{x}_i) = \begin{bmatrix} \mathbf{f}_1(\tilde{\mathbf{x}}_{i,j}) \\ \mathbf{f}_2(\tilde{\mathbf{x}}_{i,-j}) \end{bmatrix}$$

where $\mathbf{f}_1(\tilde{\mathbf{x}}_{i,j})$ and $\mathbf{f}_2(\tilde{\mathbf{x}}_{i,-j})$ correspond to the changing and fixed coordinates, respectively. In other words, the vector $\mathbf{f}(\mathbf{x}_i)$ is partitioned into two groups – changing and fixed coordinate vectors. A $p \times p$ matrix $\mathbf{A}$ is defined as

$$\mathbf{A} = [1 - v(\mathbf{x}_i)](\mathbf{X}'\mathbf{X})^{-1} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{f}(\mathbf{x}_i)\mathbf{f}'(\mathbf{x}_i)(\mathbf{X}'\mathbf{X})^{-1}. \tag{3.2}$$

$\mathbf{f}(\mathbf{x}_i)$, $\mathbf{A}$ can be partitioned into:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}, \tag{3.3}$$

$\mathbf{A}_{11}$ is a sub-matrix that involves the changing coordinates, while $\mathbf{A}_{22}$ involves the fixed coordinates. Replacing the changing coordinates $\tilde{\mathbf{x}}_{ij}$ by $\tilde{\mathbf{x}}_j$, the modified Fedorov delta function becomes:

$$\Delta_D^{ij}(\tilde{\mathbf{x}}_{ij}, \tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}_{i,-j}) = \mathbf{f}_1'(\tilde{\mathbf{x}}_j)\mathbf{A}_{11}\mathbf{f}_1(\tilde{\mathbf{x}}_j) + \mathbf{a}'\mathbf{f}_1(\tilde{\mathbf{x}}_j) + b, \tag{3.4}$$

where $\mathbf{a} = 2\mathbf{A}_{12}\mathbf{f}_2'(\tilde{\mathbf{x}}_{i,-j})$ and $b = \mathbf{f}_2'(\tilde{\mathbf{x}}_{i,-j})\mathbf{A}_{22}\mathbf{f}_2(\tilde{\mathbf{x}}_{i,-j}) + (1 - v(\mathbf{x}_i))$.

The candidate $\tilde{\mathbf{x}}_i$ that maximizes the delta function replaces the current coordinates $\tilde{\mathbf{x}}_{ij}$. These steps repeat for every coordinate from the selected rows until no more improvements are made in the delta function. It is also recommended that the algorithm iterates on multiple random starting designs due to the dependency of the final design on the selected starting random design. An obvious advantage of the CEA over the PEA is the absence of a set of candidate points, resulting in improved computational efficiency and reduced memory requirements.

Meyer and Nachtsheim (1995) also adapted the CEA for constructing optimal designs for other criteria, such as A- and I-optimality. Similar to the CEA for D-optimality, CEAs for these linear criteria start off by constructing a feasible randomly

generated design. Instead of the variance function in Equation (3.1), the deletion function is revised in Equation (3.5) to identify the rows for exchange.

$$d_L(\mathbf{x}_i) = \frac{\phi(\mathbf{x}_i)}{1 - \mathbf{v}(\mathbf{x}_i)}, \tag{3.5}$$

where $\phi(\mathbf{x}_i) = tr(\mathbf{D}\mathbf{f}(\mathbf{x}_i)\mathbf{f}'(\mathbf{x}_i)\mathbf{D})$, and $\mathbf{D} = (\mathbf{X}'\mathbf{X})^{-1}$ and $\mathbf{v}(\mathbf{x}_i)$ are similar to the parameters in Equation (3.1). The delta function becomes:

$$\Delta_L^{ij}(\tilde{x}_{ij}, \tilde{x}_j, \tilde{x}_{i,-j}) = \frac{\mathbf{f}_1'(\tilde{\mathbf{x}}_j)\mathbf{B}_{11}\mathbf{f}_(\tilde{\mathbf{x}}_j) + \mathbf{b}'\mathbf{f}_1(\tilde{\mathbf{x}}_j) + d}{\mathbf{f}_1'(\tilde{\mathbf{x}}_j)\mathbf{A}_{11}\mathbf{f}_1(\tilde{\mathbf{x}}_j) + \mathbf{a}'\mathbf{f}_1(\tilde{\mathbf{x}}_j) + c}, \tag{3.6}$$

$\mathbf{A}$, $\mathbf{a}$ and c are as given in Equation (3.2) and Equation (3.4). The $\mathbf{B}$ matrix is given by

$$\mathbf{B} = [1 - \mathbf{v}(\mathbf{x}_i)][\mathbf{I}_p + 2\mathbf{D}\mathbf{f}(\mathbf{x})\mathbf{f}'(\mathbf{x})]\mathbf{D}\mathbf{M}\mathbf{D} - \phi(\mathbf{x}_i)\mathbf{D}, \tag{3.7}$$

where for I-optimality, $\mathbf{M} = \int_{\mathbf{R}} \mathbf{f}(\mathbf{x})\mathbf{f}'(\mathbf{x})d\mathbf{x}$ and for A-optimality, $\mathbf{M} = I$. With respect to a partitioned $\mathbf{B}$ matrix, $\mathbf{b} = 2\mathbf{B}_{12}\mathbf{f}_2'(\tilde{\mathbf{x}}_{i,-j})$ and $\mathbf{d} = \mathbf{f}_2'(\tilde{\mathbf{x}}_{i,-j})\mathbf{B}_{22}\mathbf{f}_2(\tilde{\mathbf{x}}_{i,-j})$, which have constant values for each coordinate.

The coordinate exchange algorithm is a significant milestone in the development of optimal experimental design algorithms. The efficiency of this algorithm in finding exact optimal design for linear models inspired extensions to accommodate both Generalized Linear Models (see Gotwalt *et al.* (2009); Rodriguez *et al.* (2010); Saleh and Pan (2014b)) and models for mixture experiments (see Piepel *et al.* (2005)).

It is noteworthy that besides exchange algorithms, there are also proposals from mathematical optimization theory that have been used to construct optimal designs. Examples of these include simulated annealing (SA) (see Haines (1987); Meyer and

Nachtsheim (1988)), genetic algorithm (GA) (see Broudiscou *et al.* (1996); Heredia-Langner *et al.* (2003)), and an integer programming method (see Welch (1984)).

3.3 An Algorithm for Finding Exact G-Optimal Designs

In this section, we present a clustering-based Coordinate Exchange Algorithm (cCEA) for finding exact, G-optimal designs. We compared the performance of this algorithm with other proposals in literature with respect to G-efficiency and computational efficiency.

Suppose that the model of interest is a polynomial regression model given by $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, where $\mathbf{X}$ is the expanded design matrix or model matrix, $\boldsymbol{\beta}$ is the vector of parameters, and $\boldsymbol{\epsilon}$ is random error so that $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$. The variance-covariance matrix of the least squares estimator of the parameters is given by:

$$Cov(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}, \tag{3.8}$$

where $\mathbf{X}'\mathbf{X}$ is also known as the expected Fisher information matrix. For a design point inside the experimental design region, the variance of a predicted response is calculated as:

$$Var(\hat{y}(\mathbf{x})) = \sigma^2\mathbf{x}^{(m)}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}^{(m)\prime}, \tag{3.9}$$

where $\mathbf{x}^{(m)}$ is the row vector of a design point expanded to the model form. A scaled version of Equation (3.9) is often used in comparing designs to take into account the number of runs (N) and disregard the error variance:

$$SPV(\mathbf{x}) = \frac{NVar(\hat{y})}{\sigma^2} = N\mathbf{x}^{(m)}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}^{(m)\prime}. \tag{3.10}$$

A G-optimal design is a design that minimizes the maximum scaled prediction variance over the entire design space. That is, for a design ζ,

$$\text{Min}_{\zeta \in \Xi}[\text{Max}_{\mathbf{x} \in \mathcal{R}} SPV(\mathbf{x})], \tag{3.11}$$

where Ξ is the set of all possible designs and $\mathcal{R}$ is the design region. G-optimality considers the worst case scenario (the largest prediction variance) across the design space.

To generate an exact design, the common approach is to discretize the range of the experimental variables. For example, for 5 design variables each with 10 possible values, there are 10^5 distinct design points in the design space. To generate a design in 8 runs, there are $\begin{pmatrix} 10^5 \\ 8 \end{pmatrix}$ possible design combinations. An exhaustive search over this complete set of candidate designs to find the optimal is computationally tedious.

By the General Equivalence Theory (Pukelsheim, 1993), an approximate or continuous D-optimal design for a linear model is also G-optimal. While the GET is, in general, not applicable to exact optimal designs, Atwood (1969) showed that this property can be used to define G-efficiency:

$$G_{\text{efficiency}} = \frac{p}{\text{Max}_{\mathbf{x} \in \mathcal{R}} SPV(\mathbf{x})}. \tag{3.12}$$

where $0 \leq G_{\text{efficiency}} \leq 1$ and p is the number of model parameters.

Various exchange algorithms for constructing D-optimal designs have been implemented in commercial software. In particular, CEAs are efficient for high-dimensional design problems. However, due to the complexity of the G-optimality criterion, there is no established CEA proposals for G-optimal designs, with the exception of the method in Rodriguez *et al.* (2010). Further, there are no exchange algorithm propos-

als for finding G-optimal designs under GLMs. In the following section, CEAs for G-optimal design construction for linear models and GLMs are proposed.

3.3.1 The Clustering-Based Coordinate Exchange Algorithm (cCEA)

One of the obstacles in finding G-optimal design is the repeated evaluation of prediction variance function on a large set of candidate design points. Borkowski (2003) suggested evaluating the prediction variance function on at most 5^{p-1} design points, where each scaled independent variable could only take values from $\{0, \pm 0.5, \pm 1\}$. By this method, it is likely that the true maximum prediction variance is located outside the search grid, hence it under-optimizes the maximum prediction variance. To search through a larger design region, Borkowski (2003) proposed the genetic algorithm (GA). Rodriguez *et al.* (2010) suggested a coordinate exchange algorithm with Brent's optimization method, where the random starting design is optimal in some way (e.g, D-optimal or I-optimal). However, Brent's method is a local search method, so it could be trapped in a local optimal point.

The proposed algorithm in this section is a hybrid of the point exchange and the coordinate exchange algorithms. It proceeds in two steps. In the first step, we try to find a near G-optimal design by clustering and exchanging design points. In the second step, a coordinate exchange algorithm is applied on the initial design matrix obtained from the first step to generate a design with better G-optimality. In general, the first step can quickly identify the neighborhood in which the optimal design points are likely to be located. In this way, it relieves the exhaustive search requirement that results in decreased computational time. In the second step, the coordinate values of some design points are systematically adjusted within a limited range so as to further reduce the maximum prediction variance.

3.3.1.1 Step 1: Clustering and Exchanging Candidate Design Points

Let the design space be a hypercube and let the value of each independent variable vary from -1 to 1. To initiate design construction, a random design matrix, denoted by $\mathbf{X}$, is generated. The G-efficiency and the maximum prediction variance are then calculated for the current design matrix. A set of candidate design points, denoted by $\mathbf{C}$, are randomly generated. In order to cover the design space as much as possible, the candidate set must have a large number of candidate design points. For instance, for a linear model with 4 factors, we recommend a candidate set of at least 10^4 candidate design points. The G-optimal design is constructed by following these steps:

1. Cluster the candidate design points in $\mathbf{C}$ into K groups. The value of K may vary for different applications, but we recommend 5 to 10 clusters in order to create a diverse set of points while avoiding a big increase in computational burden.

2. Based on the current design matrix $\mathbf{X}$, evaluate the prediction variance on all candidate points and find the maximum prediction variance in each cluster. Denote the cluster with the largest maximum prediction variance as MX and the cluster with the smallest maximum prediction variance as MU.

3. Determine the cluster membership of each design point in the current design matrix. If there is no design point in MU, choose the cluster that has the next smallest maximum prediction variance and denote it as MU. Repeat this process until there is at least one design point in MU.

4. Remove one design point from the MU cluster, thus the design matrix is changed to an $(N-1) \times p$ matrix. Based on this modified design matrix, find the maximum prediction variance from the candidate points in MU and calculate

the change in maximum prediction variance. Repeat this process for all design points in the MU cluster. Finally, delete the design point that causes the minimal change in maximum prediction variance.

5. Use the modified design matrix to evaluate the prediction variance on each candidate point in the MX cluster. Add the candidate point that has the maximum prediction variance into the design matrix.

6. Repeat Steps 2-5 until no meaningful reduction in the maximum prediction variance of each cluster is observed.

Five to ten clusters are suggested because if there are too few clusters, the MU cluster may contain many design points, increasing the computational time for searching for the design point to be excluded. On the other hand, if there are too many clusters, it is likely that the MU cluster may not contain any design point, then the algorithm has to repeatedly search for a new MU, and this will also cause fewer candidate points in MX to be considered for replacing the point in MU and hinder the improvement of the design matrix after each iteration.

The proposed clustering and point exchange algorithm partitions the design space into several sub-regions and finds the sub-region that has the maximum prediction variance. Adding a new design point in this sub-region can greatly reduce the prediction variance in this local region. Concurrently, the algorithm identifies a design point from the current design matrix that does not induce a large change on the local prediction variance if this point is removed. Thus, it is better to exchange this design point with the candidate point in the large prediction variance sub-region.

Figure 3.1 shows the transition of a design point from one cluster to another cluster on a two-dimensional design space. Figures 3.1(a) and 3.1(b) are the scaled prediction variance graphs of the initial experimental design and the design after one iteration,

respectively. Figures 3.1(c) and 3.1(d) are their corresponding design spaces, where candidate points are clustered and design points are marked by black dots. Cluster 1 initially has very large scaled prediction variance (see Figure 3.1(a)), but after removing one design point in Cluster 2 and adding one design point in Cluster 1, the scaled prediction variance in Cluster 1 is significantly reduced (see Figure 3.1(b)).

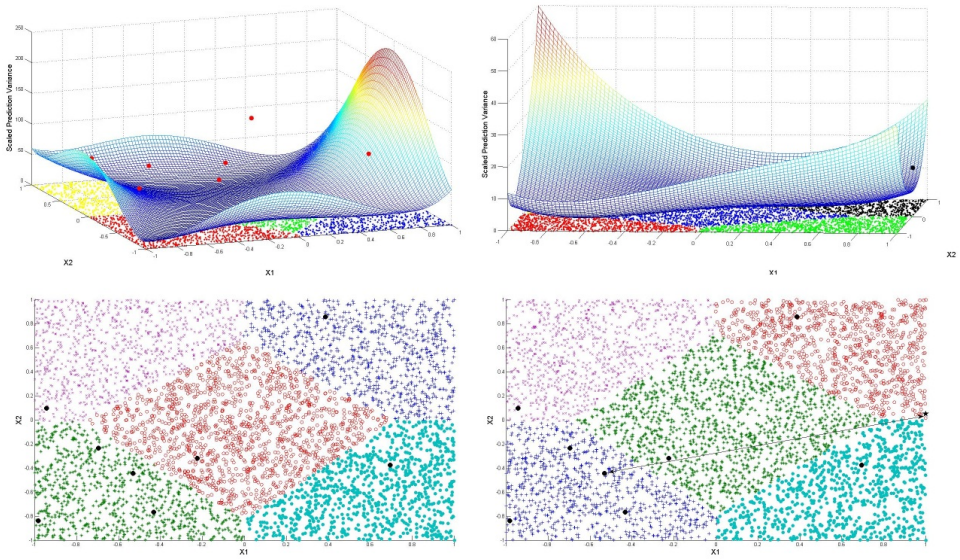


Figure 3.1: The Scaled Prediction Variance Plot and the Design Space Region That Is Divided into 5 Clusters. From the Left Panel to the Right Panel, One Design Point in the MU Cluster Is Replaced by a Candidate Point in the MX Cluster.

Although the proposed clustering and point exchange algorithm can generate a good design for a linear model, it does not guarantee finding the exact G-optimal design, especially for a complicated model such as response surface models and GLMs. Thus, we use the design matrix produced in this step as the initial matrix for further improvement.

3.3.1.2 Step 2: Coordinate Exchange of Design Matrix

A coordinate exchange algorithm is proposed to fine-tune the initial design matrix found in the point exchange step. Same as the first step, we will first partition the whole design space into several clusters, and then the coordinate ranges of these

clusters will be utilized to reduce the computational effort of the coordinate exchange step.

Unlike PEA, where current design points are replaced by candidate design points, CEA works directly on one or more coordinates of an existing design point. From the initial design matrix, the design points that need to be modified are first determined. As the G-optimality criterion is more sensitive to the design points that are located in the regions with large prediction variance, those design points will be modified.

We first randomly generate a large set of evaluation points within the design space. They are denoted as $\mathbf{E}$ and are clustered into k clusters. To increase diversity of evaluation points, $\mathbf{E}$ should be different from $\mathbf{C}$ used in the previous step. Set up a candidate value set $\mathcal{A}_l$, $l = 1, 2, ..., k$, for each cluster. The elements in $\mathcal{A}_l$ are possible values to replace the current coordinate values of a design point that is located in cluster l. The range of the candidate value set for the i^{th} variable (x_i) in cluster l is given by (Min_{il}, Max_{il}), where Min_{il} is defined by

$$Min_{il} = \max(\min(E_{il}), -1),$$

and Max_{il} is defined by

$$Max_{il} = \min(\max(E_{il}), 1),$$

where E_{il} denotes the i^{th} coordinate values of evaluation points in cluster l. Furthermore, we would widen this range slightly after each search iteration. At the t^{th} iteration, the ranges become

$$Min_{il} = max(min(E_{il}) - d \times (t-1), -1),$$

and

$$Max_{il} = min(max(E_{il}) + d \times (t-1), 1),$$

where d is a small number, such as 0.01. Figure 3.2 shows how the search range is widened after one iteration.

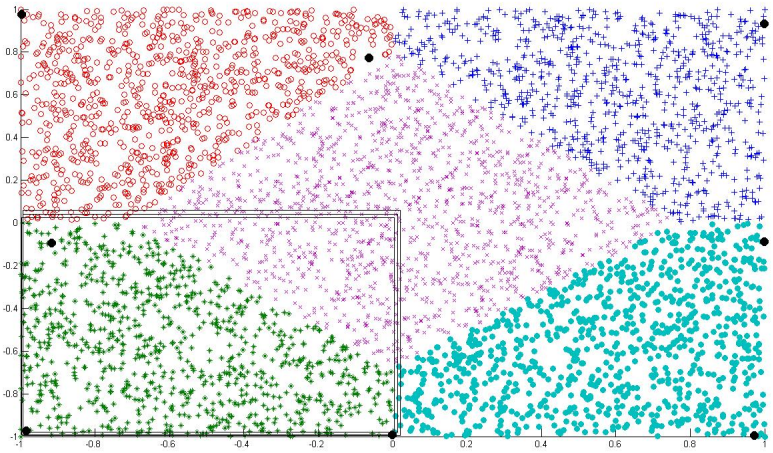


Figure 3.2: Expanding the Design Region after One Iteration

The proposed coordinate exchange algorithm consists of the following steps:

1. Cluster the evaluation points in $\mathbf{E}$ into k groups.

2. Determine the cluster membership of each of the current design points.

3. Order clusters based on their largest prediction variance values on the evaluation points in these clusters.

4. Consider the design points in the cluster with the largest prediction variance. Exchange the value of the first coordinate of one design point with an element in its candidate set $\mathcal{A}_{1l}$. Recalculate the maximum prediction variance of evaluation points. Find the element that can produce the largest reduction in maximum prediction variance and use it to replace the current coordinate value of the design point. When calculating the information matrix of a new design matrix, $\mathbf{X}'_{New}\mathbf{X}_{New}$, the inverse of information matrix can be updated by

$$(\mathbf{X}'_{New}\mathbf{X}_{New})^{-1} = \mathbf{M} - \mathbf{MB}(I + \mathbf{MB})^{-1}\mathbf{M} \tag{3.13}$$

with

$$\mathbf{M} = (\mathbf{X}'\mathbf{X})^{-1} \qquad \mathbf{B} = \mathbf{x}'^{*}_{i}\mathbf{x}^{*}_{i} - a \qquad a = \mathbf{x}'_{i}\mathbf{x}_{i},$$

where $\mathbf{x}_i$ is the row of the old design matrix to be modified and $\mathbf{x}_i^*$ is the new row in the new design matrix. Note that $\mathbf{M}$ and a are constants within each iteration.

5. Repeat Step 4 for all coordinates of all design points in this cluster, until no further improvement in reducing the maximum prediction variance can be achieved.

6. Compute the Euclidian distance of new points to k cluster centroids, and reassign their cluster memberships.

7. Monitor the reduction of maximum prediction variance on the evaluation points. If no meaningful reduction can be obtained, stop; otherwise, begin to modify the coordinate values of the design points in the cluster with the next largest prediction variance.

An example of G-optimal design points obtained by the coordinate exchange algorithm is shown in Figure 3.3, along the initial design points obtained by the clustering and point exchange algorithm. One can see that during the step of coordinate exchange, 3 initial design points are modified and 5 initial design points remain unchanged. Among those modified points, 2 of them still belong to their initial clusters and one design point has moved to its neighbor cluster.

Because the candidate and evaluation sets are randomly selected, it is recommended that both PEA and CEA steps should be repeated more than 10 times. Notice that the proposed algorithm cannot be applied to saturated or supersaturated designs due to the singularity of the information matrix from those designs.

The computational complexity of our algorithm depends on the size of candidate set, the number of rows of design matrix and the number of clusters. In the point

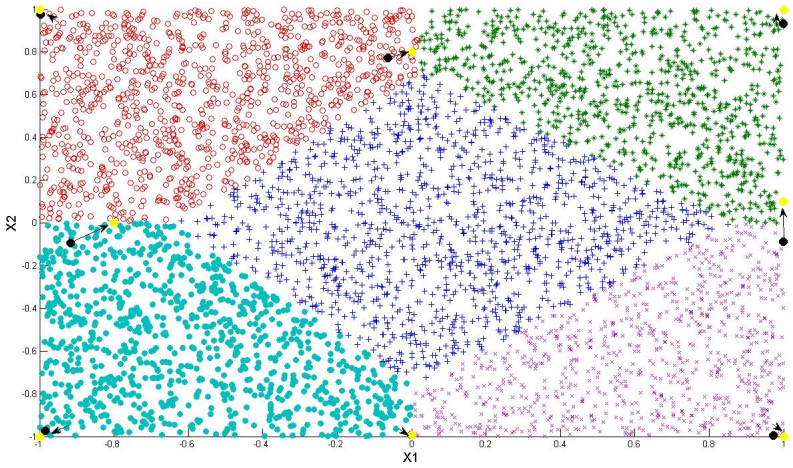


Figure 3.3: Compare the Initial Design from the First Step and the Final Optimal Design Generated by cCEA.

exchange step of cCEA, let C be the number of elements in the candidate set, K and N be the number of clusters and the number of rows of design matrix, respectively. We assume that C is significantly larger than N, and N is larger or equal to the number of columns of design matrix. The computational complexity of k-means clustering with C candidate points is $O(CKN)$. The complexity of calculating the inverse of the information matrix, $(X'X)^{-1}$, is $O(N^3)$ and the complexity of calculating prediction variances is $O(C^2N)$. Sorting out the maximum or the minimum prediction variance requires $O(C)$ order of time. In the worst case scenario, $N-1$ design points may fall into the cluster MU and all of these points need to be evaluated to find the row with the least effect on the prediction variance; then, the computational complexity of this step equals to $O(C^2N^2)$. Because these operations are sequential rather than nested, the total computational requirement is the sum of individual operation requirements; therefore, the computation of the point exchange step is in the polynomial order of C, K and N. Similarly, it can be shown that the computation of the coordinate exchange step of cCEA is solvable in the polynomial order of the number of candidate points,

the size of design matrix and the size of possible coordinate values. In summary, the cCEA algorithm is solvable in polynomial time.

3.4 Results

We compare the performance of the G-optimal designs generated by the cCEA in Tables 3.4 to 3.4 with the designs generated in Borkowski (2003) and Rodriguez *et al.* (2010) for linear models with 2 to 5 factors and where the desired number of runs varies from 7 to 26 design points. We also show that the cCEA is superior with respect to computing times.

Table 3.1: G-Optimal Designs for Experiments with Two Factors

N	x_1	x_2
1	0.063523	0.126625
2	0.995599	0.063243
3	-0.99941	0.959794
4	-1	-0.9092
5	0.225831	0.990961
6	0.799327	-1
7	0.998935	0.978566
1	1.11E-16	0.8
2	-0.79957	0
3	-1	-1
4	-1	1
5	1	1
6	0.99949	0.1
7	1	-1
8	0	-0.99268
1	0.028336	0.998537
2	0.1	-0.2
3	-1	0.988177
4	0.997797	-1
5	-1	-1
6	0.990061	0.031361
7	-0.99996	0.1
8	0.997289	0.973432
9	-0.10749	-0.99939
1	0.4	-0.8
2	-0.5	-0.08299
3	-0.2	0.981516
4	0.992719	0.948816
5	-1	-0.8
6	-1	0.964564
7	0.999772	0.201816
8	-0.99874	0.1
9	0.994314	-0.97379
10	-1	-1

N	x_1	x_2
1	-1	-0.6
2	0.7	-0.8
3	-1	0.322269
4	-0.1	0.1
5	1	-1
6	1	0.3
7	-1	1
8	-0.9	-1
9	1	0.971705
10	-0.2	-0.9
11	-0.01374	1
1	-0.8	-0.9
2	1	0.2
3	-0.6	0.6
4	0.1	-0.34376
5	-0.95982	-0.99399
6	0.317641	1
7	1	-1
8	1	1
9	-0.99955	0.248549
10	-1	1
11	0.2	-0.6
12	1	-1

Table 3.2: G-Optimal Designs for Experiments with Three Factors

N	x_1	x_2	x_3
1	0.596941	1	-1
2	0.9	-0.9	0.998856
3	0.996941	0.060626	-0.2
4	-1	-0.6	-1
5	-0.1	-1	-0.3
6	0.9	1	0.8
7	-0.2	0.1	0.904886
8	-0.9	1	-0.69712
9	-1	-1	0.854886
10	-0.9	0.9	0.9
11	0.9518	-0.82285	-1
1	-0.26109	-0.28743	0.050025
2	0.809143	-1	-0.8
3	0.832277	-1	0.906175
4	0.968781	0.457014	-1
5	8.33E-17	1	-1
6	-0.9	-0.9	-1
7	-1	0.743678	-0.7
8	0.944488	0.994064	0.016065
9	-0.98765	-0.89602	0.888092
10	-1	0.957466	0.934525
11	0.944488	0.155501	0.966065
12	0.144488	0.855501	0.991799
1	0	-1	-1
2	0.9	0.995178	-1
3	-1	1	0.8
4	0.973954	-1	0.916209
5	1	-0.1	-0.1
6	-1	-0.8	1
7	-0.1427	0.995178	-0.3
8	-1	0.70495	-1
9	-0.03113	0.094547	1
10	1	0.972501	0.910111
11	-0.7	-1	0.1
12	1	-0.9	-0.99181
13	-1	-0.69644	-0.9
1	0	0	-1
2	-1	0.05	1
3	-1	1	1
4	-0.05	-1	0
5	0.15	1	1
6	-1	1	0.05
7	-1	-1	-1
8	-1	0.95	-1
9	1	-1	1
10	1	1	-1
11	1	1	1
12	-1	-1	1
13	1	0	0
14	1	-1	-1
1	-0.2	1	-0.1
2	0.05	-1	0
3	1	-1	1
4	1	-1	-1
5	1	0.95	1
6	-1	1	1
7	-1	-1	1
8	0.05	-0.3	-1
9	-0.2	0.05	1
10	1	0	0.2
11	-1	-1	-1
12	0.95	1	-1
13	0.9	0.9	0.05
14	-1	-0.1	0
15	-1	1	-1
1	-0.1	1	0
2	0.15	-0.05	1
3	-0.45	-1	0
4	1	-1	1
5	-1	1	1
6	1	-1	-1
7	1	1	-1
8	-1	-1	-1
9	1	0	0.15
10	0.15	-0.8	0.1
11	-1	-1	1
12	0.1	0.2	-1
13	1	-0.9	-1
14	1	1	1
15	-1	1	-1
16	-1	0.05	0

Table 3.3: G-Optimal Designs for Experiments with Four Factors

N	x_1	x_2	x_3	x_4
1	-0.19526	0.974286	0.009128	-1
2	-1	-0.8	0.910656	-1
3	-1	0.959887	-0.6	0.9584
4	-1	0.674286	-1	-1
5	0.988471	-1	1	0.15845
6	0.913718	0.795782	0.990158	-1
7	0.275092	-1	-1	-1
8	-1	-1	-0.6	0.2
9	0.8	0.859887	0.6	0.9584
10	0.975534	-1	-0.9	0.95845
11	0.875092	-0.1	-0.7	-0.5
12	0.975092	1	-1	-0.4
13	-0.1	0.459887	-1	0.4584
14	0.875092	-0.5	-0.6	-0.6
15	-0.7	-0.6	0.967297	0.996628
16	-1	1	1	-0.04026
1	0.5	-1	1	1
2	1	-1	-0.65	0.95
3	0.9	1	-1	0.35
4	1	1	0.95	-0.4
5	-1	-1	1	-1
6	-1	-0.85	0.9	1
7	0.05	1	0	1
8	-1	0.9	0.95	0.85
9	1	-0.8	0.8	0.35
10	-1	1	-0.95	1
11	-1	1	0.95	-1
12	-1	-1	-0.1	-1
13	0	0.05	0.95	-1
14	0.9	1	-1	-0.95
15	-0.05	-1	-1	-0.2
16	0.95	0.65	0.45	-1
17	1	0.05	-1	1
18	1	-1	-1	-1
19	-0.9	-0.05	0.5	0
20	1	-0.9	0.85	-1
21	1	1	1	1
22	-1	-0.25	-1	-0.95
23	-0.95	1	-0.85	-1
24	-1	-1	-1	1

Table 3.4: G-Optimal Designs for Experiments with Five Factors

N	x_1	x_2	x_3	x_4	x_5
1	-0.3	0.9	-0.5	-0.9	-0.5
2	0.2	0.083461	-1	0.2	0.4
3	-1	-1	1	0.727208	0.516052
4	0.901536	-1	-0.4	0.835002	-0.00043
5	-0.7	0.8	-1	-1	0.779079
6	-0.6	0.619248	1.11E-16	0.6	1
7	-1	0.3	-1	0.8	-1
8	1	0.744798	1	-1	-0.9
9	-1	-0.4	-0.3	-1	-0.8
10	-1	0.971427	0.9	-0.21703	-1
11	1	1	-0.6	0.5	-1
12	0.9	-1	0.613798	-1	0.3
13	0.5	0.971427	0.9	0.982974	-1
14	1	1	-0.6	-0.9	1
15	0.973318	1	1	0.5	0.622308
16	-0.8	1	-0.4	1	0.606144
17	0.964851	-1	-1	-1	-0.99254
18	0.364851	-0.6	-0.1	-1	1
19	-0.9	-1	-0.8989	-0.22279	0.966052
20	-0.1	-1	-0.8	0.985007	-1
21	1	-0.5	0.717211	0.3	-1
22	0.723318	-0.8	0.954698	0.9	1
23	-0.6	-1	1	-1	-1
24	-1	-0.22857	1	0.982974	-0.7
25	-0.9	0.513433	1	-1	0.979079
26	1	-0.26654	-0.8	1	1

Table 3.5 gives the G-efficiencies of Brokowski's, Rodriguez et al.'s, and cCEA designs. The cCEA method generally provides designs with higher G-efficiencies. It is also noteworthy to explore the I-efficiencies of the G-optimal designs. The G-criterion minimizes the maximum prediction variance in lieu of the average prediction variance (APV). Hence, its APV value is expected to be larger than the APV value of an I-optimal design. The I-criterion optimizes with respect to the prediction variance in the entire region of the design space. To compare the cCEA-generated G-optimal and an I-optimal design with respect to the I-criterion, we utilize the fraction of design space (FDS) plot developed by Zahran *et al.* (2003). The FDS plot shows the proportion of design space that has a prediction variance less than or equal to a specific value. It therefore provides a comprehensive assessment of the behavior of the

prediction variance over the whole experimental region. A good design is expected to produce a lower and flatter prediction variance profile. Figure 3.4 shows that the prediction variance profile of the G-optimal design is flatter than that of the I-optimal design for the case with 2 factors and 7 design points. Compared to the I-optimal design, the values of the prediction variance are more uniform for the G-optimal design.

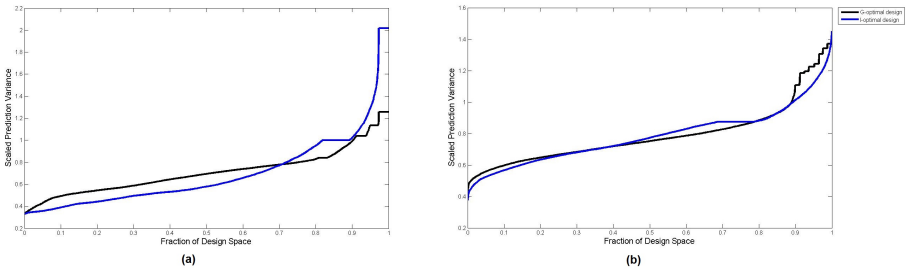


Figure 3.4: The FDS Plots of G-Optimal Design and I-Optimal Design for the Linear Model with 2 Factors and 7 Design Points

In Table 3.6, we compare the computation time of the cCEA algorithm with the algorithm used in Rodriguez *et al.* (2010). Both algorithms were implemented on a computer with Intel Core i5, 2.40 GHz. The computation times of Rodriguez et al.'s algorithm were obtained by running the author's JMP script code. According to the author, the search should be iterated using at least 100 random initial designs to avoid premature convergence to a local optimum. In all cases explored, the cCEA yielded lower computation time. Figure 3.5 plots the computation time versus the size of design for both Rodriguez *et al.* (2010)'s algorithm and the cCEA. It shows that the computation time of Rodriguez et al.'s algorithm increases exponentially when the size of the problem increases, while the cCEA only increases in a polynomial fashion.

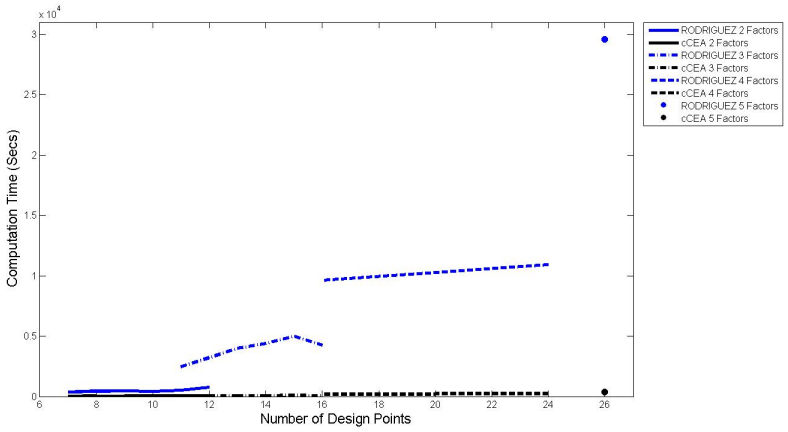


Figure 3.5: Plot of Computation Time vs. The Size of Design for Different Algorithms.

3.5 Conclusion

In this chapter, we gave an overview of computer algorithms for generating optimal experimental designs. We proposed a new computer algorithm, cCEA, to generate G-optimal designs for linear models. G-optimal designs are typically tedious to construct because of the repeated evaluation of the G-criterion over a sufficiently large number of points in the design region. The proposed algorithm is a hybrid of the point and coordinate exchange algorithms that uses a k-means clustering method to reduce the number of evaluations of the G-optimality objective function. This alleviates some of the computational burden so that even as the size of the design matrix increases, the objective function is still solvable in polynomial time. To demonstrate the effectiveness of the cCEA, a comparison was made with existing methods in the literature. We found that our proposed algorithm outperformed existing methods with respect to G-efficiency. More importantly, the G-optimal design was also found to have significantly lower computing times especially for larger problems. The cCEA-generated G-optimal design also performed well with respect to the I-criterion.

Table 3.5: Compare the G-Efficiencies of Optimal Designs Generated by Different Algorithms.

		G-efficiencies			
Number of factors	Number of design points	Brokowski's design	Rodriguez et al.'s design	cCEA design	G-optimal design APV
2	7	0.8	0.8	0.854	0.67
2	8	0.879	0.879	0.913	0.59
2	9	0.863	0.863	0.888	0.43
2	10	0.859	0.859	0.861	0.49
2	11	0.861	0.861	0.864	0.41
2	12	0.848	0.848	0.855	0.37
3	11	0.772	0.772	0.887	0.74
3	12	0.801	0.801	0.868	0.69
3	13	0.836	0.836	0.913	0.6
3	14	0.893	0.893	0.889	0.54
3	15	0.839	0.839	0.863	0.52
3	16	0.793	0.793	0.838	0.34
4	16	-	.729	.772	0.84
4	24	-	0.811	0.846	0.44
5	26	-	0.778	0.809	0.66

Table 3.6: Compare the Computation Times (in Seconds) of the cCEA with Rodriguez et apl.'s Algorithm with One Run.

Number of factors	Number of design points	Rodriguez et al.'s algorithm	cCEA
2	7	359	6
2	8	443	32
2	9	467	32
2	10	420	64
2	11	507	42
2	12	793	66
3	11	67	40
3	12	3220	49
3	13	3981	38
3	14	4398	42
3	15	4986	119
3	16	4247	54
4	16	9622	193
4	24	10953	247
5	26	29580	368

Chapter 4

OPTIMAL EXPERIMENTAL DESIGNS FOR FMRI EXPERIMENTS

Notational Conventions

Q	**Number of Stimuli in fMRI Experiment**
$\mathbf{d}$	**Stimulus Sequence**
d_i	**The i-th Element of Stimulus Sequence**
τ_{ISI}	**Interval Time Between Two Consecutive Stimuli Demonstration**
τ_{TR}	**Interval Time Between Two Consecutive fMRI Scans**
τ_{dur}	**Duration of One HRF**
$\mathbf{X_q}$	**Design Matrix for q-th Stimulus**
K	**Number of Columns in $\mathbf{X_q}$**
$\mathbf{X}$	**Concatenation of All $\mathbf{X_q}$'s**
$\mathbf{h_q}$	**Height Vector for q-th Stimulus**
$\mathbf{h}$	**Concatenation of All $\mathbf{h_q}$'s**
$\delta_\mathbf{i}$	**The Time Point When Stimulus Type i Appears in the Design Dequence d**
$\mathbf{D_i}$	**Modification Matrix**
ΔT	**The largest denominator that makes both $\tau_{ISI}/\Delta T$ and $\tau_{TR}/\Delta T$ integers**
$\mathbf{h}^*$	**The Basis Matrix for the Detection Problem**
$\theta_\mathbf{q}$	**Magnitude in Detection Problem for the q-th Stimulus**
$\mathbf{M_h}$	**Moment Matrix for the fMRI Estimation Problem**
$\mathbf{M}_\theta$	**Moment Matrix for the fMRI Detection Problem**
$\mathbf{O}$	**Matrix of Zeros**
Δ_E	**Fedorov Delta Function for The Estimation Problem**
Δ_D	**Fedorov Delta Function for The Detection Problem**

Functional magnetic resonance imaging (fMRI) is a powerful procedure for obtaining information about how human brains work. In a typical fMRI study, a subject is presented a sequence of stimuli. Information about brain activity are collected over a short period of time after the presentation of a stimulus. Because fMRI studies involve human subjects, efficient experimentation is of paramount importance. As mentioned in Chapter 2.2.1, the main concern in fMRI studies is in determining the

optimal sequence of the presentation of stimuli to experimental subjects. In this chapter, we focus on this particular case of optimal designs for functional data.

Two fast computer algorithms for finding D-optimal designs for fMRI studies are proposed in this chapter for the estimation and detection problems of fMRI experiments. The fMRI data are modeled using linear models such as the mixed-effects model, but the construction of the fMRI design matrix is different from conventional linear design problems. The proposed algorithms are based on a greedy search along each dimension in the design region. This problem has been solved in the literature using genetic algorithm (GA). For example, see Kao *et al.* (2009); Kubilius *et al.* (2011); Eck *et al.* (2013); Mijović *et al.* (2014). As a general optimization method, the GA technique is sometimes criticized for being inefficient due to its stochastic nature, particularly when the size of the optimization problem becomes large (see Romeijn and Pardalos (2002)). The proposed algorithm promises to improve the computational efficiency of the GA. Further, it handles a common constraint in fMRI studies – where every stimulus type appears in the design a specified number times – which the GA cannot accommodate.

To accomplish this, we look at the literature on exchange algorithms for linear models. The two most widely-used exchange algorithms namely, PEA and CEA, were discussed in Chapter 3. CEA has been lauded for its computational efficiency and flexibility to handle a wide variety of design problems (see, e.g., Gotwalt *et al.* (2009), Goos and Jones (2011)). However, adapting the CEA to the special case of fMRI design problems is not trivial. One major difficulty lies in the special structure of the fMRI design matrix, which will be shown in this chapter. By working on the specific nuances of this design matrix, we derive useful updating formulas that significantly reduce the computational burden when calculating the D-criterion values.

The usefulness of our proposals is demonstrated in our test cases for both single- and multi-objective fMRI design problems.

This chapter proceeds as follows. Section 4.1 discusses fMRI studies in the context of experimental designs. More specifically, we discuss the primary objectives of the optimal design problem and the fMRI experimental method. Section 4.2 reviews the linear models that will be used for the two major objectives – estimation and detection. Section 4.3 defines the design matrix for fMRI data and explains how these matrices are constructed. The proposed algorithms for finding D-optimal fMRI designs for the estimation, detection and multi-objective optimization problems are discussed in detail in Section 4.5. Finally, in Section 4.6, we compare the performance of the proposed algorithm to other approaches in the literature.

4.1 Experimental Design of fMRI

An fMRI study is conducted by measuring the increase or decrease of neural activities in the voxels of the brain due to the presentation of different stimuli to a patient at specific time points. The neuronal activities in the brain result in an increased (decreased) cerebral blood flow to the brain area of interest. The increase in blood flow increases the ratio of oxygenated hemoglobin to deoxygenated hemoglobin at specific voxels. The difference between the magnetic properties of oxygenated and deoxygenated hemoglobin is the basis for the Hemodynamic Response Function (HRF). Figure 4.1 shows a sample of an HRF resulting from the presentation of one stimulus that could have a lasting (carry-over) effect of 32 seconds. This duration is denoted by τ_{dur}.

Studies have shown that HRF's accumulate additively when multiple stimuli are presented to a patient and the interval time between two consecutive demonstration exceeds 2 seconds (see Lindquist *et al.* (2008)). In other words, the amplitude of the

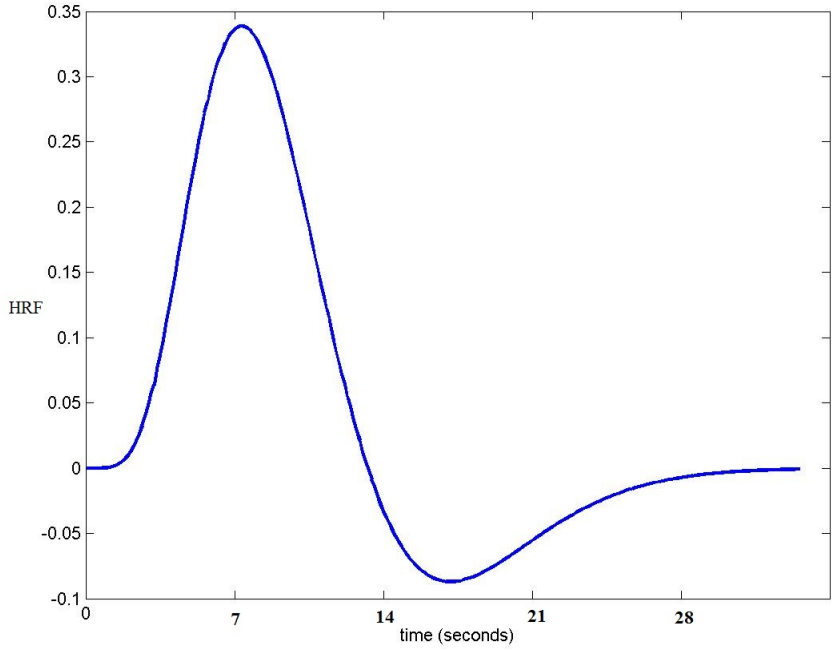


Figure 4.1: Single Hemodynamic Response Function, Stimulated by Only One Stimulus

HRF increases when several stimuli are presented in accelerated succession. Figure 4.2 shows three HRFs (broken curves) resulting from the presentation of one stimulus at different time points. This graph highlights the additive property of the fMRI signal.

An fMRI experiment is conducted by presenting each stimulus type, $q = 1, 2, \ldots Q$, to a patient at time $(j-1)\tau_{ISI}$, $j = 1, ..., N$. τ_{ISI} denotes the pre-specified interstimulus interval or the minimum time between the presentation of consecutive stimuli. Numerous studies have been conducted on using fixed τ_{ISI} of at least 15 seconds in order to achieve a design with higher statistical efficiency or power. However, Dale (1999) has demonstrated the high statistical efficiency of using τ_{ISI} as short as $500ms$ by quantitative analysis, especially if τ_{ISI} is properly jittered from trial to trial.

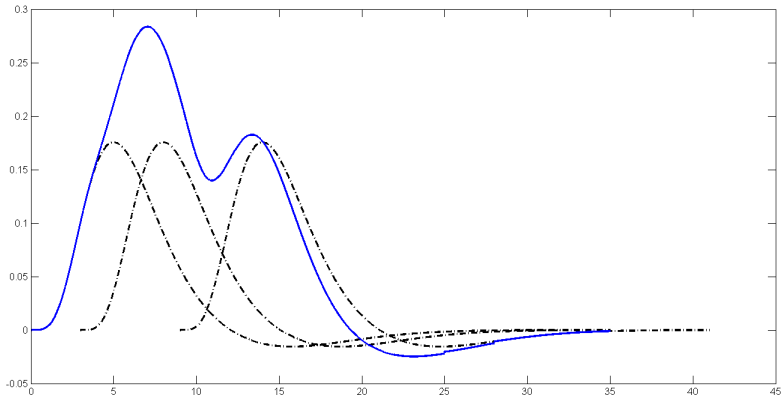


Figure 4.2: Three Hemodynamic Response Function (Broken Curves) Resulted by Demonstrating One Stimulus in Different Time Points Form the Accumulated HRF (Blue Curves).

HRF signals in one voxel are measured by the MR scanner every τ_{TR}. An fMRI experimental design is a sequence of stimuli, such as in Figure 4.1.

$$\begin{array}{lccccccc} \mathbf{j} & \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{N=7} \\ \mathbf{d} = \{1 & 2 & 0 & 2 & 1 & 1 & 0\}, & \\ \mathbf{Time} & \mathbf{0} & \tau_{ISI} & \mathbf{2}\tau_{ISI} & \mathbf{3}\tau_{ISI} & \mathbf{4}\tau_{ISI} & \mathbf{5}\tau_{ISI} & \mathbf{6}\tau_{ISI} \end{array}$$

The j^{th} entry of this sequence is the stimulus type shown to the patient at time $(j-1)\tau_{ISI}$, $j = 1, ..., 7$.

As mentioned in Chapter 2.2.1, there are two useful pieces of information obtained from fMRI studies. First is the HRF height produced by a stimulus type. Because the MR scanner provides accumulated HRF's (in lieu of individual HRF's), the goal is to estimate the individual HRF curves induced by a stimulus accurately. This problem is also known as "estimation" in fMRI literature. The second primary objective is to detect the activated brain voxels using the HRF magnitude. This is referred to as the detection problem. These two objectives will be explored in more detail in the next section.

4.2 Linear Models for Estimation and Detection Problems

For the estimation problem in fMRI studies, the following model has been proposed (see Liu (2004)):

$$\mathbf{y} = \sum_{q=1}^{Q} \mathbf{X}_q \mathbf{h}_q + \mathbf{S}\gamma + \mathbf{e}. \tag{4.1}$$

The response vector, $\mathbf{y} = (y_1, ..., y_T)'$, contains information about the signals collected in the MR scan of a voxel. The $y_{j,1}$ entry of vector $\mathbf{y}$ is the j^{th} height of the accumulated HRF signal at time $(j-1)\tau_{TR}$. The length of this time series, T, depends on the experimental duration and the scanning frequency of the fMRI machine. If an fMRI experiment has N stimuli demonstrations (including the rest period), then $T = N\tau_{ISI}/m_{TR}$ where $m_{TR} = \tau_{TR}/\Delta T$. The discretization interval ΔT is the largest denominator that makes both $\tau_{ISI}/\Delta T$ and $\tau_{TR}/\Delta T$ integers. ΔT may or may not be an integer.

The parameter vector $\mathbf{h}_q = (h_{q,1}, ..., h_{q,K})'$ in Equation (4.1) represents the heights of the discretized HRF for the q-th stimulus type, where $q = 1, ..., Q$. More specifically, $h_{q,j}, j = 1, ..., K$ represents the unknown HRF height at time $(j-1)\Delta T$ for stimulus q. K can be also defined as the number of discretization intervals for estimating the HRF curves, where $K = 1 + \lfloor \tau_{dur}/\Delta T \rfloor$, $\lfloor \rfloor$ is the flooring function, and τ_{dur} is the duration of an HRF signal. A notable aspect of the HRF curves is presence of carry-over or lasting effects of a stimulus, denoted by τ_{dur}. This implies that the first stimulus still affects the HRF height at the MR scan of the second stimulus. The effect of a single stimulus may last, say, $\tau_{dur} = 32$ seconds. In this case, if $\tau_{ISI} = 4$ and $\tau_{TR} = 2$, then $\Delta T = 2$ and $K = 17$. The matrix $\mathbf{X}_q$ in Equation (4.1) is a T-by-K matrix filled with 1's and 0's. Its (t, k) element indicates whether or not the

k-th height of the HRF of the q-th stimulus type contributes to the t-th fMRI signal, y_t. Deriving the $\mathbf{X}_q$ will be discussed with more details in the next section.

Finally, the nuisance term $\mathbf{S}\boldsymbol{\gamma}$ models the frequently observed trend/drift of the fMRI time series. Here, $\mathbf{S}$ is a specific matrix and $\boldsymbol{\gamma}$ is the corresponding parameter vector. Vector $\mathbf{e}$ in Equation (4.1) is a T-by-1 noise vector with mean $\mathbf{0}$ and $cov(\mathbf{e}) = \sigma^2\mathbf{R}$ for some $\sigma^2 > 0$ and a positive definite matrix $\mathbf{R}$.

In the estimation problem, the heights of the discretized HRF of each stimulus type are of interest in estimation. An experimental design that minimizes the general variance of the least-square estimate of $\mathbf{h} = (\mathbf{h}_1', ..., \mathbf{h}_Q')'$, is desired.

The detection problem, on the other hand, aims to identify which stimulus type activates brain voxels. For this problem, the following linear model is often used:

$$\mathbf{y} = \sum_{q=1}^{Q} \mathbf{X}_q \mathbf{h}^* \theta_q + \mathbf{S}\gamma + \mathbf{e}. \tag{4.2}$$

In this model, the HRF of the q-th stimulus type is modeled by an specified and assumed-to-be-known HRF shape, $\mathbf{h}^*$, with an unknown magnitude, θ_q. The other terms in Equation (4.2) are defined as in Equation (4.1). For the detection problem, the parameter vector $\boldsymbol{\theta} = (\theta_1, ..., \theta_Q)'$ is of interest, and it indicates the strength of brain activation due to each stimulus type.

For the model in Equation (4.1), the information matrix for $\mathbf{h}$ is given by Liu and Frank (2004):

$$\mathbf{M}_h = \mathbf{X}'\mathbf{V}'(\mathbf{I}_T - \mathbf{P}_{\mathbf{VS}})\mathbf{V}\mathbf{X}, \tag{4.3}$$

where $\mathbf{X}$ is the concatenation of all $\mathbf{X}_q$'s (i.e., $\mathbf{X} = [\mathbf{X}_1 ... \mathbf{X}_Q]$), $\mathbf{V}$ is a matrix such that $\mathbf{V}\mathbf{R}\mathbf{V}' = \mathbf{I}_T$ with $\mathbf{I}_T$ being the T-by-T identity matrix, and $\mathbf{P}_{\mathbf{VS}} = \mathbf{VS}(\mathbf{S}'\mathbf{V}'\mathbf{VS})^{-1}\mathbf{S}'\mathbf{V}'$ is the orthogonal projection matrix onto the column space of

VS. For the detection problem using model (4.2), the information matrix of $\boldsymbol{\theta}$ is given by:

$$\mathbf{M}_\theta = (\mathbf{I}_Q \otimes \mathbf{h}^*)'\mathbf{X}'\mathbf{V}'(\mathbf{I}_T - \mathbf{P}_{\mathbf{VS}})\mathbf{V}\mathbf{X}(\mathbf{I}_Q \otimes \mathbf{h}^*), \tag{4.4}$$

where $\otimes$ denotes the Kronecker product.

To find the D-optimal fMRI design, the objective is to maximize the determinant of the information matrix $\mathbf{M}_h$ or $\mathbf{M}_\theta$. The information matrix depends on the design $\mathbf{d}$ through the design matrix $\mathbf{X}$, which is the concatenation of $\mathbf{X}_q$'s, $q = 1, 2, ..., Q$. Systematically constructing $\mathbf{X}_q$ will be discussed in the next section.

4.3 Design Matrix Construction for fMRI Problem

The construction of the design matrices for fMRI experiments depend on whether the times of the onsets of stimuli and the scanning times of fMRI equipment are synchronized ($\tau_{ISI} = k\tau_{TR}$) or not synchronized ($\tau_{ISI} \neq k\tau_{TR}$). For both conditions, the design matrix ($\mathbf{X}$) concatenates the design matrices from each stimulus ($\mathbf{X_q}, q = 1, ..., Q$), i.e. , $\mathbf{X} = [\mathbf{X_1}\mathbf{X_2}, ..., \mathbf{X_Q}]$, where the j^{th} row of the design matrix shows the status of the system at time $(j-1)\tau_{TR}$.

When $\tau_{ISI} = \tau_{TR}$, the number of rows in the design matrix is equal to N, where again N is the number of elements in the fMRI design sequence ($\mathbf{d}$). The number of columns in the matrix $\mathbf{X_q}$ becomes $1 + \lfloor \tau_{dur}/\Delta T \rfloor$. Consequently, the number of columns in $\mathbf{X}$ is QK.

Under the first experimental condition, the $\mathbf{X}$ matrix can be constructed by converting an fMRI design sequence with Q stimulus types into Q sequences, where the q^{th} sequence indicates the time points where stimulus type q appears in the design sequence $\mathbf{d}$. For example, for the sequence $\mathbf{d} = \{122021\}$, $\delta_1 = (1, 0, 0, 0, 0, 1)$ and $\delta_2 = (0, 1, 1, 0, 1, 0)$. Hence, the first column of $\mathbf{X}_q$ is δ_q' and the subsequent columns

are obtained by shifting the previous column one position down and padding 0's on the $c-1$ rows, where c is the column number.

To illustrate, suppose the stimulus sequence is $\mathbf{d} = \{121020\}$. Including rest periods, there are six time points ($N = 6$) with two stimulus types ($Q = 2$). Let the interval time between two consecutive time points be four seconds ($\tau_{ISI} = 4$) and a total of six fMRI scans ($T = 6$) of a voxel is acquired with a four-second scanning interval ($\tau_{TR} = 4$). Assume that the duration of each HRF is sixteen seconds ($\tau_{dur} = 16$). Thus, we have $\Delta T = 4$, and $K = 1 + \lfloor 16/4 \rfloor = 5$. In this case, δ_1 and δ_2 are $(1, 0, 1, 0, 0, 0)$ and $(0, 1, 0, 0, 1, 0)$, respectively. The design matrix becomes $\mathbf{X} = [\mathbf{X}_1 \mathbf{X}_2]$ with

$$\mathbf{X}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \text{ and } \mathbf{X}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

As a result,

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

In interpreting the design matrix, a value of "1" in x_{11} indicates the 1^{st} height of the HRF of the 1^{st} stimulus type contributes to the 1^{st} fMRI signal. The "1"'s in x_{31},

x_{33}, x_{37} show that the 1^{st} height of the HRF of the 1^{st} stimulus type, the 3^{rd} height of the HRF of the 1^{st} stimulus type, and the 2^{nd} height of the HRF of the 2^{nd} stimulus type all contribute to the 3^{rd} accumulated HRF.

For cases where $\tau_{ISI} \neq \tau_{TR}$, construction of the design matrix is more complex. Here, we discuss the logical construction of the design matrix. We first construct an expanded design matrix based on an expanded design sequence, $\mathbf{d}^*$. The expanded design matrix is denoted by $\mathbf{X}^*$. The final design matrix $\mathbf{X}$ is extracted from $\mathbf{X}^*$ by taking some rows of the expanded matrix. The expanded design matrix has $N\tau_{ISI}/\Delta T$ rows, with KQ columns. $\tau^*_{ISI} = \Delta T$ for the expanded design matrix. For example, for an fMRI experiment with $\tau_{ISI} = 3$ and $\tau_{TR} = 2$, $\Delta T = 1$, therefore $\tau^*_{ISI} = 1$.

To construct the expanded design sequence, we add $(\tau_{ISI}/\Delta T) - 1$ resting stimuli between any two consecutive stimuli in the original sequence. For instance, if $\mathbf{d} = \{011010\}$ and $\tau_{ISI} = 3$ and $\tau_{TR} = 2$, $\Delta T = 1$, the expanded design sequence is formed by adding two resting stimuli $((3/1)-1)$ between any two consecutive stimuli in $\mathbf{d}$. As a result, the expanded design sequence becomes $\mathbf{d}^* = \{000100100000100000\}$. The length of the expanded design matrix is equal to the number of rows in the expanded design sequence, which is $N\tau_{ISI}/\Delta T = or18$ for this example.

The expanded matrix $\mathbf{X}^*$ is built as for a stimulus sequence $\mathbf{d}^*$ with τ^*_{ISI}, where the the first column of $\mathbf{X}^*$ is the same as $\mathbf{d'}^*$ (for $Q = 1$, $\mathbf{d}^* = \delta_{\mathbf{1}}$) and the subsequent columns of $\mathbf{X}^*$ are then obtained by shifting the previous column one position down and padding 0's at the $c - 1$ rows, where c is the column number.

To obtain the final design matrix, $\mathbf{X}$ is extracted from $\mathbf{X}^*$ by taking only row numbers $(j-1)\tau_{TR}/\Delta T + 1, j = 1, ..., T$. These rows correspond to the time points of fMRI scanning. Row $(j-1)\tau_{TR}/\Delta T + 1$ in the expanded matrix becomes Row j in the final design matrix.

For example, the expanded matrix $\mathbf{X}^*$ for the stimulus sequence $\mathbf{d} = \{011010\}$ becomes:

$$\mathbf{X}^* = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & ... \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & ... \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & ... \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & ... \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & ... \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & ... \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & ... \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & ... \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & ... \end{bmatrix}$$

From the expanded design matrix, we only obtain some rows for the final design matrix:

$$\mathbf{X} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & ... \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & ... \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & ... \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & ... \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & ... \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & ... \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & ... \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & ... \end{bmatrix}$$

In this section, we only considered the case with one stimulus type for the experimental condition $\tau_{ISI} \neq \tau_{TR}$. When $q \geq 2$, then δ_q vectors should be constructed from $\mathbf{d}$ and the same procedure applies in the construction of the individual design matrices. The expanded design matrix is constructed by concatenating the design matrices for each stimulus i.e., $\mathbf{X_q^*}, q = 1, ..., Q$ ($\mathbf{X}^* = [\mathbf{X_1^* X_2^*}, ..., \mathbf{X_Q^*}]$). The final design matrix is obtained as in the one-stimulus case. Equation (4.5) shows a more detailed and analytical construction of the design matrix.

$$\mathbf{X}_q = \left(\mathbf{I}_T \otimes \left[1, \mathbf{0}'_{m_{TR}-1}\right]\right) \mathbf{U}[\mathbf{B}^0, \mathbf{B}^1, ..., \mathbf{B}^{K-1}] \left(\mathbf{I}_K \otimes \boldsymbol{\delta}_q \otimes \begin{bmatrix} 1 \\ \mathbf{0}_{m_{ISI}-1} \end{bmatrix}\right). \quad (4.5)$$

Here, the N-by-1 vector $\boldsymbol{\delta}_q$ is a 0-1 vector whose i-th element is 1 if $d_i = q$. In addition, $m_{ISI} = (\tau_{ISI}/\Delta T)$, $m_{TR} = (\tau_{TR}/\Delta T)$, $\mathbf{0}_a$ is the a-by-1 vector of zeros, and

$$\mathbf{B} = \begin{bmatrix} \mathbf{0}'_{m_{ISI}N-1} & 0 \\ \mathbf{I}_{m_{ISI}N-1} & \mathbf{0}_{m_{ISI}N-1} \end{bmatrix},$$

$$\mathbf{U} = \begin{cases} [\mathbf{I}_{m_{TR}T}, \mathbf{O}_{m_{TR}T,(m_{ISI}N - m_{TR}T)}], & m_{TR}T < m_{ISI}N; \\ \begin{bmatrix} \mathbf{I}_{m_{ISI}N} \\ \mathbf{O}_{(m_{TR}T - m_{ISI}N),m_{ISI}N,} \end{bmatrix}, & m_{TR}T > m_{ISI}N; \\ \mathbf{I}_{m_{TR}T}, & m_{TR}T = m_{ISI}N; \end{cases}$$

and $\mathbf{O}_{a,b}$ is the a-by-b zero matrix.

4.4 Proposed Algorithms

The illustrative example in the previous section indicated that perturbing a single element in $\mathbf{d}$ alters multiple rows in the design matrix. This is in contrast to the conventional design problems for linear models, where changing one row does not necessarily affect the other rows. Hence, using conventional methods may not work for the fMRI case.

In this section, two novel greedy search algorithms are proposed for finding D-optimal fMRI designs for the estimation and detection problems. Updating formulas, also known as the delta functions, are derived to simplify the calculation of the new determinant value after changing one element in a design sequence.

4.4.1 The Estimation Problem

Based on the specific feature of the design matrix $\mathbf{X}$, a deterministic greedy search algorithm is proposed for finding D-optimal designs for the fMRI estimation problem. The idea is to sequentially perturb every element in the current design sequence, $\mathbf{d}$, by replacing each element with another label (0, 1, ..., or Q) for yielding the greatest improvement in the determinant of the information matrix. By changing the i-th element of $\mathbf{d}$ from q to q^*, the design matrix becomes $\mathbf{X}_{NEW} = \mathbf{X} + \mathbf{D}_i$, where

$\mathbf{D}_i = \mathbf{D}_{i,q^*} - \mathbf{D}_{i,q}$ for some T-by-QK matrices $\mathbf{D}_{i,q}$ and $\mathbf{D}_{i,q^*}$. Here, $\mathbf{D}_{i,0} = \mathbf{O}_{T\times QK}$, a matrix of zeros, for $q = 0$. For $q = 1, ..., Q$, $\mathbf{D}_{i,q} = [\mathbf{D}_{i,q,1} \cdots \mathbf{D}_{i,q,Q}]$ with $\mathbf{D}_{i,q,j} = \mathbf{O}_{T\times K}$ for all $j \neq q$, and $\mathbf{D}_{i,q,q}$ being a 0-1 matrix. The positions of 1's in $\mathbf{D}_{i,q,q}$ correspond to the altered element in the design sequence. Let (r_i, c_i) be the position where the first 1 appears in $\mathbf{D}_{i,q,q}$. We have

$$r_i = \left\lceil \frac{(i-1)\tau_{ISI}}{\tau_{TR}} \right\rceil + 1; \; c_i = \frac{(r_i - 1)\tau_{TR} - (i-1)\tau_{ISI}}{\Delta T} + 1,$$

where $\lceil . \rceil$ denotes a ceiling operation. In particular, the r_i-th scan is the first fMRI scan after the presentation of d_i. The c_i-th discretized HRF value, h_{q,c_i}, which is evaluated at $(c_i - 1)\Delta T$ seconds after the stimulus onset, contributes to y_{r_i}. Following the construction rule of the fMRI design matrix, it can also be shown that other positions of 1's in $\mathbf{D}_{i,q,q}$ are $(r_i + \ell, c_i + m_{TR}\ell)$, with $\ell = 1, ..., \lfloor (K - c_i)/m_{TR} \rfloor$. The matrix $\mathbf{D}_{i,q^*,q^*}$ for the q^*-th stimulus type can also be constructed using this method.

Let the i-th element $d_i = q$ in the current design be replaced by q^*, then the determinant of the new information matrix is given by $|\mathbf{X}'_{NEW}\mathbf{E}\mathbf{X}_{NEW}| = |\mathbf{X}'\mathbf{E}\mathbf{X}|\Delta_E(\mathbf{D}_i)$, where $\mathbf{E} = \mathbf{V}'(\mathbf{I}_T - \mathbf{P}_{\mathbf{VS}})\mathbf{V}$, and

$$\Delta_E(\mathbf{D}_i) = |\mathbf{I}_{QK} + \mathbf{M}_h^{-1}(\mathbf{X}'\mathbf{E}\mathbf{D}_i + \mathbf{D}_i'\mathbf{E}\mathbf{X} + \mathbf{D}_i'\mathbf{E}\mathbf{D}_i)| \tag{4.6}$$

is the delta function that depends on q^* through $\mathbf{D}_i$. Thus, the goal is to find $q^* \in \{0, 1, ..., Q\}\backslash\{q\}$ that maximizes the delta function (see also Lemma 2.5.1 in Fedorov (1971)). Since $\mathbf{D}_i$ is sparse, the matrix operations in Equation (4.6) can be calculated efficiently. For example, when $\tau_{ISI} = \tau_{TR}$, $\mathbf{D}_i$ is a T-by-QK matrix with

$$\mathbf{D}_i = [\mathbf{O}_{T\times K}, ..., -\mathbf{I}^*_{T\times K}, ..., \mathbf{I}^*_{T\times K}, ..., \mathbf{O}_{T\times K}];$$

$$\mathbf{I}^*_{T\times K} = \left[\begin{array}{c|c} \mathbf{O}_{i-1,\ell} & \mathbf{O}_{i-1,K-\ell} \\ \hline \mathbf{I}_\ell & \mathbf{O}_{\ell,K-\ell} \\ \hline \mathbf{O}_{T-i-\ell+1,\ell} & \mathbf{O}_{T-i-\ell+1,K-\ell} \end{array}\right].$$

Here, $\mathbf{O}_{a\times b}$ is the a-by-b matrix of zeros, and $\mathbf{I}_\ell$ is the identity matrix of order $\ell = min\{K, (T-i+1)\}$. The matrix $-\mathbf{I}^*_{T\times K}$ forms the $((q-1)K+1)$-th to qK-th columns of $\mathbf{D}_i$, whereas $\mathbf{I}^*_{T\times K}$ forms the $((q^*-1)K+1)$-th to q^*K-th columns. Corresponding to $\mathbf{D}_i$, $\mathbf{E}$ can then be partitioned as

$$\mathbf{E} = \left[\begin{array}{c|c|c} \mathbf{E}_{11} & \mathbf{E}_{12} & \mathbf{E}_{13} \\ \hline \mathbf{E}'_{12} & \mathbf{E}_{22} & \mathbf{E}_{23} \\ \hline \mathbf{E}'_{13} & \mathbf{E}'_{23} & \mathbf{E}_{33} \end{array}\right],$$

where $\mathbf{E}_{11}$, $\mathbf{E}_{22}$, and $\mathbf{E}_{33}$ are symmetric matrices of orders $(i-1)$, ℓ, and $T-i-\ell+1$, respectively. Consequently, $\mathbf{D}'_i\mathbf{E}\mathbf{D}_i$ is reduced to the following form:

$$\mathbf{D}'_i\mathbf{E}\mathbf{D}_i = \left[\begin{array}{c|c|c|c|c} \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{E}_{22} & \mathbf{O} & -\mathbf{E}_{22} & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \hline \mathbf{O} & -\mathbf{E}_{22} & \mathbf{O} & \mathbf{E}_{22} & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} \end{array}\right].$$

Here, $\mathbf{O}$ is a matrix of zeros. The positions of $\mathbf{E}_{22}$ and $-\mathbf{E}_{22}$ depend on q and q^*, and can be easily obtained. We note that when either q or q^* is 0, the matrix $\mathbf{D}_i$, and hence $\mathbf{D}'_i\mathbf{E}\mathbf{D}_i$, may even be more sparse. Similarly, we may partition the product of $\mathbf{E}$ and $\mathbf{X}$ into

$$\mathbf{EX} = \begin{bmatrix} \mathbf{EX}_1 \\ \mathbf{EX}_2 \\ \mathbf{EX}_3 \end{bmatrix},$$

where $\mathbf{EX}_1$, $\mathbf{EX}_2$, and $\mathbf{EX}_3$ form, respectively, the first $(i-1)$ rows, the i-th to the $(i+\ell-1)$-th rows, and the last $(T-i-\ell+1)$ rows of $\mathbf{EX}$. We then have

$$\mathbf{D}_i'\mathbf{EX} = \begin{bmatrix} \mathbf{O} \\ -\mathbf{EX}_2 \\ \mathbf{O} \\ \mathbf{EX}_2 \\ \mathbf{O} \end{bmatrix},$$

where $-\mathbf{EX}_2$ starts from the $((q-1)K+1)$-th row and $\mathbf{EX}_2$ starts from the $((q^*-1)K+1)$-th row of $\mathbf{D}_i'\mathbf{EX}$.

The algorithm begins with a randomly generated sequence $\mathbf{d}$ and its corresponding design matrix $\mathbf{X}$. Next, it sequentially perturbs every element of $\mathbf{d}$ for improving the value of the D-criterion. Specifically, each run of the algorithm has N iterations. At every i-th iteration of a run, $d_i = q$ is replaced by the $q^* \in \{0, 1, ..., Q\} \backslash \{q\}$ that yields the greatest improvement in the D-value. The delta function $\Delta_E(\mathbf{D}_i)$ and the previously derived formulas allow a fast calculation of the updated D-value. The process is repeated several times until no further improvement is achieved. The pseudo-code of this algorithm is found in Algorithm 1. In the next section, the algorithm is modified to obtain D-optimal designs for the detection problem.

4.4.2 The Detection Problem

For the detection problem, the focus is on obtaining an fMRI design that facilitates precise inferences on the magnitudes of the HRFs, specifically $\boldsymbol{\theta}$ in model (2.13). To

this end, we apply the same idea of sequentially perturbing each d_i in the design sequence $\mathbf{d}$ as described in the previous section. Let $\mathbf{Z} = \mathbf{X}(\mathbf{I}_Q \otimes \mathbf{h}^*)$. The information matrix of $\boldsymbol{\theta}$ becomes $\mathbf{Z}'\mathbf{E}\mathbf{Z}$. Similar to the estimation problem, it can be shown that changing one element of design sequence from q to q^* changes the information matrix to:

$$\mathbf{Z}'_{NEW}\mathbf{E}\mathbf{Z}_{NEW} = \mathbf{Z}'\mathbf{E}\mathbf{Z} + \mathbf{Z}'\mathbf{T}_i + \mathbf{T}'_i\mathbf{Z} + \mathbf{T}'_i\mathbf{T}_i,$$

where $\mathbf{Z}_{NEW} = (\mathbf{X} + \mathbf{D}_i)(\mathbf{I}_Q \otimes \mathbf{h}^*)$, $\mathbf{T}_i = \mathbf{D}_i(\mathbf{I}_Q \otimes \mathbf{h}^*)$, and $\mathbf{D}_i$ is defined as in the previous section. The delta function for updating the determinant of $\mathbf{Z}'_{NEW}\mathbf{E}\mathbf{Z}_{NEW}$ becomes:

$$\Delta_D(\mathbf{T}_i) = |\mathbf{I}_Q + \mathbf{M}_\theta^{-1}(\mathbf{Z}'\mathbf{E}\mathbf{T}_i + \mathbf{T}'_i\mathbf{E}\mathbf{Z} + \mathbf{T}'_i\mathbf{E}\mathbf{T}_i)|, \tag{4.7}$$

where $\mathbf{T}_i = [\mathbf{t}_{i,1}, ..., \mathbf{t}_{i,Q}]$, $\mathbf{t}_{i,j}] = \mathbf{O}_T$ is the vector of T zeros for $j \neq q$ and $j \neq q^*$, $\mathbf{t}_{i,q^*} = -\mathbf{t}_{i,q} = \mathbf{D}_{i,q,q}\mathbf{h}^*$ when both q and q^* are positive. Only part of $\mathbf{D}_{i,q,q}\mathbf{h}^*$ is non-zero. In particular,

$$\mathbf{D}_{i,q,q}\mathbf{h}^* = \begin{cases} [\mathbf{0}_{1,r_i-1}, h^*_{c_i}, h^*_{c_i+m_{TR}}, ..., h^*_{c_i+m_{TR}\lfloor(K-c_i)/m_{TR}\rfloor}, \mathbf{0}_{1,K-(c_i+m_{TR}\lfloor(K-c_i)/m_{TR}\rfloor)}], \\ \quad for\ c_i + m_{TR}\lfloor(K-c_i)/m_{TR}\rfloor \leq K-1; \\ [\mathbf{0}_{1,r_i-1}, h^*_{c_i}, h^*_{c_i+m_{TR}}, ..., h^*_{c_i+m_{TR}\lfloor(K-c_i)/m_{TR}\rfloor}], \\ \quad for\ c_i + m_{TR}\lfloor(K-c_i)/m_{TR}\rfloor > K-1. \end{cases}$$

Like $\mathbf{D}_i$, $\mathbf{T}_i$ is also a sparse matrix, allowing the adaptation of the previous algorithm to this problem (see Algorithm 1). The delta function $\Delta_D(\mathbf{T}_i)$ and the sparsity of $\mathbf{T}_i$ facilitates the efficient calculation of the objective functions during the search.

4.4.3 Multi-Objective Optimization

In some cases, fMRI studies simultaneously need to satisfy both objectives of estimation and detection. The optimal design for such cases can be obtained by formulating and solving a multi-objective optimization (MOO) problem. Following previous studies (see, e.g., Wager and Nichols (2003) and Kao *et al.* (2009)), we consider a set of weighted-sum criteria, each being a weighted sum of the standardized optimality criteria for detection and estimation. The criterion has the following form:

$$f(\lambda) = \lambda \left(\frac{|\mathbf{X}'\mathbf{E}\mathbf{X}|}{\max |\mathbf{X}'\mathbf{E}\mathbf{X}|} \right)^{\frac{1}{KQ}} + (1-\lambda) \left(\frac{|\mathbf{Z}'\mathbf{E}\mathbf{Z}|}{\max |\mathbf{Z}'\mathbf{E}\mathbf{Z}|} \right)^{\frac{1}{Q}}, \tag{4.8}$$

where $\lambda \in [0, 1]$. The maxima in Equation (4.8) are taken over all possible fMRI designs, which are obtained by using the proposed algorithm for maximizing $|\mathbf{X}'\mathbf{E}\mathbf{X}|$ and $|\mathbf{Z}'\mathbf{E}\mathbf{Z}|$. For a given value of λ, the elements in a design sequence are sequentially changed so as to maximize:

$$\lambda \left(\frac{|\mathbf{X}'\mathbf{E}\mathbf{X}|_{(i)} \Delta_E(\mathbf{D}_i)}{\max |\mathbf{X}'\mathbf{E}\mathbf{X}|} \right)^{\frac{1}{KQ}} + (1-\lambda) \left(\frac{|\mathbf{Z}'\mathbf{E}\mathbf{Z}|_{(i)} \Delta_D(\mathbf{T}_i)}{\max |\mathbf{Z}'\mathbf{E}\mathbf{Z}|} \right)^{\frac{1}{Q}}. \tag{4.9}$$

Here, $|\mathbf{X}'\mathbf{E}\mathbf{X}|_{(i)}$ and $|\mathbf{Z}'\mathbf{E}\mathbf{Z}|_{(i)}$ are the determinants of the information matrix for estimation and detection before perturbing the value of d_i. The delta functions, $\Delta_E(\mathbf{D}_i)$ and $\Delta_D(\mathbf{T}_i)$, are the improvements in the D-optimality criterion value for estimation and detection after changing the value of d_i in design sequence. Equation (4.9) enables a recursive update of the objective function's value after a change in the design.

Data: $\tau_{ISI}, \tau_{TR}, N, Q, R$

Result: Optimal design for the estimation problem

Initialization;

for $r = 1...R$ **do**

  Generate a random fMRI sequence $\mathbf{d}$;

  Construct the design matrix $\mathbf{X}$;

  $z = 1$;

  $Det_z = |\mathbf{X}'\mathbf{E}\mathbf{X}|$; $Det^{old} = 0$;

  while $z = 1$ *or* $Det_z - Det^{old} \geq \epsilon$ **do**

    $Det^{old} = Det_z$;

    for $i = 1, 2, ..., N$ **do**

      for $q = 0, 1, ..., Q$, *except* $j = d_i$ **do**

        Calculate $\Delta_E(\mathbf{D}_i)$ ($\Delta_D(\mathbf{T}_i)$), as defined by Equation (4.6) (Equation (4.7)), for the estimation (detection) problem;

      end

      Find the stimulus type q^* such that $q^* = \arg\max \Delta_E(\mathbf{D}_i)$ ($q^* = \arg\max \Delta_D(\mathbf{T}_i)$) for the estimation (detection) problem;

      if $\max \Delta_E(\mathbf{D}_i) > 1$ ($\max \Delta_D(\mathbf{T}_i) > 1$) *for the estimation (detection) problem* **then**

        $d_i = q^*$;

        $\mathbf{X} = \mathbf{X} + \mathbf{D}_i$ ($\mathbf{Z} = \mathbf{Z} + \mathbf{T}_i$) for the estimation (detection) problem;

        $Det_z = Det_z \times \Delta_E(\mathbf{D}_i)$ ($Det_z = Det_z \times \Delta_D(\mathbf{T}_i)$) for the estimation (detection) problem;

      end

    end

    $Det_{z+1} = Det_z$;

    $z = z + 1$;

  end

  $\mathbf{X}^{(r)} = \mathbf{X}$ ($\mathbf{Z}^{(r)} = \mathbf{Z}$) for the estimation (detection) problem;

  $Det^{(r)} = Det_z$;

end

Output the optimal design $\mathbf{X}^* = \mathbf{X}^{(r)}$ such that $r = \arg\max Det^{(r)}$;

Algorithm 1: Finding Optimal Designs for the fMRI Estimation (Detection) Problem

4.5 Comparing CEA and GA for fMRI Experimental Design Problems

The proposed CEA-inspired algorithm developed in the previous section for the fMRI problem is compared with the modified version of the GA for fMRI developed by Kao *et al.* (2009).

4.5.1 Performance of the Estimation Algorithm in fMRI Experiments

To explore the performance of the proposed algorithm for the estimation problem, the following two scenarios are considered: (1) $(\tau_{ISI}, \tau_{TR}) = (2, 2)$, and (2)

$(\tau_{ISI}, \tau_{TR}) = (3, 2)$. For both scenarios, the duration of the HRF is set to $\tau_{dur} = 32$ seconds. Therefore, $K = 1 + \lfloor 32/2 \rfloor = 17$ and $K = 1 + \lfloor 32/1 \rfloor = 33$ for the first and second scenarios, respectively. Following Kao *et al.* (2009), a total of eight combinations of Q and N are considered namely, (Q, N) =(2, 252), (3, 255), (4, 624), (6, 342), (7, 624), (8, 728), (10, 1330), and (12, 2196).

Table 4.1 compares the determinants and relative efficiency of the optimal designs found by the our algorithm and GA for the first scenario. To be comparable, each algorithm is repeated 10 times with 10 different initial designs. The design sequence with the maximum determinant from the 10 runs is chosen as the optimal design. In the literature on GA, initial designs are randomly generated at each implementation, while in other cases, traditional fMRI designs are chosen as the initial design (see Kao *et al.* (2009) for details). In our implementation of the proposed algorithm, we randomly generated an initial design for 9 runs, while the starting design for the 10^{th} run is a traditional, fMRI design.

Table 4.1: Comparing the Determinants of the Designs Obtained by Genetic Algorithm and Coordinate Exchange Algorithm for the Estimation Problem with $\tau_{ISI} = 2$ and $\tau_{TR} = 2$

		Determinant			Time (minutes)	
Q	N	GA	New algorithm	Relative efficiency	GA	New algorithm
2	255	$2.84(\times 10^{57})$	2.94	1.03	12.2	0.3
3	255	$3.84(\times 10^{81})$	3.96	1.03	21.9	0.5
4	624	$4.64(\times 10^{130})$	5.24	1.12	21.7	5.0
6	342	$1.41(\times 10^{157})$	1.41	1.00	56.0	4.3
7	624	$1.87(\times 10^{209})$	2.42	1.29	130.8	24.0
8*	728	$1.33(\times 10^{243})$	1.33	1.00	19.7	7.2
10*	1330	$3.12(\times 10^{336})$	3.13	1.00	81.8	50.0
12*	2196	$6.95(\times 10^{435})$	6.96	1.00	173.7	181.1

* Indicates cases where the GA did not improve on its initial designs.

In all cases, optimal designs from the new algorithm are comparable or better than designs from the GA approach with respect to D-efficiency. Table 4.1 also shows the computing times of each approach. In all cases but one, the new algorithm has consistently faster implementations than GA. We note that the GA terminates prematurely when it could not find a design that outperforms the initial design, hence in some cases, the computational times are seemingly short. For example, in the case where $Q = 4, N = 624$, the proposed algorithm is only 3.34 times faster than GA, compared to the case when $Q = 2, N = 255$ where the improvement factor is close to 40. In the former case, however, the relative efficiency is 1.12, and it is obvious that the GA could not find a better design so it likely terminated early.

For the case where $\tau_{ISI} = 3$ and $\tau_{TR} = 2$, $K = 33$, so the size of the information matrix is approximately twice of that in the previous scenario. The advantage of the proposed algorithm over the GA is unequivocal for these cases as shown in Table 4.2. Based on these results, we can see that the new algorithm is more effective in finding optimal fMRI designs, particularly with more stimulus types and when fMRI scanning times are out-of-sync with the onset times of the stimuli.

Table 4.2: Comparing the Determinants of the Designs Obtained by Genetic Algorithm and Coordinate Exchange Algorithm for the Estimation Problem with $\tau_{ISI} = 3$ and $\tau_{TR} = 2$

		Determinant			Time (minutes)	
Q	N	GA	New algorithm	Relative efficiency	GA	New algorithm
2	255	$5.76(\times 10^{94})$	6.41	1.11	13.9	0.7
3	255	$6.12(\times 10^{131})$	7.24	1.18	60.9	2.6
4	624	$2.36(\times 10^{217})$	4.19	1.77	435	29.1
6	342	$0.442(\times 10^{246})$	1.05	1.12	315	28.8
7	624	$0.0373(\times 10^{340})$	1.8	48.25	1324	140.8
8	728	$0.363(\times 10^{394})$	6.6	18.1	1578	214.6
10*	1330	$3.25(\times 10^{553})$	1.23×10^4	3784	3727.2	233.5
12*	2196	$6.06(\times 10^{726})$	2.73×10^4	4504	663.22	591.18

* Indicates cases where the GA did not improve on its initial designs.

4.5.2 *Performance of the Detection Algorithm in fMRI Experiments*

Similar to what was done for the estimation problem, we compare the performance of GA and our new algorithm for the detection problem when $\tau_{ISI} = 2, \tau_{TR} = 2$ and $\tau_{ISI} = 3, \tau_{TR} = 2$. For each of these cases, the number of columns for each stimulus is respectively 17 and 33, and the number of fMRI events varies between 200+ to 2000+. Results are shown in Tables 4.3 and 4.4.

Table 4.3: Comparing the Determinants of the Designs Obtained by Genetic Algorithm and Coordinate Exchange Algorithm for the Detection Problem with $\tau_{ISI} = 2$ and $\tau_{TR} = 2$

		Determinant			Time (minutes)	
Q	N	GA	New algorithm	Relative efficiency	GA	New algorithm
2	255	$2.46(\times 10^4)$	2.46	1	6.7	0.07
3	255	$2.052(\times 10^6)$	2.054	1	12.2	0.08
4	624	$4.66(\times 10^9)$	4.72	1.01	17.6	0.2
6	342	$1.72(\times 10^{12})$	1.73	1.005	12.4	0.52
7	624	$5.5(\times 10^{15})$	5.89	1.07	5.5*	1.2
8	728	$1.61(\times 10^{18})$	1.69	1.04	6.5*	2.11
10	1330	$4.87(\times 10^{24})$	4.9	1.006	18.7*	2.28
12	2196	$2.37(\times 10^{31})$	2.75	1.16	47.9*	38.3

The GA approach was unable to improve on the initial fMRI traditional designs in cases with 7, 8, 10 and 12 stimulus types. The optimal designs for detection constructed using the proposed algorithm are superior across the board compared to the designs from the GA approach. The greatest improvements are realized in problems with more stimulus types and bigger information matrices.

4.6 Conclusion

fMRI studies are useful for mapping brain activity and determining the effects of various stimuli. fMRI experiments are costly and require the participation of human

Table 4.4: Comparing the Determinants of the Designs Obtained by Genetic Algorithm and Coordinate Exchange Algorithm for the Detection Problem with $\tau_{ISI} = 2$ and $\tau_{TR} = 2$

		Determinant			Time (minutes)	
Q	N	GA	New algorithm	Relative efficiency	GA	New algorithm
2	255	$1.161(\times 10^4)$	1.174	1.01	7.5	0.25
3	255	$6.4(\times 10^5)$	6.32	0.98	25	0.8
4	624	$0.997(\times 10^9)$	1.01	1.01	13.3	1.1
6	342	$1.65(\times 10^{11})$	1.69	1.02	142.1	3.23
7	624	$3.95(\times 10^{14})$	4.11	1.04	17.1*	0.8
8	728	$7.8(\times 10^{16})$	7.93	1.01	16.2*	1.3
10	1330	$1.01(\times 10^{23})$	1.02	1.01	87.9*	6
12	2196	$2.52(\times 10^{29})$	2.94	1.16	516.5*	31.95

beings as test subjects, so it is of paramount importance that these experiments are designed as efficiently and effectively as possible. In this chapter, a new algorithm was proposed for constructing D-optimal designs for the estimation and detection problems in fMRI studies. The two problems have separate goals in statistical inference and yield different linear models for estimation. The main goal of the optimal design problem is to find a sequence of stimuli, $\mathbf{d}^*$ that optimizes a functional of the design matrix. This is not a trivial problem, because each stimulus has a presumed "carryover" or lasting effect. The carryover effect further complicates the structure of the design matrix.

The methods proposed in this chapter adapted the linear models proposed for analyzing fMRI data to cases where there are several experimental subjects. The primary contributions include the derivation of update formulas similar to Fedorov's delta function in point-exchange and coordinate exchange, as well as a complete algorithm for generating the optimal designs. The algorithms resemble the coordinate exchange algorithm, where each element of the design matrix is perturbed until an

optimal design is achieved. By exploiting the sparseness of the design matrix, update formulas are calculated faster hence resulting in faster overall computation times.

The performance of the new algorithm is compared with the only existing approach published in the literature namely, the Genetic Algorithm applied to fMRI studies in Kao *et al.* (2009). In almost of the cases considered, the new algorithm showed improvements in the D-criterion, as well as improvements in computing times. The disparity between the two algorithms becomes more evident as the problem becomes more complex, especially in cases with more stimuli and when the presentation times of stimuli and MR scanning times are out-of-sync.

A comprehensive comparison of the two algorithms are shown in Figures 4.3 and 4.4. These two graphs demonstrate the superior performance of the new algorithm over the GA in finding more statistically efficient designs with less CPU time. More notably, the GA fails to make improvements on its initial designs for larger problems. In contrast, the new algorithm consistently improves on the initial design to achieve optimal experiments that are highly efficient for detecting brain activation.

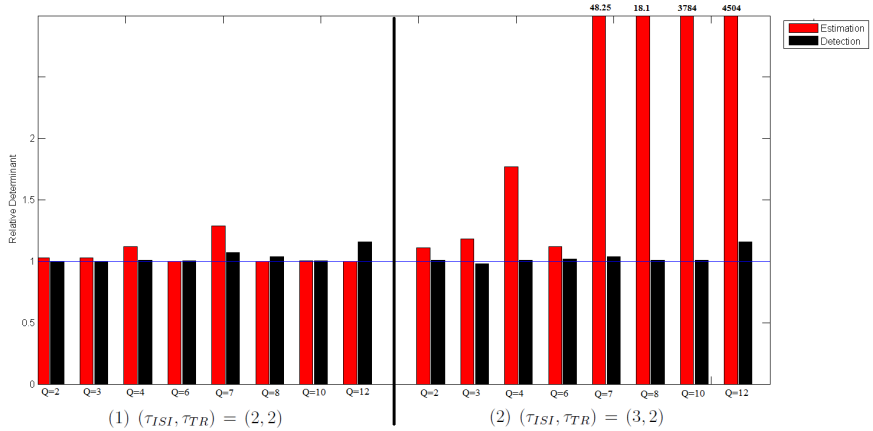


Figure 4.3: Relative Determinant of the Optimal Designs Calculated by the New Algorithm over GA's Optimal Designs for Design Sequences with (Q, N) with $(2, 255)$, $(3, 255)$, $(4, 624)$, $(6, 342)$, $(7, 624)$, $(8, 728)$, $(10, 1330)$ and $(12, 2196)$.

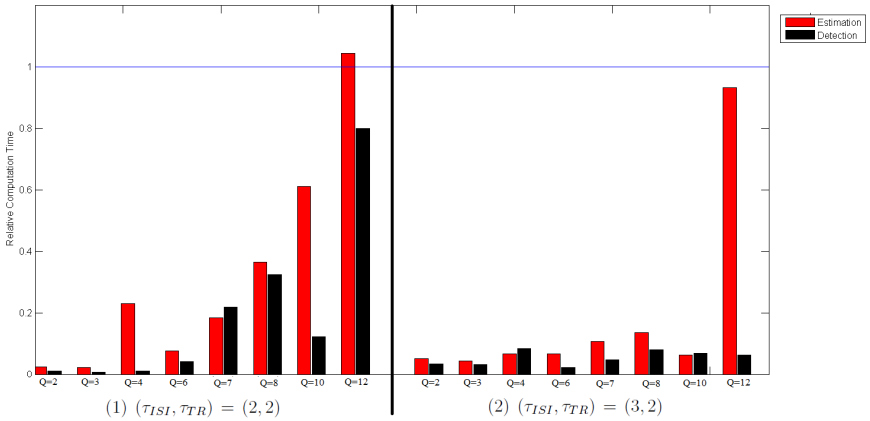


Figure 4.4: Relative Computation Time of the New Algorithm over GA's Computation Time for Design Sequences with (Q, N) with $(2, 255)$, $(3, 255)$, $(4, 624)$, $(6, 342)$, $(7, 624)$, $(8, 728)$, $(10, 1330)$ and $(12, 2196)$.

Chapter 5

OPTIMAL EXPERIMENTAL DESIGNS FOR DYNAMIC RESPONSES

Notational Conventions

$\mathbf{Y}$	**Response Vector**
N	**Number of Experiments**
$\mathbf{B(x)}$	**Basis Matrix**
$\mathbf{t}$	**Vector of Sampling Time**
$\boldsymbol{\theta}$	**Vector of Unknown Model Parameters for Smoothing**
$\mathbf{W}$	**Variance-Covariance Matrix of Observations**
$\mathbf{m}$	**Order of B-spline Bases**
L	**Number of Interior Knots for B-spline system**
p	**Number of B-spline system bases**
$\boldsymbol{\Sigma}$	**Covariance Matrix of Mixed Effects Model**
$\boldsymbol{\beta}$	**Vector of Unknown Model Parameters in the Mixed Effects Model**
$\boldsymbol{\omega}$	**Vector of Random Effects in the Mixed Effects Model**
$\mathbf{X}$	**Design Matrix**
$\mathbf{F(x)}$	**Design Matrix for Independent Factors**
$\mathbf{H}$	**Matrix of the Unknown Model Parameters in Second Stage of the Hierarchical Modeling**
$\mathbf{f(x_i)}$	**Column Vector That Contains the Values of i-th Experimental Factors**

Similar to fMRI studies, designing experiments for functional data in other engineering and scientific applications require precision in estimation and economy in design. Experiments typically involve multiple independent variables or covariates that need to be optimized. For example, Binde *et al.* (2012) studied soot and NO_x emissions of diesel engines through the design of a spatially separated pilot injection. This experiment involved three independent variables namely, the position of the pilot injector, the start of injection (SOI), and the number of pilot injections. Soot temperature and concentration (mass) are recorded over the entire engine cycle using

pyrometrics. It is immediately evident that soot temperature and concentration are dynamic over the engine cycle, so this can be modeled as functional data.

Experimenters will raise two practical questions regarding this type of experimental design namely, (1) at what settings should these factors be held at different runs of the experiment and (2) at what points in the continuum should the response be measured? In many cases, the sampling points need to be determined, not only for modeling precision, but also from the standpoint of economy. Measurement equipment that generate experimental data may not be automated, so manpower is required to collect data points. This is the primary difference between static-response and functional response experiments – when collecting functional data over a spectrum variable, it is necessary to plan for the sampling or measurement points.

This chapter addresses the topic of designing experiments for systems with dynamic responses. Methods in this chapter focus on two cases, namely, (1) derivation of the optimal sampling points of spectrum variable and (2) determining the optimal experimental settings and sampling points simultaneously.

This chapter consists of two contributions to the literature. First, we propose an algorithm for finding D-optimal sampling times and second, the method for constructing designs when both sampling times and factor settings need to be planned for an experiment is proposed. In the next two sections, we provide a reviews of similar work on optimal designs (Section 5.1) and non-parametric models for functional data (Section 5.2). Section 5.3 deals with the case when only the sampling points need to be determined for the experiment. Robust designs are also explored when there are uncertainties in the functional form of the data. Section 5.4 considers cases when both sampling times and experimental factor settings need to be determined simultaneously. Finally, Section 5.5 compares a design generated by the proposed algorithm against an ordinary D-optimal design commonly used in literature.

5.1 Introduction

The dynamic response systems studied in this chapter are functions of time or an observable variable that is referred as spectrum variable. Instead of a single response value, these systems generate response curves over the spectrum variable. The dynamic characteristics of these curves and their interactions with other experimental factors are of interest to experimenters, thus it requires new experimental design methods for exploring such system efficiently.

Experiments that generate dynamic responses are found in, e.g., Crowder and Hand (1990), Fan and Zhang (2000), Nair *et al.* (2002), Woods *et al.* (2003), and Kao *et al.* (2009).

Statistical models for modeling dynamic responses will be discussed in a later section. For an experimenter, designing these experiments requires her to make decisions on (1) when or where to take response measurements along the spectrum variable and (2) how to choose the combination of experimental factors and their levels. The first consideration is unique for such experiments, especially when the measurement cost is high. A good experimental design should demand less experimental runs and/or less measurement frequency, but it can achieve higher statistical efficiency in model parameter estimation or response prediction. In this chapter we will represent the methods for deriving the optimal sampling times of response variable and the optimal settings of experimental factors based on the D-optimal criterion.

In contrast with standard experimental designs, which require standard cubic or spherical design regions and normal/linear models (see Montgomery (2008), Wu and Hamada (2011b) for more details on the standard designs). Designing experiments with irregular design regions and non-normally distributed responses was founded on Kiefer (1959, 1961); Kiefer and Wolfowitz (1959) via introducing alphabetic criteria

for constructing optimal experimental designs. Among these optimality criteria, D-optimality is the most popular one for evaluating the quality of a design for model parameter estimation. The D-optimal criterion maximizes the determinant of expected Fisher information matrix of parameter estimators. Fedorov (1969, 1971) introduced the point exchange algorithm (PEA) for constructing exact D-optimal designs for linear models. This algorithm and its variants are widely adopted by existing statistical software (see, e.g., Mitchell and Miller Jr (1970), Wynn (1970), Mitchell (1974), Cook and Nachtrheim (1980), Atkinson and Donev (1989), Welch (1984) , Nguyen and Miller (1992), Vahl and Milliken (2011), and Nguyen (1993)). The PEAs proposed by these researchers take the exhaustive search approach to finding the optimal design from a large set of candidate design points. Generating and storing the candidate matrix and comparing each candidate design point with others impose a huge computational burden in many optimal design problems. Therefore, some meta-heuristic optimization algorithms, such as genetic algorithm (GA) and simulated annealing (GA) algorithm, have been adapted to obtain optimal experimental designs (see, e.g., Haines (1987), Meyer and Nachtsheim (1988), Broudiscou *et al.* (1996), and Heredia-Langner *et al.* (2003)). On the other hand, the coordinate exchange algorithm (CEA) proposed by Meyer and Nachtsheim (1995) has been able to address the PEA's shortcoming by avoiding the explicit list of candidate design points. Until today, this type of algorithm is still one of the most popular algorithms for constructing D-optimal designs for linear and nonlinear models. Despite the popularity of the alphabetic optimal designs, they are usually planned for physical experiments. Some recent development of experimental designs for computer experiments, where computer systems could be deterministic, can be found in, e.g., Johnson *et al.* (1990), Santner *et al.* (2013) and Fang *et al.* (2005). Joseph *et al.* (2015) recently proposed a new criterion to calculate maximum projection designs in

computer experiments. This design simultaneously considers the space-filling property and the projection power of a design. Morris (2015) proposed a criterion that resembles I-optimality to support experiments on stochastic computer models, where model uncertainty and experimental information insufficiency exist.

While the optimal design of experiments with static responses has been widely discussed in literature, the research on experimental designs for dynamic responses has been sparse. To design the experiment with a dynamic response, one needs to select a set of response measurement points on the spectrum variable, as well as the setting of other experimental factors. These two aspects may be considered separately or jointly. Most of the existing literature on experimental designs for dynamic responses focused on the first aspect only, i.e., the response measurement locations or the sampling times on the response curve. Gaffke *et al.* (1999) used B-spline bases for modeling response curves and then found the D-optimal design for dynamic response. Woods *et al.* (2003) considered an additional interaction term of B-spline bases and ordinary polynomial models. Heiligers (1998) used Chebyshev splines for designing E-optimal experiments with dynamic responses. Finding the D_s and T optimal sampling times for functional data was also discussed by Fisher and Woods Fisher (2012). Our proposed algorithm extends these approaches by considering both experimental settings and sampling times simultaneously via an mixed effects modeling method of dynamic response. This method is derived from the hierarchical modeling approach to fitting functional data recommended by Del Castillo *et al.* (2012), Wu and Hamada (2011b), Tsui (1999) and Nair *et al.* (2002). We also note that mixed effects models are widely studied in longitudinal data analysis (see, e.g., Laird and Ware (1982), Verbeke and Molenberghs (2009b)).

The remainder of the chapter is organized as follows. The B-spline, as a flexible nonparametric model for fitting dynamic response data, will be discussed in the next

section. A novel algorithm to be presented in Section 3 will focus on the planning of sampling times on response curves. Another algorithm will be provided in Section 4 for simultaneously producing optimal experimental settings and optimal sampling times. Finally, the performance of optimal designs obtained by our algorithms will be compared with other designs through several examples.

5.2 Regression Models for Dynamic Responses

It is common to use polynomial regression models to depict nonlinear relationships between a response variable and a predictor variable, but these models are inadequate at capturing complicated local behaviors of a dynamic response. Figure 5.1 plots such a data set where the response variable demonstrates different dynamic behaviors in different region. The type of data is referred as whiplike structured data in Ruppert *et al.* (2003). A direct remedy of polynomial regression is to construct many local polynomial functions based on the data, or piecewise polynomial regression. Each local polynomial function is able to model response dynamics within its corresponding region, while the overall smoothness of the response curve is obtained by imposing connectivity and smoothness constraints at the end points (knots) shared by the two adjacent local polynomials. Cubic spline regression model is a popular piecewise polynomial model, in which the second-order derivatives of adjacent local polynomial functions can be set equal, thus the connection is smooth to visual inspection. In Figure 5.1, the cubic polynomial function, quartic polynomial function, and cubic spline function with equally spaced knots are fitted to the data and it demonstrates the inadequacy of polynomial models in comparison with piecewise models.

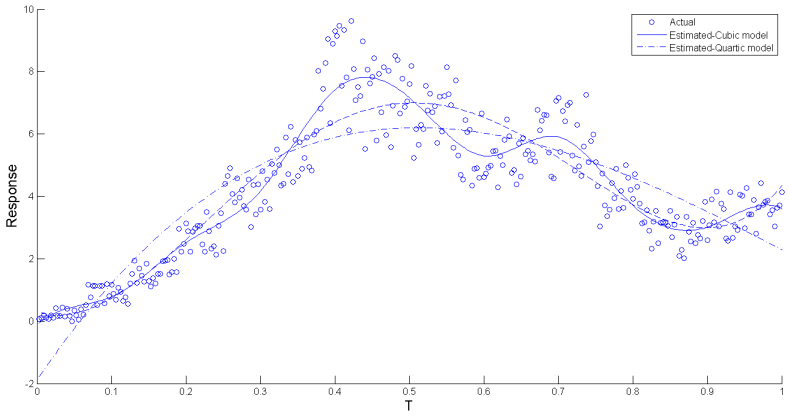


Figure 5.1: Whiplike Structured Data. Complexity of the Data Changes Significantly at $t = 0.4$.

5.2.1 B-Spline Model

Piecewise polynomial regression models constitute a flexible method for the nonparameteric curve fitting of multiple local features. In the piecewise regression model, dynamic responses are segmented to pieces by knots and each piece is fitted by a polynomial function. Boundary constraints of these polynomial functions are implemented to guarantee the continuity and smoothness of adjacent curves at knots. Although it is easy to directly construct piecewise models by truncated power basis functions, this approach is computationally unstable due to its imbalanced design matrix; instead, B-spline bases are commonly used for constructing piecewise polynomial regression models.

Without loss of generality, let the function $f(t)$ span over the dynamic variable t from 0 to 1, i.e., $0 \leq t \leq 1$, and there are L number of interior knots within this range. The basis functions of a B-spline of order m are polynomial functions with the degrees $m-1$. Expand the knot set by adding m additional knots at each end of t spectrum and order these knots. Let $\boldsymbol{\tau}$ to be the order knot vector; i.e., $\boldsymbol{\tau} = [\tau_1, \tau_2, ...\tau_{L+2m}]$, where $\tau_1 = \tau_2 = ... = \tau_m = 0$ and $\tau_{L+m+1} = \tau_{L+m+2} = ... = \tau_{L+2m} = 1$. Then, the

m-order B-spline basis functions can be expressed in the following recursive form (De Boor *et al.* (1978)):

$$B_{k,1}(t,\boldsymbol{\tau}) = \begin{cases} 1 & \tau_k \leq t < \tau_{k+1} \\ 0 & \text{otherwise} \end{cases}$$

$$B_{k,m}(t,\boldsymbol{\tau}) = \frac{t-\tau_k}{\tau_{k+m-1}-\tau_k} B_{k,m-1}(t,\boldsymbol{\tau}) + \frac{\tau_{k+m}-t}{\tau_{k+m}-\tau_{k+1}} B_{k+1,m-1}(t,\boldsymbol{\tau}), \tag{5.1}$$

where k is the index of knots; i.e., $k = 1, 2, ..., L + 2m$. Thus, a B-spline function, $S(t)$, is defined as

$$S(t) = \sum_{k=1}^{L+2m} c_k B_{k,m}(t,\boldsymbol{\tau}),$$

where c_k is the coefficient for each basis, and $B_{k,m}(t,\boldsymbol{\tau})$ will be shown as $B_k(t)$ in abstract.

An order-m B-spline function with L interior knots has $p = L+m$ number of non-zero basis functions. The sum of these basis functions at any t equals to 1. These B-spline basis functions are compact, which refers to the property that an order-m B-spline basis is non-zero in at-most m adjacent segments between knots. This property allows a design matrix from B-spline model to have few non-zero entries. An order-4 B-spline function has polynomial bases with degrees of 3, or cubic polynomial bases. Then, these functions can have up-to-2nd-order differentiations equated at knots and make the whole curve smooth to visual inspection. Figure 5.2 (a) shows a set of order-4 B-spline basis functions with internal knots at $\{0.3, 0.6, 0.9\}$. The differentiability at knots can be reduced by adding replicated knots. For example, Figure 5.2 (b), (c), (d) have additional 1, 2, 3 replicates of knot 0.6, respectively, which result in a basis system that has one-degree continuous derivative, zero-degree continuous derivative but continuous function, and discontinuous function, respectively, at this point.

Using B-splines to model a dynamic response yields $Y(t) = S(t) + \varepsilon = B^T(t)\boldsymbol{\theta} + \varepsilon$, where $B^T(t)$ is the transpose of a basis function vector, $\boldsymbol{\theta}$ is a vector of coefficients

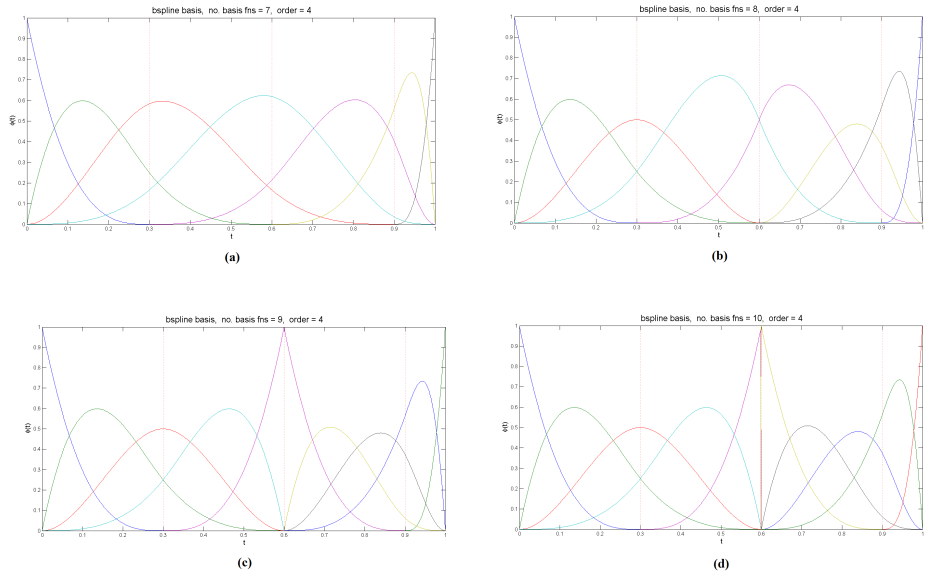


Figure 5.2: The Order-4 B-Spline System with 7,8,9 and 10 Basis Functions That Are Derived from the Internal Knots Located at {0.3,0.6,0.9},{0.3,0.6,0.6,0.9}, {0.3,0.6,0.6,0.6,0.9} and {0.3,0.6,0.6,0.6,0.6,0.9}, Respectively.

and $\varepsilon \sim N(0, \sigma^2)$. Suppose there are N dynamic response profiles and M observations at $t_1, t_2, ..., t_M$ on each profile, then the response vector of each profile is $\mathbf{y}_j = [y_{1j}, y_{2j}, ..., y_{Mj}]^T$, $j = 1, 2, ..., N$. Let the response matrix be as $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N]$, then

$$\mathbf{Y}(\mathbf{t}) = \mathbf{B}(\mathbf{t})\boldsymbol{\Theta} + \boldsymbol{\varepsilon}, \tag{5.2}$$

where $\mathbf{B}(\sqcup)$ is the design matrix of basis functions and its elements are as $b_{ik} = B_k(t_i)$, $i = 1, 2, ..., M$; $\boldsymbol{\Theta}$ is a matrix of coefficients; $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$. As aforementioned, the sum of all elements in each row of $\mathbf{B}$ matrix equals to one. This constraint needs be adopted in developing efficient algorithms for finding optimal design matrix in the next section.

The following example demonstrates the flexibility of B-spline function on modeling dynamic response. The data plotted in Figure 5.3 are taken from Binde *et al.* (2012), which are the mass of soot emitted from diesel engine over crank angle from 0 to 50 degrees. Consider the order-4 B-spline function with equal distance interior

knots. Figure 5.3 shows the fitted curves with 9 interior knots (13 basis functions) and with 19 interior knots (23 basis functions). One can see that with more interior knots (or number of basis functions) more local behaviors of the dynamic response can be captured by the model. It is difficult to obtain this level of flexibility by a parametric model. On the other hand, one should be careful of selecting the number of knots and their locations of B-spline model, as an unwise selection may cause over-fitting or under-fitting of the data. For designing an experiment with dynamic responses, these knots can be selected based on the prior engineering knowledge of the system; thus, any optimal design derived from a B-spline model is locally optimal to the specific prior knowledge only. Robust experimental designs under the uncertainty of knots selection will be discussed later in this chapter.

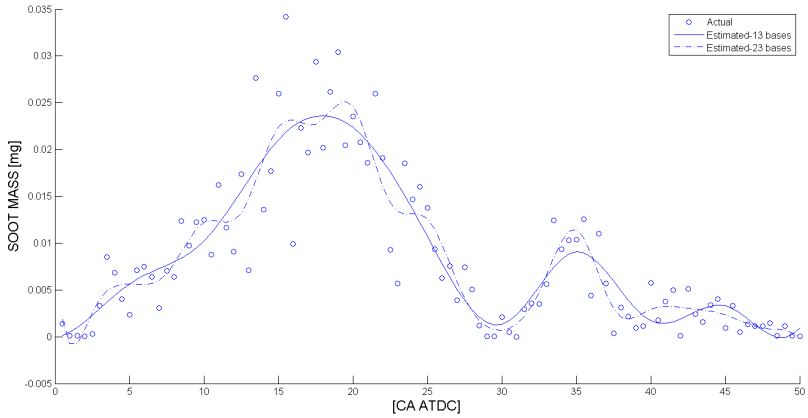


Figure 5.3: Soot Mass Data for Crank Angle after Top Dead Center [CA ATDC] between 0^O to 50^O Are Shown by Circles. Two Different B-Spline Bases System Are Fitted to the Data. One Can See the Local Behavior of the Models Boost by Increasing the Number of Bases Significantly

5.2.2 Mixed Effects Model

Beside of specifying the sampling points on dynamic outputs, an experimental design of dynamic response system concerns with the study of effects of experimental factors on the system's dynamic behavior. To model these effects, a popular approach

is the hierarchical modeling approach, in which the coefficients in spline model (5.2) are functions of experimental factors (see e.g., Wu and Hamada (2011a), Tsui (1999), Del Castillo et al. Del Castillo *et al.* (2012), and Verbeke and Molenberghs (2009a)).

Consider an experiment that consists of multiple treatments on experimental units and the outputs from each experimental unit are measured over time. The hierarchical modeling approach has two stages – first, the response curve of each experimental unit is modeled by a spline function; second, the coefficients of spline function are modeled as functions of treatments.

This approach yields:

$$\mathbf{y}_j = \mathbf{B}(\mathbf{t})\boldsymbol{\theta}_j + \boldsymbol{\epsilon}_j \quad \boldsymbol{\epsilon}_j \sim N_M(0, \boldsymbol{\Sigma}), \tag{5.3}$$

and

$$\boldsymbol{\theta}_j = \mathbf{H}\mathbf{f}(\mathbf{x}_j) + \boldsymbol{\omega}_j \quad \boldsymbol{\omega}_j \sim N_M(0, \boldsymbol{\Sigma}_\omega), \tag{5.4}$$

where $\mathbf{x}_j$ is the vector of experimental factors applied on the j-th experimental unit and $\mathbf{f}(\mathbf{x}_j)$ is the transformation of experimental factors, and $\mathbf{H}$ is a matrix of unknown model parameters. Thus, the stage-1 model smooths the actual observed data profile, $\mathbf{y}_j$, individually, and the stage-2 model assesses the relationships between smoothing parameters and experimental factors.

Del Castillo *et al.* (2012) and Verbeke and Molenberghs (2009a) proposed to combine Equations (5.3) and (5.4) to derive the mixed effects model such as

$$\mathbf{y}_j = \mathbf{B}(\mathbf{t})[\mathbf{H}\mathbf{f}(\mathbf{x}_j) + \boldsymbol{\omega}_j] + \boldsymbol{\epsilon}_j = \mathbf{B}(\mathbf{t})\mathbf{H}\mathbf{f}(\mathbf{x}_j) + \mathbf{B}(\mathbf{t})\boldsymbol{\omega}_j + \boldsymbol{\epsilon}_j, \tag{5.5}$$

Using Kronecker product of two matrices, the first term of the right hand side of the equation can be rewritten as $(\mathbf{f}(\mathbf{x}_j)^T \otimes \mathbf{B})\mathbf{vec}(\mathbf{H})$, where $\mathbf{vec}()$ operator stacks columns of $\mathbf{H}$ to one column. Then, the mixed effects model of the j-th dynamic response becomes

$$\mathbf{y_j} = \mathbf{X}_j\boldsymbol{\beta} + \mathbf{B}\boldsymbol{\omega}_j + \boldsymbol{\epsilon}_j, \tag{5.6}$$

where $\mathbf{X}_j = \mathbf{f}(\mathbf{x}_j)^T \otimes \mathbf{B}$ and $\boldsymbol{\beta} = \mathbf{vec}(\mathbf{H})$. It is easy to show the variance of $\mathbf{y}_j$ is given by $\mathbf{V}_j = \boldsymbol{\Sigma} + \mathbf{B}(\mathbf{t})\boldsymbol{\Sigma}_w\mathbf{B}(\mathbf{t})^T$.

Given that there are multiple experimental units and each of them generates one response curve, the mixed effects model becomes

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + (\mathbf{I_N} \otimes \mathbf{B})\boldsymbol{\omega} + \epsilon, \tag{5.7}$$

where $NM \times 1$ vector $\mathbf{Y}$ is equal to $[\mathbf{y_1}, \mathbf{y_2}, ..., \mathbf{y_N}]$, $NM \times pq$ matrix $\mathbf{X}$ is $(\mathbf{I_N} \otimes \mathbf{B})\mathbf{F}(\mathbf{x})$, where $Np \times pq$ matrix $\mathbf{F}(\mathbf{x}) = [\mathbf{I_p} \otimes f(\mathbf{x_1}), \mathbf{I_p} \otimes f(\mathbf{x_2}), ..., \mathbf{I_p} \otimes f(\mathbf{x_N})]^T$. $\boldsymbol{\beta}$, the fixed unknown parameters of the model, is equal to $[\boldsymbol{\beta}_1^T, \boldsymbol{\beta}_2^T, ..., \boldsymbol{\beta}_p^T]^T$, where $\boldsymbol{\beta}_k^T = [\beta_{k1}, \beta_{k2}, ..., \beta_{kq}]$. Finally, the random unknown parameters of the model, $\boldsymbol{\omega}$, is equal to $[\boldsymbol{\omega}_1^T, \boldsymbol{\omega}_2^T, ..., \boldsymbol{\omega}_N^T]^T$ where $\boldsymbol{\omega}_j^T = [w_{j1}, w_{j2}, ..., w_{jp}]$.

The maximum likelihood estimation of unknown parameters in the mixed effects model provided above is

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{V^{-1}}\mathbf{X}^T)^{-1}\mathbf{X}^T\mathbf{V^{-1}}\mathbf{y}. \tag{5.8}$$

This estimator is unbiased to the parameter being estimated. The covariance of these estimates is given by

$$COV(\hat{\boldsymbol{\beta}}) = (\mathbf{X}^T\mathbf{V^{-1}}\mathbf{X})^{-1}, \tag{5.9}$$

where $\mathbf{V} = \Sigma + (\mathbf{I_N} \otimes \mathbf{B})\Sigma_{\boldsymbol{\omega}}(\mathbf{I_N} \otimes \mathbf{B})^T$. To obtain a D-optimal experimental design, one needs to minimize the determinant of $COV(\hat{\boldsymbol{\beta}})$ or maximize the determinant of information matrix; therefore, the D-optimal criterion is defined as

$$D_{\boldsymbol{\beta}} := \max_{\mathbf{X}} |\mathbf{X}^T\mathbf{V^{-1}}\mathbf{X}|. \tag{5.10}$$

Mixed effects models have been applied on a wide variety of experimental design studies. For examples, Goos and Jones (2011) used this model for designing split-plot experiments; Laird and Ware (1982) used it to study the repeated measurement

problem; Liu and Frank (2004) and Kao *et al.* (2009) applied it on fMRI experiments. However, the experimental response we considered is much more complicated than those in previous studies. We will derive the optimal experimental plan for both the sampling time of dynamic response and the setting of experimental factors on individual experimental unit.

The design matrix of mixed effects model, $\mathbf{X}$, is a sparse matrix. This matrix is constructed by the multiplication of stacked basis matrix $\mathbf{B}$ and experimental design points $\mathbf{f}(\mathbf{x})$. According to the B-spline bases properties, order-m B-spline bases are nonzero only at the m adjacent intervals separated by knots. Therefore, in the case of using order-4 B-spline function to model a dynamic response with 10 interior knots, there are 14 basis functions, but each basis function has non-zero values in 4 adjacent intervals only, so at any sampling point there are at most 4 non-zero basis values. Note that, if there is one sampling point in each interval, the basis matrix will become a banded matrix with a bandwidth of 4. For example, in the experiment give by Grove *et al.* (2004) there are 55 subjects, 3 independent factors and 7 observations on each response curve, using the order-4 B-spline model yields a design matrix of size 385×21 $((NM) \times (pq))$. To make all model parameters estimatible, only 385 nonzero entries are needed in this matrix, which is 5% of the size of design matrix.

5.3 Optimal Sampling Times for Dynamic Response

Finding optimal sampling times of responses is a unique problem that would not be seen in experiments with static responses. Sampling is required when there is a high cost associated with the response measurement. Gaffke *et al.* (1999) presented D-optimal designs for B-spline regression models and their designs were taking approximate design forms. This relaxation enables statisticians to find an explicit formula for the optimal solution, but this solution in the approximate design form may not

be feasible in practice, because the weight values of design points in an approximate design may not be able to be transformed to integers for a given sample size. Exact designs that are obtained from exchange algorithms are considered in this chapter.

5.3.1 D-Optimal Design of Sampling Time

In this section, we discuss the D-optimal sampling times for functional data in order to estimate the $\boldsymbol{\theta}$ parameter vector in Equation (5.2) accurately. The covariance matrix for this linear model is equal to

$$COV(\hat{\boldsymbol{\theta}}) = (\mathbf{B}^T\mathbf{B})^{-1},$$

where the observations are assumed to be independent and have the equal variance. Therefore, the D-optimal criterion can be specified as

$$D_\theta := \max_{\mathbf{B}} |\mathbf{B}^T\mathbf{B}| \tag{5.11}$$

$$S.T. \quad \mathbf{b}_i\mathbf{1} = 1, \quad \textit{for all } i\textit{'s.}$$

where $\mathbf{b}_i$ is the ith row of $\mathbf{B}$. It is defined by $\mathbf{b}_i = [B_1(t_i)\ B_2(t_i)...\ B_p(t_i)]$, where $p = m + l$, and $B_k(t_i)$ is given by Equation (5.1) evaluated at the sampling time t_i. The constraint simply states that each row of B must sum to unity.

To provide a general idea of what an optimal sampling plan would be like for a B-spline model, we plot two different B-spline basis systems in in Figure 5.4 and Figure 5.5, along with their optimal sampling times. One can see these optimal sampling times are either on or close to the locations where one basis function has its maximum value. The property can be explained by following theorems.

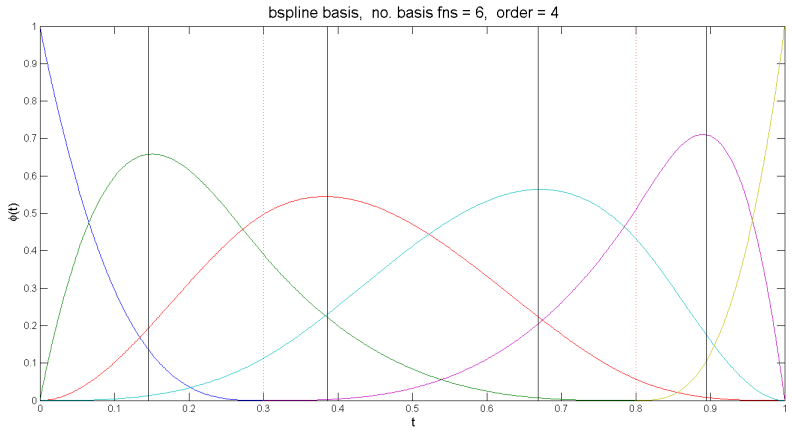


Figure 5.4: Six Bases for an Order 4 B-Spline System with Internal Knots Located at $\boldsymbol{\tau} = \{0.3, 0.8\}$. Optimal Sampling Times Are Depicted by Solid Lines, Where the Dotted Lines Indicates the Location of the Knots. Sampling Time Vector for the New Approach Is $\mathbf{t} = \{0, 0.145, 0.385, 0.669, 0.895, 1\}$

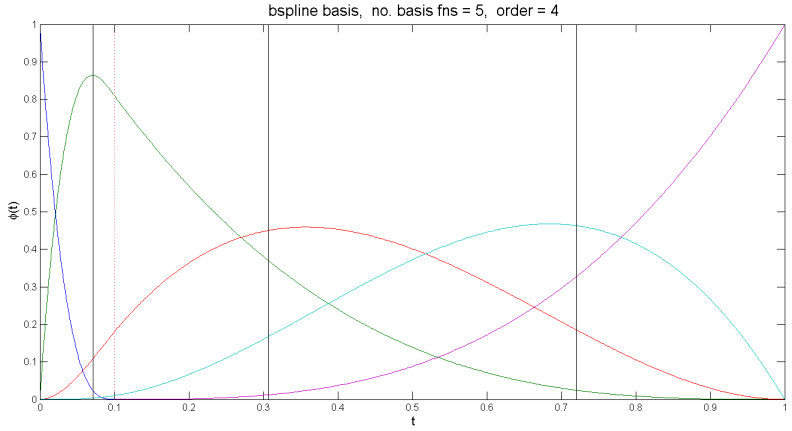


Figure 5.5: Five Bases for an Order 4 B-Spline System with Internal Knot Located at $\boldsymbol{\tau} = \{0.1\}$. Optimal Sampling Times Are Depicted by Solid Lines, Where the Dotted Lines Indicates the Location of the Knots. Sampling Time Vector for the New Approach Is $\mathbf{t} = \{0, 0.071, 0.307, 0.72, 1\}$.

Lemma 5.3.1 *Let the symmetric square matrix* $\mathbf{M}$ *be a positive definite matrix with non-negative elements, then its determinant can be calculated by*

$$|\mathbf{M}| = \prod_{k=1}^{p} m_{kk} - \omega_T, \tag{5.12}$$

where m_{ii}*s are the diagonal elements of* $\mathbf{M}$ *and* ω_T *is a term involved the second-order cofactors of diagonal elements and it is always positive.*

The proof can be extended from the Cauchy's expansion of the determinant of a positive definite matrix. As a result, to maximize the determinant function, we can try to increase the values of diagonal elements and reduce the values of non-diagonal elements at the same time. In general, if there exists a B-spline basis matrix with all non-diagonal elements equal to zero and $Tr(\mathbf{M}) = J$, where $\mathbf{M} = \mathbf{B}^T\mathbf{B}$ and J is the number of rows in B-spline basis matrix, then $\mathbf{B}$ is universally optimal design (see Kiefer (1974)).

Now, considering the B-spline basis matrix defined in Equation (5.11) and $\mathbf{M} = \mathbf{B}^T\mathbf{B}$, it is easy to show that the summation of all elements in $\mathbf{M}$, $\sum_i \sum_j m_{ij} = M$, where M is the number of rows of $\mathbf{B}$ or the number of sampling times. This property implies that increasing the values of diagonal elements in $\mathbf{M}$ will simultaneously decrease the values of non-diagonal elements in $\mathbf{M}$, when $\mathbf{M}$ is the information matrix of the B-spline design matrix $\mathbf{B}$.

Theorem 5.3.2 *Let* $\mathbf{t} = [t_1, t_2, ..., t_M]$ *be the ordered sequence of optimal sampling times for a dynamic system modeled by a B-spline model, then the two end points of the spectrum variable must be included in this sequence, i.e.,* $t_1 = 0$ *and* $t_M = 1$.

This theorem can be proved by applying the previous lemma. Suppose $\mathbf{B}$ is the design without including $t_1 = 0$ or $t_M = 1$ sampling time. Based on the Cauchy's expansion theorem and also Laplace's formula, a D-optimal design can be found by maximizing the diagonal elements in $\mathbf{B}^T\mathbf{B}$ and minimizing non-diagonal elements at the same time. Constant summation property of the B-spline matrix shows replacing a row in the design matrix by another one does not change the summation of the elements in the information matrix. Therefore, changing the first row of $\mathbf{B}$ to be [1 0 0 ... 0] and the last row to be [0 0 ... 0 1] will increase the determinant of information

matrix. Thus, $t_1 = 0$ and $t_M = 1$ must exist in the sequence of optimal sampling times.

We can apply the same argument to other sampling times in the optimal sequence. As the spline function is supported by $p = m + L$ bases, it requires at least p sampling times to make all coefficients estimatible. To have the diagonal elements of information matrix to be large while non-diagonal elements to be small, the corresponding diagonal elements in $\mathbf{B}$ should be large, which implies that optimal sampling times should be located around the time when one basis function reaches its maximum. Although we have not found a precise proof, this speculation has been supported by all numerical examples we have tried. On the other hand, we can utilize this insight to reduce the size of candidate points for constructing the optimal sampling time sequence by using exchange algorithms. This idea will be further elaborated in the next section.

Theorem 5.3.3 *Let* $\mathbf{M}$ *be the information matrix corresponding to a B-spline design matrix* $\mathbf{B}$*. Suppose this B-spline system has its interior knots equal-distancedly placed between the two ends of the spectrum variable, then optimal sampling times must be symmetrically located between 0 and 1.*

With uniformly spaced internal knots, it is realized that a basis function of B-spline are symmetric to another basis function or itself. Using the Cauchy's expansion, it can be shown that if the time t, $t < 0.5$, is included in the optimal sampling sequence, then $1 - t$ must also appear in the sequence. Assume $\mathbf{b}_1$ and $\mathbf{b}_2$ are the vectors that correspond to the sampling time t and $1 - t$, respectively. If $\mathbf{b_1}$ is a row vector to be augmented to $\mathbf{B}$, the new information matrix can be calculated as $\mathbf{M} + \mathbf{M}_1^d$, where $\mathbf{M}_1^d = \mathbf{b}_1^T \mathbf{b}_1$. Matrix $\mathbf{M}_1^d$ is a sparse matrix with a block of nonzero elements. This block is similar to the nonzero block in $\mathbf{M}_2^d$ calculated by $\mathbf{b}_2^T \mathbf{b}_2$. This similarity,

between $\mathbf{M_1^d}$ and $\mathbf{M_2^d}$, is caused by the symmetrical behavior of bases due to uniformly spaced internal knots. As a result, sampling at t or $1-t$ has similar impact on the information matrix. Therefore, if one of them appear in the optimal sequence the other one must also appear.

5.3.2 Algorithm for Finding D-Optimal Sampling Times

Properties of an optimal $\mathbf{B}$ matrix are discussed in the previous section and these properties will be utilized in this section to develop a deterministic search algorithm for finding the D-optimal sampling plan. Similar to PEA, the proposed algorithm requires a set of candidate points. Each row of matrix $\mathbf{B}$ (a design point) corresponds to a sampling time; i.e., for a time t there is a row vector $[b_1\ b_2\ ...\ b_p]$. Define an objective function to be

$$obj(t) = \max\{b_i^2\} - \lambda \sum_i \sum_{j>i} b_i b_j \tag{5.13}$$

We discretize the spectrum variable from 0 to 1 to give a list of t values. Then, with the list of $obj(t)$ values we find all local maximums and save their corresponding t's to the candidate set. Parameter λ in Equation (5.13) is a regularization parameter. This parameter eventually controls the trade-off between achieving a large increase in the diagonal element of information matrix and a decrease in the non-diagonal elements. This parameter needs be tuned before the implementation of our algorithm.

Starting from a random initial design where sampling times are randomly assigned between 0 and 1, our algorithm replaces these sampling times by the times in the candidate set one by one. At each iteration, the Fedorov delta function will be evaluated for assessing the improvement in the determinant of information matrix when a current sampling time is replaced by a candidate sampling time. The iteration

terminates when there is no more replacement that can increase the determinant of information matrix.

Data: $\boldsymbol{\tau}, \mathbf{t}, m, L, \mathbf{F}(\mathbf{x})$

Result: Optimal design for the estimation problem

Initialization;

Generate B-spline basis functions with order-m and L internal knots;

Generate the candidate set C;

Calculate $obj(t)$ for all discretized values of t ;

$C \leftarrow \arg(\max obj(t))$;

Generate the initial design matrix ;

while $\delta^* \geq e$ **do**

  for $j \in C$ **do**

    $\delta_{ij} \leftarrow$

    delta function of replacing i-th sampling point by the jth candidate point.

  end

  $\delta^* \leftarrow \max \boldsymbol{\delta}_{i.}$;

  $t_i \leftarrow \arg(\max \boldsymbol{\delta}_{i.})$;

end

Algorithm 2: New Approach for Finding the Optimal Sampling Times

Computation times and determinants of the optimal designs obtained by our algorithm are compared with those obtained from exhaustive search over all possible sampling plans. Woods *et al.* (2003) suggested to the candidate set by choosing only sampling times around the locations where each basis function reaches its maximal value. Our approach further reduces this candidate set to include only N candidate sampling times.

Consider the examples in Figures 5.4 and 5.5. Varying the tuning parameter λ, we compare the D-efficiency of the optimal design from our algorithm to the one from exhaustive search. Figures 5.6 and 5.7 show that our algorithm is capable of reaching to the highest possible efficiency with a proper choice of λ. The computation time of our algorithm is much reduced from Woods *et al.* (2003) For the first example (Figure 5.4), the average computation time of our algorithm is 0.06 seconds, comparing to 12 seconds by Woods *et al.* (2003). In addition, Kaishev (1989) suggested to simply use the times where each basis function has its maximal value. The efficiency of this sampling plan is also marked by circle in the Figure 5.6 and Figure 5.7. It is clear that Kaishev's sampling plan is not optimal.

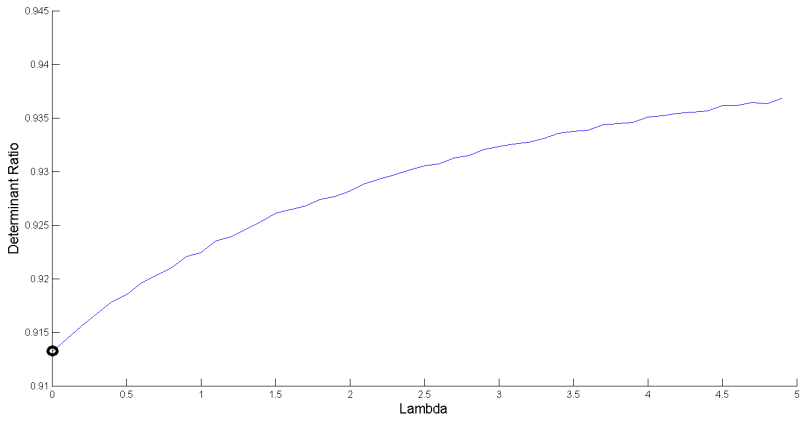


Figure 5.6: Comparing the Determinants of the Optimal Design from New Approach and the Ordinary PEA for Different λS for an Experiment with 6 Runs and Order-4 Basis System with Internal Knots at $\boldsymbol{\tau} = \{0.3, 0.8\}$

5.3.3 Robust Sampling Plans

The optimal sampling plan depends on the basis functions, thus the locations of interior knots, of B-spline system. The selection of knots in turn depends on the experimenter's knowledge of the shape of response curve. Therefore, uncertainties in this prior knowledge at the experimental design stage require the experimenter

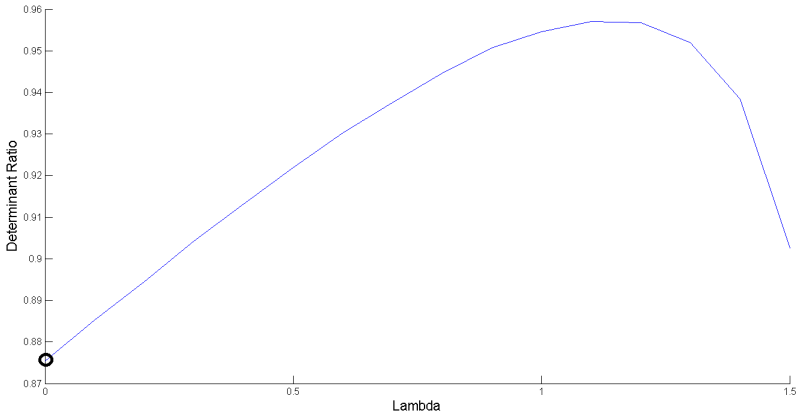


Figure 5.7: Comparing the Determinants of the Optimal Design from New Approach and the Ordinary PEA for Different λS for an Experiment with 5 Runs and Order-4 Basis System with Internal Knot at $\boldsymbol{\tau} = \{0.1\}$

to consider a robust design. In the following example, five different basis systems with different locations of internal knots are used. Optimal sampling times for five systems are shown in Figure 5.8. Then, we apply the k-means clustering algorithm to cluster these optimal sampling times into k clusters, where k is less than the total number of sampling times determined by the experimenters. In the next step, the centroids of these clusters are stored in the candidate set and exchange algorithm is applied to construct the robust sampling plan, where the objective function is set as the median of D-efficiency for the all basis systems considered. Figure 5.9 shows the robust design for the five B-spline systems provided in Figure 5.8.

5.4 Optimal Design of Experiments with Dynamic Responses

In Section 2.2 we modeled the dynamic response system by a mixed effects model such as Equation (5.7). The D-optimal experimental design for such system is obtained by applying the D-optimal criterion (5.11). However, design matrix $\mathbf{X}$ in Equation (5.7) is large matrix and it is a function of the B-spline basis matrix $\mathbf{B}$ too. Direct application of exchange algorithm on finding optimal $\mathbf{X}$ is unpractical. In this

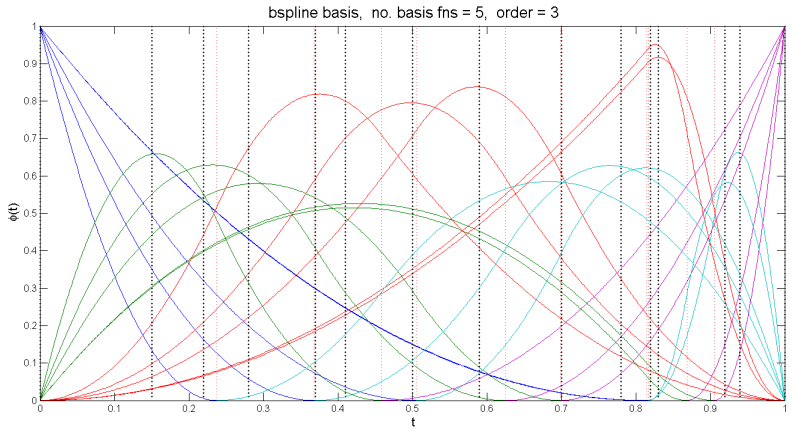


Figure 5.8: Five B-Spline Basis Systems with Order Three and Two Internal Knots at Random Locations. Optimal Sampling Times for Different Bases Are Depicted by Solid Lines, Where the Dotted Lines Indicates the Location of the Knots.

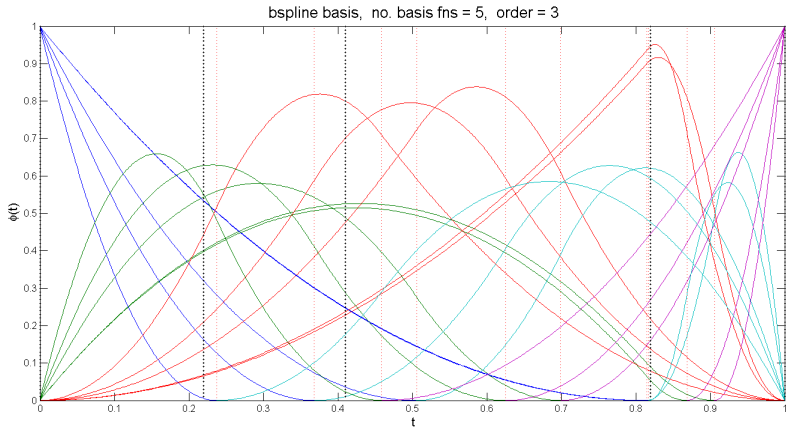


Figure 5.9: Robust D-Optimal Design for 5 B-Spline Systems Provided in Figure 5.8. Each Sampling Time Has Two Replicates for an Experiment with 10 Sampling Times. Optimal Sampling Times for Different Bases Are Depicted by Solid Lines, Where the Dotted Lines Indicates the Location of the Knots

section, we will use two steps to find the optimal design of experiments with dynamic responses.

5.4.1 The Two-Step Approach

The first step is to find the optimal sampling times for a given B-spline basis system. This is the same as maximizing the information matrix of Model (5.3),

Table 5.1: Optimal Sampling Times for an Order-4 B-Spline Basis System

Number of Samples	Optimal Sampling Times ($\mathbf{t}$)
6	{0,0.12,0.33,0.6,0.85,1}
7	{0,0.12,0.33,0.6,0.85,0.85,1}
8	{0,0.12,0.33,0.6,0.6,0.85,0.85,1}
9	{0,0.12,0.33,0.33,0.6,0.6,0.85,0.85,1}
10	{0,0.12,0.12,0.33,0.33,0.6,0.6,0.85,0.85,1}

$\mathbf{B}(\mathbf{t})^T\mathbf{B}(\mathbf{t})$. For example, consider a order-4 B-spline basis system with two interior knots at $\boldsymbol{\tau} = \{0.3, 0.6\}$. These basis functions are plotted in Figure 5.10 and the optimal sampling times with different number of samples are listed in Table 5.1.

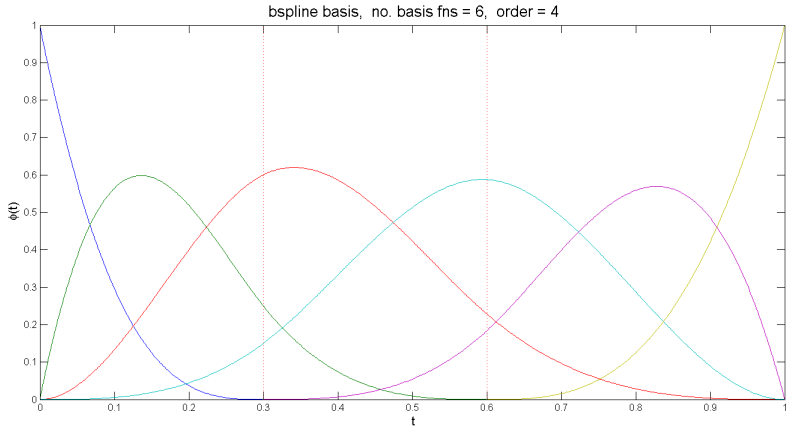


Figure 5.10: The Order-4 B-Spline Bases System with Internal Knots Located at $\boldsymbol{\tau} = \{0.3, 0.6\}$

The number of bases of a B-spline system depends on the number of knots assigned to the system. Reducing the number of bases may result in losing modeling flexibility of some local behaviors of response curves under certain experimental conditions; while increasing the number of bases requires the experimenter to have more prior knowledge of the dynamic response and increases the complexity of experimental design.

After finding optimal sampling times, the second step is to find the optimal experimental condition for each experimental unit, i.e., the optimal $\mathbf{X}$ in Model (5.7).

Table 5.2: Optimal Experimental Designs of an Order-4 B-Spline System with 3 Experimental Factors and 55 Experimental Units

	Sampling Strategies										
x (experimental conditions)	Two-step	Random-6	Random-7	Random-8	Random-9	Random-10	Equal-6	Equal-7	Equal-8	Equal-9	Equal-10
$\{-1,-1,-1\}$	14	11	14	14	14	11	14	13	12	14	14
$\{-1,-1,1\}$	13	5	10	11	5	5	13	11	4	12	13
$\{-1,1,-1\}$	14	8	12	13	8	8	14	14	8	13	14
$\{-1,1,1\}$	0	6	4	3	9	7	14	5	7	0	0
$\{1,-1,-1\}$	14	8	9	10	4	7	0	9	6	4	0
$\{1,-1,1\}$	0	5	2	1	6	6	0	0	6	0	0
$\{1,1,-1\}$	0	0	4	3	9	9	0	2	10	2	14
$\{1,1,1\}$	0	9	0	0	0	2	0	1	2	0	0

Note the $\mathbf{X} = (\mathbf{I_N} \otimes \mathbf{B})\mathbf{F}(\mathbf{x})$. With $\mathbf{B}$ is fixed, we applied exchanged algorithm to find the optimal $\mathbf{F}(\mathbf{x})$ to maximize the D-optimal design objective given in Equation (5.10).

To compare this two-step approach to other methods, we consider the previous example of order-4 B-spline system with 3 experimental factors and 55 experimental units. The range of each factor is scaled to -1 to 1, so the design region is a cube. Beside of the two-step approach, we apply two other approaches – optimizing D-objective (5.10) with randomly chosen sampling times or uniformly spaced sampling times. The designs derived from these approaches are listed in Table 5.2. We varied the number of sampling times from 6 to 10. However, using the two-step approach, the selected experimental conditions are the same for any number of sampling times, so they are listed in one column. The numbers in each column of Table 5.2 are the number of experimental units assigned to the corresponding experimental conditions. The determinants of the information matrices of these designs are given in Table 5.3. One can see that the two-step approach is clearly superior than the other two approaches in terms of providing designs with larger determinant values of information matrix.

Table 5.3: Determinants of the Information Matrices of Experimental Designs Derived from Three Approaches.

	Determinants for Sampling Strategies		
Number of Sampling Times	Two-step	Random	Equal
6	2.75E+17	1.02E+10	2.16E+16
7	1.42E+18	2.43E+13	4.41E+17
8	7.34E+18	3.28E+15	2.91E+18
9	3.74E+19	1.61E+16	1.17E+19
10	1.91E+20	1.37E+17	3.87E+19

5.4.2 Two Engineering Examples

In this section, we apply the two-step approach for finding optimal experimental designs of dynamic response systems on two engineering examples found in literature, and compare them with standard designs and engineer suggested designs. The standard design is obtained by uniformly placing sampling times combined with D-optimal design of experimental factors, while the engineer suggested design is taken from the literature.

The first example concerns with designing an electrical alternator (see Nair *et al.* (2002)). The response variable is electric current. The spectrum variable is revolutions per minute (RPM) and it is sampled at $\{1375, 1500, 1750, 2000, 2500, 3500, 5000\}$ for 108 designed alternators (see Figure 5.11). After scaling the range of spectrum variable to [0, 1], we have these sampling RPMs at $\{0, 0.03, 0.1, 0.17, 0.31, 0.58, 1\}$. Eight controllable factors and two noise factors are considered in this example.

We use an order-4 B-spline system to model the response profiles over RPM and place two interior knots at $\boldsymbol{\tau} = \{0.3, 0.6\}$. These basis functions are plotted in Figure 5.10. Then, the optimal sampling RPMs are located at $\{0, 0.12, 0.33, 0.6, 0.85, 0.85, 1\}$. The determinants of information matrices of the engineer suggested design in Nair

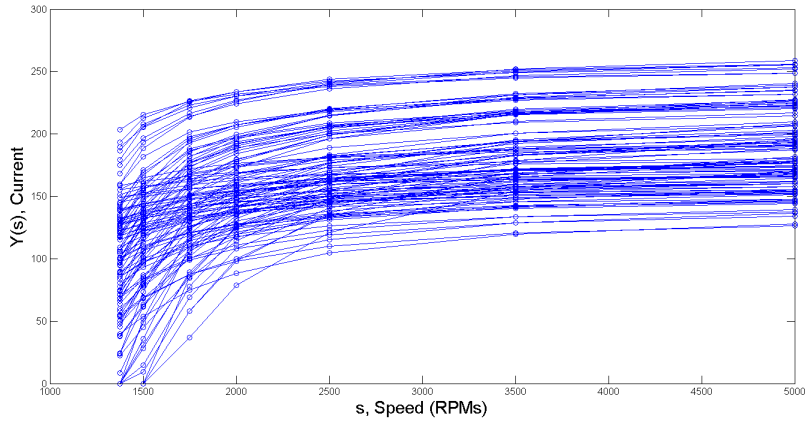


Figure 5.11: 108 Profile Curves Derived from the Experiments Conducted for Designing the Electrical Alternator. The Electric Current Values Are Marked by Circles and They Are Connected by Straight Lines.

Table 5.4: Determinants of the Information Matrix from the Optimal Design, Standard Design, and Engineer Suggested Design

Design	Determinant
Two-step	$6.04E+90$
Standard Design	$1.30E+89$
Engineer Suggested Design	$4.08E+48$

et al. (2002), the standard design, and the D-optimal design derived from the two-step approach are compared in Table 5.4.

The second example is taken from Grove *et al.* (2004). It is an engineering-mapping problem, where brake torque is studied for 55 different spark sweeps where the spark advance is varied for each sweep (see Figure 5.13). In this example, the spark degrees may be different for each sweep and three controllable factors – speed, load and AFR – can be varied in the experiment. Again, we scale the range of spark advance to [0, 1]. We use an order-4 B-spline model with one interior knot at $\boldsymbol{\tau} = \{0.5\}$. Figure 5.12 shows the set of these basis functions.

Table 5.5 presents the optimal design obtained from the two-step approach and the standard design with uniformly spaced spark advances. Comparing the determinant

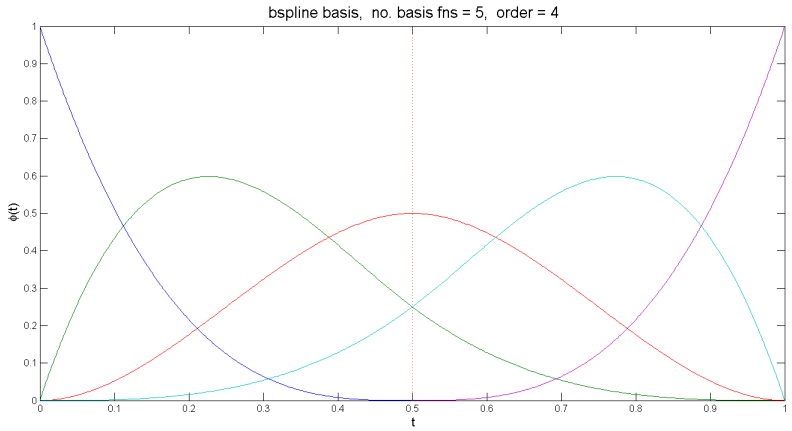


Figure 5.12: The Order-4 B-Spline Bases System with Internal Knot Located at $\boldsymbol{\tau} = \{0.5\}$

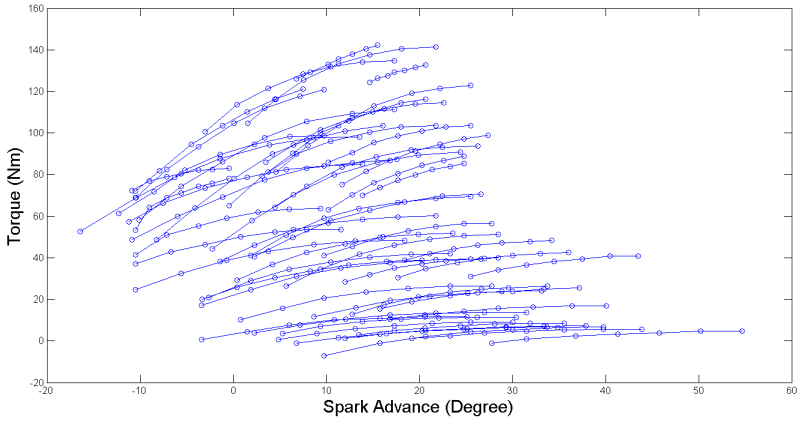


Figure 5.13: Response Profiles from the Engine-Mapping Experiments

values from these two designs, one can see the two-step approach is clearly better than the alternative.

5.5 Conclusion

Designing optimal experiments for dynamic responses need to take into account the variation over the continuum and the variation due to the changing levels of the experimental factors. In this regard, a semi-parametric model was deemed a flexible and appropriate model for designing optimal experiments for dynamic responses.

Table 5.5: Two Design Alternatives for the Engineer-Mapping Example

Design	Spark Advance ($\mathbf{t}$)	Determinant
Two-step	{0,0.18,0.18,0.5,0.82,0.82,1}	$7.24E+15$
Standard Design	{0,0.16,0.33,0.5,0.66,0.83,1}	$4.07E+15$

The model can be perceived as a hierarchical modeling strategy, where at the first stage, a non-parametric model is used to smooth out the response curves over the continuum, while in the second stage, the estimated parameters from the first model are regressed against the experimental factors. The juxtaposition of the two is simply a mixed effects model, as proposed in Del Castillo *et al.* (2012).

This chapter first considered the sampling point design problem. A fast, computational algorithm was developed to determine the optimal sampling points. The sampling points could be time in a longitudinal study or any continuous variable that serves as the continuum for the functional responses. In this chapter, we explored three different cases of the sampling points namely, time, RPM, and the angle of spark advances in both hypothetical and real-world studies. For this problem, the B-spline basis system was used as the smoothing model, both for its flexibility and special mathematical properties that facilitate faster computation.

The corresponding design matrix for the B-spline is sparse. We exploited this property and used Cauchy's expansion to simplify the computation of the determinant at each iteration and to restrict the set of possible optimal sampling points. These improvements resulted in a more efficient algorithm. In comparison to the traditional point-exchange-algorithm, the proposed method performed almost 200 times faster and yielded relatively high D-efficiencies.

The constructed optimal designs are not robust to deviations from the assumed B-spline functions. The optimal design is sensitive to changes in the number and location of the knots, as well as the order and the number of basis functions. A simple

way to incite robustness into a design is to reducing the number of basis functions, which has its own disadvantage of smoothing out local trends. We explored a robust design method using k-means clustering and demonstrated how to use this approach for the case of finding robust optimal sampling points.

After the demonstrating the capability of the proposed algorithm to calculate optimal sampling points, we considered the two-step approach to find the optimal design for both sampling points and experimental settings. In the first step, the matrix of experimental settings, $\mathbf{D}$, was held fixed while the algorithm optimized the determinant of the information matrix for a mixed effects model to find the optimal sampling times. In the second step, the optimal sampling times obtained from the first stage were held fixed while the algorithm iterated on the information matrix to find the optimal experimental settings. The designs constructed using the proposed algorithm yielded superior performance over other designs, as examples demonstrated by found in the literature.

Chapter 6

SUMMARY AND CONCLUSION

This thesis examined optimal design methods for functional or dynamic responses. Functional or dynamically varying responses abound in applications in engineering, medicine, and the sciences. While designing experiments for static-response studies is a well-tapped area in literature, proposals for functional responses still require in-depth examination. Analyzing functional data sets requires smoothing out the variation over the spectrum variable, resulting in functional data models that are more complicated than their static counterparts.

We examined two semi-parametric models for functional data–the mixed-effects model and the varying coefficient models. The intent was to find a practicable model form for our optimization problems. The mixed-effects model is an amalgamation of two stages in hierarchical modeling. In the first stage, the functional responses are smoothed with respect to the spectrum variable, and in the second, the parameters from the first-stage model are regressed against experimental factors. The choice of non-parametric smoother in the first stage is the subject of some researches. In this study, we opted for the B-spline basis system. The B-spline system possesses attractive mathematical properties for modeling and design. It is flexible for various forms of the response functions, such that its parameters can be tightened to closely capture local behaviors on the response curve or loosened to avoid over-fitting. Del Castillo *et al.* (2012) had proposed a mixed-effects model, and we adopted this model for study with incorporating the B-spline into the second-stage model. The varying coefficient model, on the other hand, fits different parameters at every sampling point, and then uses a smoother for the estimated parameters in the second stage. We compared

and contrasted the fitting and predictive capabilities of these two models for some examples found in the literature.

In terms of fitting and prediction, there is no clear superior modeling technique. Each model proved superior over the other for some data sets and required the same number of parameters. However, the parameters of the mixed-effects model can easily be estimated by ordinary least squares, while the estimators for the varying coefficients model are more complicated. In addition, it is difficult to make reasonable assumptions about the functional form of the estimated parameters in the varying coefficients model, which makes it an impractical choice for optimal design tasks. In light of these findings, we proposed the mixed-effects model as the model from for designing optimal experiments for functional responses.

This dissertation focused on finding exact optimal designs for linear models and these designs are found by point exchange or coordinate exchange algorithm. The construction of exact designs requires the specification of the information matrix for a given model.

In Chapter 3, we proposed the cCEA, a clustering, two-step exchange algorithm for constructing G-optimal designs for ordinary linear models. G-optimality aims to find a design that minimizes the maximum prediction variance over the design region. By the very nature of its functional, the G-criterion is computationally tedious because of the repeated evaluations of the prediction variance over a large number of design points. The existing algorithms in literature yield exponential computing times with respect to the size of the problem. The cCEA is able to construct an optimal design with high G-efficiency in polynomial time.

In comparison with other designs in the literature, the cCEA generally produced higher G-efficiencies, flatter prediction variance profiles over the design region volume, and performed well with respect to the I-criterion. Most importantly, the computa-

tional time of the cCEA only increased polynomially as the size of the problems increased. This improvement was due to the reduced number of exchanges in the first stage. Thus, the algorithm took advantage of the PEA's property of exhaustive search to narrow down the location of the optimal points, while using the CEA's fast computation to further improve the G-efficiency.

In Chapter 4, we dealt with the construction of optimal designs in the context of fMRI studies. The HRF curve represents the amount of brain activity triggered by the presentation of a stimulus. In this research, we assumed that the sampling times (measurement points) are known and specified. This is typically the case for fMRI studies because the sampling points are controlled by the magnetic resonance (MR) scanner. The primary problem in fMRI studies is to determine the optimal sequence of the stimuli. This is not a trivial problem because the presentation of a stimulus produces a carry-over or lasting effect, further complicating the structure of the design matrix.

We proposed a fast computational algorithm for the construction of exact, D-optimal designs for the stimuli sequence. The algorithm is similar to CEA, where each element of the design matrix is perturbed until no improvements are made on the D-criterion.

Our proposed algorithms were compared with the closest work in the literature – the genetic algorithm (GA) applied to fMRI studies. In almost of the cases considered, the new algorithm showed improvements in the D-criterion, as well as improvements in computing times. The disparity between the proposed algorithm and the GA was more evident as the problem became more complex, such as in cases with more stimuli and when the presentation times of stimuli and MR scanning times were out-of-sync.

The algorithms developed for the fMRI case were further extended to accommodate general cases with functional or dynamic responses. We dealt with two types of

problems, namely, determining optimal sampling times for studies with no co-factors and determining both sampling times and optimal experimental settings when co-factors are present.

For the problem of optimal sampling times, we considered the B-spline basis system as the non-parametric base model. Design matrices resulting from B-splines are sparse, full-column rank, and possess the constant summation property. These properties enabled the use of Cauchy's expansion theorem. As a direct result of this theorem, we found that maximizing the sum of diagonal elements of information matrix and minimizing the off-diagonals is tantamount to this optimization problem. It was also proven mathematically that the vector of optimal sampling points always included the points $t = 0$ and $t = 1$. If the knots of the B-spline are chosen to be equally-spaced, then the optimal sampling times are just symmetric around the knots, cutting the required computation time in half. As a consequence of these mathematical results, an algorithm was proposed for constructing optimal sampling times that simplified the computation of the determinant at each iteration and restricted the set of possible optimal candidates. The algorithm outperformed PEA's computational time by a factor of 200 and yielded designs with high D-efficiencies.

Finally, we proposed an iterative, two-step algorithm for finding the optimal design for both sampling points and experimental settings. In the first step, the matrix of experimental settings, $\mathbf{X}$, was held fixed while the algorithm optimized the determinant of the information matrix for a mixed effects model to find the optimal sampling times. In the second step, the optimal sampling times obtained from the first step were held fixed while the algorithm iterated on the information matrix to find the optimal experimental settings. The optimal sampling points are not dependent on the experimental settings, so we found it more beneficial to find the optimal sampling points before determining the experimental settings.

The designs constructed using the proposed algorithm yielded superior performance over other designs, such as examples found in the literature.

The area of designing optimal experiments for functional or dynamic responses is still relatively novel in literature. We propose the following extensions of our work.

In our current study, we assumed that the sampling times were equal in length and every experimental condition was sampled using the same sampling strategy. This would be useful in the case where taking response samples is a very expensive activity and the number of total samples is strictly limited. In addition, dynamic experimental factors that change their values over time will be considered in our future study.

REFERENCES

Altman, N. S., "Kernel smoothing of data with correlated errors", Journal of the American Statistical Association **85**, 411, 749–759 (1990).

Atkinson, A. C. and A. N. Donev, "The construction of exact d-optimum experimental designs with application to blocking response surface designs", Biometrika **76**, 3, 515–526 (1989).

Atwood, C. L., "Optimal and efficient designs of experiments", The Annals of Mathematical Statistics pp. 1570–1602 (1969).

Besse, P. and J. O. Ramsay, "Principal components analysis of sampled functions", Psychometrika **51**, 2, 285–311 (1986).

Binde, A., S. Busch, A. Velji and U. Wagner, "Soot and nox reduction by spatially separated pilot injection", Tech. rep., SAE Technical Paper (2012).

Borkowski, J. J., "Using a genetic algorithm to generate small exact response surface designs", Journal of Probability and Statistical Science **1**, 1, 65–88 (2003).

Broudiscou, A., R. Leardi and R. Phan-Tan-Luu, "Genetic algorithm as a tool for selection of d-optimal design", Chemometrics and Intelligent Laboratory Systems **35**, 1, 105–116 (1996).

Chiang, C., J. A. Rice and C. O. Wu, "Smoothing spline estimation for varying coefficient models with repeatedly measured dependent variables", Journal of the American Statistical Association **96**, 454, 605–619 (2001).

Cleveland, W. S., E. Grosse and W. M. Shyu, "Local regression models", Statistical models in S pp. 309–376 (1992).

Cook, R. and C. Nachtrheim, "A comparison of algorithms for constructing exact d-optimal designs", Technometrics **22**, 3, 315–324 (1980).

Crowder, M. J. and D. J. Hand, *Analysis of repeated measures*, vol. 41 (CRC Press, 1990).

Dale, A. M., "Optimal experimental design for event-related fmri", Human brain mapping **8**, 2-3, 109–114 (1999).

Dale, A. M. and R. L. Buckner, "Selective averaging of rapidly presented individual trials using fmri", Human brain mapping **5**, 5, 329–340 (1997).

De Boor, C. *et al.*, "A practical guide to splines", (1978).

Del Castillo, E., B. M. Colosimo and H. Alshraideh, "Bayesian modeling and optimization of functional responses affected by noise factors", Journal of Quality Technology **44**, 2, 117–135 (2012).

Diggle, P. J. and M. F. Hutchinson, "On spline smoothing with autocorrelated errors", Australian Journal of Statistics **31**, 1, 166–182 (1989).

Eck, J., A. L. Kaas and R. Goebel, "Crossmodal interactions of haptic and visual texture information in early sensory cortex", NeuroImage **75**, 123–135 (2013).

Eubank, R. L., *Nonparametric regression and spline smoothing* (CRC press, 1999).

Eubank, R. L., C. Huang, Y. M. Maldonado, N. Wang, S. Wang and R. J. Buchanan, "Smoothing spline estimation in varying-coefficient models", Journal of the Royal Statistical Society: Series B (Statistical Methodology) **66**, 3, 653–667 (2004).

Fan, J. and J.-T. Zhang, "Two-step estimation of functional linear models with applications to longitudinal data", Journal of the Royal Statistical Society: Series B (Statistical Methodology) **62**, 2, 303–322 (2000).

Fan, S.-K. S., B. C. Jiang, C.-H. Jen and C.-C. Wang, "Siso run-to-run feedback controller using triple ewma smoothing for semiconductor manufacturing processes", International Journal of Production Research **40**, 13, 3093–3120 (2002).

Fang, K.-T., R. Li and A. Sudjianto, *Design and modeling for computer experiments* (CRC Press, 2005).

Fedorov, V., *Theory of Optimal Experiments, Preprint No. 7 LSM* (Izd-vo Moscow State University, Moscow, 1969).

Fedorov, V., *Theory of Optimal Experiments* (Academic Press, New York, 1971).

Fisher, V. A., *Optimal and efficient experimental design for nonparametric regression with application to functional data*, Ph.D. thesis, The University of Southampton (2012).

Gaffke, N., B. Heiligers and M. U. F. fuer Mathematik;, "Optimal approximate designs for b-spline regression with multiple knots", Statistical Process Monitoring and Optimization. Marcel Dekker pp. 339–358 (1999).

Goos, P. and B. Jones, *Optimal design of experiments: a case study approach* (John Wiley & Sons, 2011).

Gotwalt, C. M., B. A. Jones and D. M. Steinberg, "Fast computation of designs robust to parameter uncertainty for nonlinear settings", Technometrics **51**, 1, 88–95 (2009).

Grove, D., D. C. Woods and S. M. LEWIS, "Multifactor b-spline mixed models in designed experiments for the engine mapping problem", Journal of quality technology **36**, 4, 380–391 (2004).

Haines, L., "The application of the annealing algorithm to the construction of exact optimal designs for linear–regression models", Technometrics **29**, 4, 439–447 (1987).

Harackiewicz, J. M., K. E. Barron, J. M. Tauer and A. J. Elliot, "Predicting success in college: A longitudinal study of achievement goals and ability measures as predictors of interest and performance from freshman year through graduation.", Journal of Educational Psychology **94**, 3, 562 (2002).

Hart, J. D., "Kernel regression estimation with time series errors", Journal of the Royal Statistical Society. Series B (Methodological) pp. 173–187 (1991).

Hastie, T. and R. Tibshirani, "Varying-coefficient models", Journal of the Royal Statistical Society. Series B (Methodological) pp. 757–796 (1993).

Hastie, T. J. and R. J. Tibshirani, *Generalized additive models*, vol. 43 (CRC Press, 1990).

Heiligers, B., "E-optimal designs for polynomial spline regression", Journal of statistical planning and inference **75**, 1, 159–172 (1998).

Heredia-Langner, A., W. Carlyle, D. Montgomery, C. Borror and G. Runger, "Genetic algorithms for the construction of d-optimal designs", Journal of Quality Technology **35**, 1, 28–46 (2003).

Hoover, D. R., J. A. Rice, C. O. Wu and L. Yang, "Nonparametric smoothing estimates of time-varying coefficient models with longitudinal data", Biometrika **85**, 4, 809–822 (1998).

Johnson, M. E., L. M. Moore and D. Ylvisaker, "Minimax and maximin distance designs", Journal of statistical planning and inference **26**, 2, 131–148 (1990).

Joseph, V. R., E. Gul and S. Ba, "Maximum projection designs for computer experiments", Biometrika p. asv002 (2015).

Juel, C., "Learning to read and write: A longitudinal study of 54 children from first through fourth grades.", Journal of educational Psychology **80**, 4, 437 (1988).

Kaishev, V. K., "Optimal experimental designs for the b-spline regression", Computational Statistics & Data Analysis **8**, 1, 39–47 (1989).

Kao, M., A. Mandal, N. Lazar and J. Stufken, "Multi-objective optimal experimental designs for event-related fmri studies", NeuroImage **44**, 3, 849–856 (2009).

Kiefer, J., "Optimum experimental designs", Journal of the Royal Statistical Society. Series B (Methodological) pp. 272–319 (1959).

Kiefer, J., "Optimum designs in regression problems, ii", The Annals of Mathematical Statistics pp. 298–325 (1961).

Kiefer, J., "General equivalence theory for optimum designs (approximate theory)", The annals of Statistics pp. 849–879 (1974).

Kiefer, J. and J. Wolfowitz, "Optimum designs in regression problems", Ann. Math. Stat. **30**, 271–294 (1959).

Kubilius, J., J. Wagemans and H. P. de Beeck, "Emergence of perceptual gestalts in the human visual cortex the case of the configural-superiority effect", Psychological science p. 0956797611417000 (2011).

Laird, N. M. and J. H. Ware, "Random-effects models for longitudinal data", Biometrics pp. 963–974 (1982).

Lazar, N., *The statistical analysis of functional MRI data* (Springer, 2008).

Lesaffre, E., M. Asefa and G. Verbeke, "Assessing the goodness-of-fit of the laird and ware model– an example: the jimma infant survival differential longitudinal study", Statistics in medicine **18**, 7, 835–854 (1999).

Lindquist, M. A. *et al.*, "The statistical analysis of fmri data", Statistical Science **23**, 4, 439–464 (2008).

Liu, T. T., "Efficiency, power, and entropy in event-related fmri with multiple trial types: Part ii: design of experiments", NeuroImage **21**, 1, 401–413 (2004).

Liu, T. T. and L. R. Frank, "Efficiency, power, and entropy in event-related fmri with multiple trial types: Part i: Theory", NeuroImage **21**, 1, 387–400 (2004).

Meyer, R. and C. Nachtsheim, "Simulated annealing in the construction of exact optimal design of experiments", American Journal of Mathematical and Management Science **8**, 329–359 (1988).

Meyer, R. and C. Nachtsheim, "The coordinate-exchange algorithm for constructing exact optimal experimental designs", Technometrics **37**, 1, 60–69 (1995).

Mijović, B., M. De Vos, K. Vanderperren, B. Machilsen, S. Sunaert, S. Van Huffel and J. Wagemans, "The dynamics of contour integration: A simultaneous eeg–fmri study", NeuroImage **88**, 10–21 (2014).

Mitchell, T., "An algorithm for the construction of d-optimal experimental designs", Technometrics **16**, 2, 203–210 (1974).

Mitchell, T. and F. Miller Jr, "Use of design repair to construct designs for special linear models", Math. Div. Ann. Progr. Rept.(ORNL-4661) pp. 130–131 (1970).

Montgomery, D. C., *Design and analysis of experiments* (John Wiley & Sons, 2008).

Morris, M. D., "Physical experimental design in support of computer model development", Technometrics **57**, 1, 45–53 (2015).

Nair, V. N., W. Taam and K. Q. Ye, "Analysis of functional responses from robust design studies", Journal of Quality Technology **34**, 4, 355–370 (2002).

Nguyen, N., "An algorithm for constructing optimal resolvable incomplete block designs", Communications in Statistics - Simulation and Computation **22**, 3, 911–923 (1993).

Nguyen, N. and A. Miller, "A review of some exchange algorithms for constructing discrete d-optimal designs", Computational Statistics & Data Analysis **14**, 4, 489–498 (1992).

Pan, H. and H. Goldstein, "Multi-level repeated measures growth modelling using extended spline functions", Statistics in medicine **17**, 23, 2755–2770 (1998).

Piepel, G. F., S. K. Cooley and B. Jones, "Construction of a 21-component layered mixture experiment design using a new mixture coordinate-exchange algorithm", Quality Engineering **17**, 4, 579–594 (2005).

Pukelsheim, F., *Optimal design of experiments*, vol. 50 (siam, 1993).

Rice, J. A. and B. W. Silverman, "Estimating the mean and covariance structure nonparametrically when the data are curves", Journal of the Royal Statistical Society. Series B (Methodological) pp. 233–243 (1991).

Rodriguez, M., B. A. Jones, C. M. Borror and D. C. Montgomery, "Generating and assessing exact g-optimal designs", Journal of quality technology **42**, 1, 3–20 (2010).

Romeijn, H. E. and P. M. Pardalos, *Handbook of global optimization* (Kluwer Academic, 2002).

Royston, P. and D. G. Altman, "Regression using fractional polynomials of continuous covariates: parsimonious parametric modelling", Applied Statistics pp. 429–467 (1994).

Ruppert, D., M. P. Wand and R. J. Carroll, *Semiparametric regression*, no. 12 (Cambridge university press, 2003).

Saleh, M. and R. Pan, "A clustering-based coordinate exchange algorithm for generating g-optimal experimental designs", Journal of Statistical Computation and Simulation, Under Review (2014a).

Saleh, M. and R. Pan, "Constructing efficient experimental designs for generalized linear models", Communications in Statistics - Simulation and Computation (2014b).

Santner, T. J., B. J. Williams and W. I. Notz, *The design and analysis of computer experiments* (Springer Science & Business Media, 2013).

Silverman, B. and J. Ramsay, *Functional Data Analysis* (Springer, 2005).

Silverman, B. W., "Some aspects of the spline smoothing approach to non-parametric regression curve fitting", Journal of the Royal Statistical Society. Series B (Methodological) pp. 1–52 (1985).

Stone, M., "Cross-validation and multinomial prediction", Biometrika **61**, 3, 509–515 (1974).

Swallen, K. C., E. N. Reither, S. A. Haas and A. M. Meier, "Overweight, obesity, and health-related quality of life among adolescents: the national longitudinal study of adolescent health", Pediatrics **115**, 2, 340–347 (2005).

Tsui, K., "Modeling and analysis of dynamic robust design experiments", IIE Transactions **31**, 12, 1113–1122 (1999).

Vahl, C. I. and G. A. Milliken, "Whole-plot exchange algorithms for constructing d-optimal multistratum designs", Communications in Statistics - Simulation and Computation **40**, 7, 1030–1042 (2011).

Verbeke, G. and G. Molenberghs, *Linear mixed models for longitudinal data* (Springer, 2009a).

Verbeke, G. and G. Molenberghs, *Linear mixed models for longitudinal data* (Springer, 2009b).

Wager, T. D. and T. E. Nichols, "Optimization of experimental design in fmri: a general framework using a genetic algorithm", Neuroimage **18**, 2, 293–309 (2003).

Wald, A., "Tests of statistical hypotheses concerning several parameters when the number of observations is large", Transactions of the American Mathematical Society **54**, 3, 426–482 (1943).

Welch, W. J., "Computer-aided design of experiments for response estimation", Technometrics **26**, 3, 217–224 (1984).

West, M., P. J. Harrison and H. S. Migon, "Dynamic generalized linear models and bayesian forecasting", Journal of the American Statistical Association **80**, 389, 73–83 (1985).

Woods, D., S. Lewis and J. Dewynne, "Designing experiments for multi-variable b-spline models", Sankhyā: The Indian Journal of Statistics pp. 660–677 (2003).

Wu, C. and M. S. Hamada, *Experiments: planning, analysis, and optimization*, vol. 552 (John Wiley & Sons, 2011a).

Wu, C. J. and M. S. Hamada, *Experiments: planning, analysis, and optimization*, vol. 552 (John Wiley & Sons, 2011b).

Wynn, H., "The sequential generation of d-optimum experimental designs", The Annals of Mathematical Statistics **41**, 5, 1655–1664 (1970).

Zahran, A., C. M. Anderson-Cook and R. H. Myers, "Fraction of design space to assess prediction capability of response surface designs", Journal of Quality Technology **35**, 4, 377 (2003).

Zhou, H. and A. B. Lawson, "Ewma smoothing and bayesian spatial modeling for health surveillance", Statistics in medicine **27**, 28, 5907–5928 (2008).