SIMULATION OPTIMIZATION USING OPTIMAL COMPUTING BUDGET ALLOCATION

XIAO HUI

(B. Eng. (Hons.), NUS)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2013

Declaration

I hereby declare that the thesis is my original work and it has been written by me its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Xiao Hui

Xiao Hui

20 June 2013

Acknowledgments

I would like to thank my supervisor Associate Professor Ng Kien Ming for his patient guidance throughout my PhD journey. In addition, I am grateful for the guidance and advice given by Associate Professor Lee Loo Hay. I would also like to thank my Oral Qualifying Examiners, Associate Professor Ng Szu Hui and Assistant Professor Chen Nan for their valuable comments and suggestions during the proposal of the thesis.

I would also like to express my gratitude to the fellow students in our department. I have learned a lot from discussion and collaboration with them, especially those working in simulation optimization, Nugroho Artadi Pujowidianto, Zhang Si and Li Haobin.

Finally, I would like to thank my family for their continuous support and encouragement.

Table of Contents

Summary

Previous research in ranking and selection focused on selecting the best design and subset selection. Little research has been done for ranking all designs completely. In the first part of the thesis, we consider the problem of ranking all designs completely where the performance of each design can only be estimated with noise via simulation. Simulation is time consuming and thus simulation budget needs to be allocated efficiently. We propose efficient simulation procedures to optimally allocate simulation replications with the objective of maximizing the probability of correctly ranking all designs completely. Compared with the previous indifference zone allocation strategy, our proposed allocation rule performs the best under different scenarios. The second part of the thesis extends the idea of complete ranking to rank top *m* designs out of *k* total alternatives, where *m* can be any value from 1 to *k*. It is motivated by the idea of integrating ranking procedures into evolutionary algorithms in a noisy environment where the fitness value of candidate solution can only be evaluated through simulation. Using optimal computing budget allocation (OCBA) framework, we formulate this problem as that of maximizing the probability of correctly ranking the top *m* designs subject to the constraint of a fixed simulation budget *n*. Based on large deviation theory, we have derived the asymptotically optimal allocation rule. The proposed simulation budget allocation rule is integrated with the genetic algorithm to solve simulation optimization problems. Numerical experiments have shown that a significant number of simulation replications could be saved by integrating our proposed allocation procedure. The last part of this thesis considers the simulation budget allocation when the simulation output can be modeled by quadratic equations. The entire domain is divided into many partitions and the simulation output of each partition is modeled as a quadratic regression line. We formulate an optimization model to determine the asymptotically optimal simulation budget allocation for the problem. The optimization model is nonlinear and highly complex; therefore, we analyze the limiting allocation rule when the number of partitions goes to infinity in order to obtain an easily implementable budget allocation. The resulting simulation budget allocation rule matches our intuition, and the cross partition allocation rule is similar to the original OCBA rule. In addition, the allocation rule within partition is simply the optimal simulation design for the best partition, which contains the best design location and a feasibility check problem with quadratic regression for other partitions. The effectiveness of our proposed allocation rule is shown through several numerical experiments.

List of Tables

List of Figures

List of Symbols

The following are some selected notations,

k	=	number of designs
i	=	index for designs, $i = 1,...,k$
$\bar{X}_i(\bullet)$	=	sample mean performance of design i
μ_i	=	mean performance of design i
$P(CR)$	=	probability of correct ranking
α_i	=	fraction of budget allocated to design i , $i = 1,...,k$
$P(FR)$	=	probability of false ranking
$G_i(\alpha_i, \alpha_{i+1})$	=	convergent rate function of false ranking probability of design i , $i+1$
$I(\bullet)$	=	Fenchel-Legendre transform
$\Lambda(\bullet)$	=	log-moment generating function
$P(CR_m)$	=	probability of correct ranking top *m* designs
$P(FR_m)$	=	probability of false ranking top *m* designs
m	=	number of partitions
h	=	index for partitions, $h = 1,...m$
$y(x_{hi})$	=	performance at design location i of partition h
$f(x_{hi})$	=	simulated performance at design location i , partition h
ε_h	=	simulation noise for partition h
$\mathbf{W_h}$	=	coefficients of quadratic equations for partition h

$\hat{\mathbf{W}}_{\mathbf{h}}$ = estimated coefficients of quadratic equations for partition h

$\hat{y}(x_{hi})$ = the estimated performance at design location i of partition h

$R_{Bb,hi}(\bullet)$ = the convergent rate function of false selection probability for design location hi

β_h = the fraction of total budget allocated to partition h

$\alpha_{h1}, \alpha_{hs}, \alpha_{hk}$ = the fraction of total budget allocated to design location $1, s, k$ within partition h

$\tilde{R}_{Bb,hi}(\bullet)$ = approximation of rate function $R_{Bb,hi}(\bullet)$ when the number of partition goes to infinity

$\mathbf{F_h}$ = a vector with n entries of simulation output $f(x_{hi})$

$\mathbf{X_h}$ = a $n \times 3$ matrix with each row $[1 \;\; x_{hi} \;\; x_{hi}^2]$ corresponding to $\mathbf{F_h}$

$\mathbf{X_h^t X_h}$ = information matrix

δ_{hi}^2 = squared performance difference between design location hi and the best design location

List of Abbreviations

OCBA	=	Optimal Computing Budget Allocation
PCS	=	Probability of Correct Selection
PCR	=	Probability of Correct Ranking
IZ	=	Indifference Zone
R&S	=	Ranking and Selection
OSD	=	Optimal Simulation Design
POSD	=	Optimal Simulation Design in Partitioned Domain
EA	=	Equal Allocation
DOPT	=	D-optimal
DES	=	Discrete Event Simulation

Chapter 1. Introduction

1.1. Background

With the increasing complexity of modern industrial systems, it becomes more difficult to have an analytic model to evaluate the system while satisfying all the assumptions. As a result, discrete event simulation (DES) has been widely used to evaluate the performance of complex systems. Simulation has many advantages such as incorporating new information into the system without disrupting the on-going operations, compressing or expanding time scales for speed-up or slow-down investigation, performing bottleneck analysis and sensitivity analysis. Among all the advantages, the most significant advantage lies in the fact that fewer assumptions are needed in simulation models compared with analytical models, and therefore, simulation models are closer to the practical situation. While simulation has been successfully applied in many areas such as semiconductor manufacturing, construction engineering, project management, logistics and supply chain, the concern on the efficiency of simulation has never stopped, particularly when the simulation cost is very expensive (Law and Kelton, 1991).

The performance of a simulation model can be mathematically represented as $L(x(t,\theta,\xi))$, where $x(t,\theta,\xi)$ is the sample path evolving through time, θ is the design variable and ξ is the randomness involved in the simulation. Because of the inevitable randomness within the simulation, we could only estimate the performance by its expected value as $L(\theta) = E(L(x(t,\theta,\xi))) \approx (1/n)\sum_{i=1}^{n} L(x(t,\theta,\xi_i))$. Therefore, in order to have a statistically significant steady mean performance value, a large number of simulation replications are needed and this sample mean estimator cannot converge faster than $O(1/\sqrt{N})$, where N is the number of replications of simulation (Chen, 2002; Ho et al., 2007; Chen and Lee, 2010). On the other hand, the cost of per simulation run increases with the increasing complexity of the system. Therefore, the total simulation cost will be extremely high when evaluating a highly complex system with the requirement of high accuracy. For example, it will cost about 36 to 160 hours of computation for Ford Motor Company to conduct crash simulation of a full passenger car (Gu, 2001).

To improve the efficiency of simulation, various simulation optimization methods have been proposed. Simulation optimization is the process of finding the best design where the performance of the design can only be estimated via simulation. Based on whether the decision variables are continuous or discrete, simulation optimization can be classified into continuous simulation optimization and discrete simulation optimization. Statistical ranking and selection (R&S) method is commonly used when the discrete decision variables are discrete, fixed and finite. As reported in Branke et al. (2007), optimal computing budget allocation (OCBA) is one of the top three R&S procedures in the context of simulation. OCBA aims to determine the optimal allocation rule of simulation replications in order to compare a finite number of simulated alternatives. A comprehensive review of the recent development of OCBA problems can be found in Lee et al. (2010) and Zhang et al. (2013).

In this thesis, we consider three new types of OCBA problems. The first problem is to determine how to allocate the simulation replications efficiently in order to rank a finite number of alternatives completely. We further extend complete ranking to top *m* ranking which only tries to rank the top *m* designs out of a finite number of alternatives. The last problem considered in this thesis is to select the best design location when quadratic equations are used to model the simulation output.

Compared with selecting the single best alternative, top *m* ranking and complete ranking require more simulation budget because more information would be needed in order to rank the alternatives. In practical situations, the ranking information could help the decision maker to select the most appropriate design when selection of the design is also subject to other qualitative constraints. For example, when several lift systems are available for selection for a multi-storey warehouse, we could rank the lift systems quantitatively based on the cost or efficiency. However, other qualitative requirements such as space utility, safety issue and environmental issue also need to be considered. Decision makers can use the ranking information and make the tradeoff among the designs based on the qualitative requirements. In addition, ranking information could also be incorporated into population-based search algorithms to enhance the search efficiency. For example, in the selection step of a genetic algorithm, the better candidate is usually given a higher probability to be selected as the parents to produce the offspring. Therefore, the ranking of the solutions is important in determining the search direction of GA. It is straightforward to identify the ranking of the solutions in a deterministic scenario, but this is difficult and costly in a stochastic environment. Therefore, we can use the proposed budget allocation rule to minimize the number of samples to simulate while achieving the same probability of correctly ranking the elite solutions. Other examples using ranking information can be found in maritime safety assessment (Dourmas et al., 2008), evolutionary algorithms (Schmidt et al., 2007), data envelopment analysis (Alirezaee and Afsharian, 2007).

When the total number of alternatives is large, selecting the best alternative also requires extremely large simulation budget even though OCBA (Chen et al., 2000) reduces the simulation replications significantly. However, if certain performance structure across the domain can be explored, we could further enhance the simulation efficiency by reducing the number of design points to be simulated. For example, the proposed approach can be well integrated with polynomial regression where an underlying regression function is assumed across the domain.

In this thesis, we attempt to study the three problems discussed above to determine the most efficient way of allocating the simulation budget so as to maximize the probability of correct ranking or selection given a fixed number of simulation budget. We will also apply our budget allocation rule to population-based search algorithms to enhance the search efficiency in stochastic environments.

1.2. Motivation

The research in this thesis is motivated by the fact that previous research had all focused on selecting a single best or a best subset from all the alternatives and little research has been done to rank the alternatives. Many simulation budget allocation procedures have been proposed to select the best alternative with a fixed limited computing budget. Three typical procedures include optimal computing budget allocation (OCBA) (Chen et al., 2000), indifference zone (IZ) formulation (Kim and Nelson, 2006) and value information procedure (VIP) (Chick and Inouem, 2001). The problem of selecting an optimal subset from a finite number of alternatives is also well studied in the literature (Koenig and Law, 1985; Dudewicz and Dlal, 1975; Chen et al., 2008; Zhang et al., 2012). However, the only substantial work addressing the problem of ranking alternatives that we are aware of is based on indifference zone formulation in Bishop (1978) and Beirlant et al. (1982). The indifference zone formulation aims to find a feasible way to guarantee that the pre-specified probability of correct ranking is achieved. Although OCBA has been shown to be effective in selecting the single best alternative (Chen et al., 2000) and an optimal subset (Chen et al., 2008; Zhang et al., 2012), none of the previous research works has used OCBA for ranking the alternatives. This motivates us to use the OCBA framework to further enhance the simulation efficiency for ranking problems. As different problem settings may need different information about ranking, it is important to consider both complete ranking and top *m* ranking. The proposed allocation rule has the potential to enhance the search efficiency for population-based search algorithms.

Although OCBA has proven to be very efficient in selecting the best alternative, the simulation budget also increases rapidly when the number of alternatives becomes large. Little research has been done to explore the underlying performance structure to further reduce the simulation budget. Brantley et al. (2013a; 2013b) proposed optimal simulation design procedure (OSD) and optimal simulation design procedure in partitioned domain (POSD) which incorporated the simulation output into regression equations, and successfully reduced the simulation budget significantly when the underlying performance structure is quadratic or approximately quadratic. However, the approach used in Brantley et al. (2013a; 2013b) to solve the problem is heuristic and non-optimal. This motivates us to use large deviation theory to derive an optimal allocation rule when regression analysis is integrated to model the simulation output. In addition, the complexity of the existing allocation rule inspires us to derive an easily implementable closed-form allocation rule.

1.3. Objective

The objective of our research is to enhance the simulation efficiency by intelligently controlling the allocation of simulation replications so as to maximize the probability of correct ranking or selection with a fixed limited number of simulation replications. The simulation budget allocation problems considered in this thesis include complete ranking of all alternatives, ranking of top *m* designs out of *k* alternatives and selecting the best design when the underlying performance can be modeled by quadratic regression equations. We formulate the problems as nonlinear optimization models using large deviation theory and aim to derive the respective simulation budget allocation rules that are easy to implement.

The effectiveness of the proposed allocation rules will be demonstrated in simulation experiments by comparing the number of simulation runs needed for our proposed rules with that of other existing rules. In addition, we also want to show how our proposed simulation budget allocation rule can be integrated with evolutionary algorithms to enhance the simulation efficiency for simulation optimization problems.

1.4. Scope

Our research concentrates on black-box simulation optimization, and belongs to a part of discrete simulation optimization. We do not consider how to speed up the process of an individual simulation run. Instead, our scope of study is to find a way to reduce the number of simulation runs. We aim to derive the optimal simulation budget allocation procedure to find the best design or the ranking of the designs based on the mean performance of each design when their performance can only be estimated with noise via simulation, where the number of designs in our study is finite and fixed.

1.5. Contribution

The contributions of this thesis are listed as follows:

- We extended the OCBA framework to address the problem of ranking all alternatives completely. Based on large deviation theory, we derived the asymptotically optimal allocation rule and approximated allocation rule to maximize the probability of correct ranking. Practitioners who are more concerned about optimality can use a nonlinear programming solver to obtain the asymptotically optimal rule. The approximated allocation rule is an easily implementable closed-form solution which can be more useful for users of our allocation rule in finite budget with less effort spent on obtaining the allocation rule.

- We further considered the problem of ranking the top *m* designs out of *k* total alternatives. When *m* is equal to *k*, the top *m* ranking problem becomes the complete ranking. It also reduces to the original OCBA problem if *m* is equal to 1. Therefore, it can be thought of as a generalization of the complete ranking problem and OCBA problem for selecting the single best design.

- The proposed allocation procedure for ranking is integrated with genetic algorithm to reduce the number of samples needed for genetic algorithm in noisy environments. The numerical experiments indicate that significant simulation replications are saved by using the proposed budget allocation rule.

- We formulated the optimal simulation budget allocation problem when the design locations can be divided into various partitions and the performance at every design location in each partition can be modeled as a quadratic regression line. Relying on the large deviation theory, we developed efficient simulation budget allocation rule to select the best design point from all design locations in all partitions.

- We further analyzed the asymptotical behavior of the allocation rule when the number of partitions goes to infinity and derived the limiting asymptotically optimal allocation rule. Important insights on the allocation rule have been drawn based our derivation. The allocation rule cross partitions matches our intuition and follows similarly with the original OCBA rule. In addition, the allocation rule within partition is simply OSD for the best partition which contained the best design location and a feasibility check problem with quadratic regression for other partitions.

1.6. Organization of the Thesis

The thesis is organized as follows. Chapter 2 provides a comprehensive literature review of related works and Chapter 3 provides the formulation of complete ranking problem and derives the allocation rules. Chapter 4 studies the top *m* ranking problem and applied the allocation procedure to genetic algorithms to enhance the search efficiency. Chapter 5 studies the simulation budget allocation problem when quadratic regression functions are used to model the simulation output. Finally, we conclude the thesis in Chapter 6 and discuss some possible future research directions.

Chapter 2. Literature Review

We provide a comprehensive literature review that is related to our research in this section. Section 2.1 summarizes the existing research in simulation optimization. Section 2.2 reviews the development and extensions of the optimal computing budget allocation problem and their applications to various problems such as inventory control, production scheduling and search algorithms. A review on the optimal design of experiment that is related to the research work in Chapter 5 is presented in Section 2.3. Finally, we summarize the research gaps in Section 2.4.

2.1. Simulation Optimization

Simulation is the process of modeling the real-world operational process or systems over time. It can capture the key characteristics and behaviors of the selected system without making any assumption. Optimization is the process of finding the best solution based on certain criteria subject to some constraints. Simulation optimization is the process of selecting the best solution when the performance of each solution can only be estimated with noise via simulation. It is also commonly called optimization via simulation. In general, simulation optimization can be categorized into two groups based on whether the decision variable is discrete or continuous (Ólafsson and Kim, 2002).

Continuous simulation optimization deals with the problems where the decision variables are continuous, i.e., uncountable and infinite. This is probably the most well studied area, and it can be traced back to 1950s, when stochastic approximation (SA) was first proposed by Robbins and Monro (1951) and Kiefer and Wolfowitz (1952). SA has been extended by Kushner and Yin (2003) and Borkar (2008). SA is an iterative process of moving from one solution to anther based on the estimation of the gradient. It is similar to the steepest descent gradient search method in nonlinear optimization problem. The difference is that we do not have a closed-form expression for the objective function. The challenge of using SA is in estimating the gradient in the midst of the noise from the uncertainties. The simplest way to estimate the gradient is to use the finite difference method. The one-side estimation of finite difference requires $n+1$ simulation experiments and the two-sided estimation needs $2n$ simulation experiments. Therefore, considerable computational effort would be spent on using the finite different method. A more efficient way of estimating gradient will be simultaneous perturbation stochastic approximation (SPSA, Spall, 1999), which requires only two measurements of the objective function regardless of the dimension of the optimization problem. Both the finite difference method and SPSA treat the simulation process as a black box. No knowledge of the underlying simulation mechanics is known and no change is made in execution of the simulation model. Thus, they are referred to as indirect estimation methods. To further improve the computational efficiency and convergence properties of SA, other methods utilize the information about the simulation setting such as the distribution in generating the random variables to estimate the gradient. They are referred to as direct gradient estimation methods. These methods include perturbation analysis (Glasserman, 1991; Ho and Cao, 1991; Fu and Hu, 1997) and likelihood ratio (Glynn, 1989; Rubinstein and Shapiro, 1993). Compared with the indirect gradient estimation method, the direct gradient estimation method usually provides an unbiased estimator which leads to faster convergence rate and the resulting estimator is usually more efficient computationally. A more detailed summary of different gradient estimation methods can be found in Fu (2006 and 2008). Although SA receives the most attention in the literature, there are also some other alternative methods suggested by various researchers, such as the sample path method proposed by Gurkan et al. (1994). The basic idea of this method is to fix a sample path first, and use the deterministic optimization methods to find the optimal solution. It then moves to another sample path. By doing so, it moves towards the optimal solution iteratively. This method has been shown to converge almost surely by Robinson (1996). Another popular approach is to apply the response surface methodology (RSM) to simulation optimization, which aims to find a functional relationship between the input and output of the simulation. An example of such a method can be found in (Kleijnen, 2008).

When the decision variables are discrete or countably finite, the methods for continuous simulation optimization typically do not apply since the gradient cannot be obtained. Furthermore, it is impractical to assume the discrete domain to be continuous. For example, we want to find the most reliable design out of a few alternative designs. It is not meaningful to have a fractional number to be the optimal solution. We can generally summarize the methods for solving the discrete simulation optimization problems into two categories. The first category aims to solve the simulation optimization when the number of decision variables is finite and small. It is typically called ranking and selection. Examples of this can be traced back to as early as the

1950s, during which indifference zone (IZ) formulation was first established (Bechhofer, 1954). Gupta (1956) formulated the first subset selection problem in the area of ranking and selection. In the context of simulation, there are three major approaches to enhance the efficiency of selecting the best designs. The optimal computing budget allocation (OCBA) proposed by Chen et al. (2000) focuses on the efficiency of simulation by intelligently allocating further replications based on the mean and variance. A more detailed literature review on the OCBA is provided in Section 2.3 below. The IZ procedure focuses on finding a feasible way to guarantee that the pre-specified probability of correct selection is achieved. A typical example of such a procedure is the fully sequential two stage allocation KN++ procedure proposed by Kim and Nelson (2006). Lastly, Chick and Inoue (2001) use the Bayesian posterior distribution to describe the evidence of correct selection, and allocate further replications based on maximizing the value information. Recently, Branke et al. (2007) compared the three procedures in more detail based on their efficiency, controllability, robustness and sensitivity.

When it is not possible to evaluate every solution using the ranking and selection method, some other methods must be considered for finding the optimal solution. Numerous methods have been developed in the literature for this purpose. The simplest method will be the random search which involves an iterative process to search for a better solution in the neighborhood of the current solution. The difference of various random search methods in the literature lies in the way of specifying the neighborhood structure, the way of selecting a candidate solution, and the way of defining acceptance criterion and stopping criterion. In addition to random search, some metaheuristic methods have also been applied in simulation optimization. These metaheuristics include simulated annealing (SA) (Haddock and Mittenhal, 1992), Tabu search (Glover and Laguna, 1997), genetic algorithm and Nested partition (NP) method (Shi and Ólafsson, 1997). The main challenge of using these metaheuristics is on how to adapt these deterministic optimization techniques to noisy simulation environments with the objective of increasing the efficiency. A more detailed summary on metaheuristics can be found in (Glover and Kochenberger, 2003) and Gendreau and Potvin (2010). Other than the global optimization method, some methods are guaranteed to find the local optimal solution efficiently for practical consideration. An example of such a method is the COMPASS algorithm proposed by Hong and Nelson (2006) and Hong et al. (2010).

2.2. Optimal Computing Budget Allocation

Optimal computing budget allocation was first proposed by Chen et al. (2000). It focuses on the efficiency of simulation by intelligently allocating further replications based on the mean and variance. It has been shown that the speed-up factor of OCBA is beyond exponential rate. The OCBA problem is formulated as that of maximizing the probability of correct selection (PCS) given a fixed number of computing budget. The resulting allocation rule matches the intuition that more simulation replications should be allocated to those designs that are critical in identifying the ordinal relationship in order to obtain a high probability of correct selection. OCBA was shown to be one of the top performing methods in the work done by Branke et al. (2007) and Waeber et al. (2010).

The original OCBA is an unconstrained single objective problem aiming to select the best design from all alternatives, i.e., the performance of each design is only measured in one dimension. The OCBA framework has been extended to solve simulation optimization problems in different settings. The first category, which is known as the OCBA-m problem, aims to select an optimal subset from all designs. Chen et al. (2008) has first proposed the allocation rule for selecting the optimal subset. The allocation rule has been incorporated into evolutionary algorithms such as cross entropy, population-based incremental learning and neighborhood random search to enhance the search efficiency in simulation optimization. Zhang et al. (2012) developed an allocation rule called OCBA-m+, which has been shown to perform better than the OCBA-m rule proposed by Chen et.al (2008). When the problem of selecting the best design is subject to stochastic constraints, this type of simulation optimization is known as the OCBA-CO, which is first studied by Pujowidianto et al. (2009). The finite domain is partitioned into four subsets: the set of unique best feasible systems, the set of suboptimal feasible systems, the set of infeasible systems with better objective value than the best feasible system and the set of infeasible systems with worse objective value than the best feasible system. By intelligently controlling the allocation rule, the best feasible design is selected with the highest probability of correct selection under fixed limited computing budget. In this case, the performance of every design is assumed to be normally distributed. Further improvement and extensions of OCBA-CO problem include Lee et al. (2012), Pujowidianto et al. (2012; 2013) and Hunter et al. (2011). OCBA is also extended to solve multi-objective simulation optimization problems, i.e., the performance of each design is measured by more than one dimensions. This type of problem is known as the MOCBA problem, which was studied by Lee et.al (2004) and Lee et al. (2010). The aim of the MOCBA is to select the non-dominated designs rather than the single best design. Further research on MOCBA has also incorporated the indifference zone into the multi-objective computation budget allocation problem (Teng et al., 2010).

In addition, the OCBA framework has been extended in various other ways. Glynn and Juneja (2004) studied the simulation budget allocation when the performance of the designs follows a general distribution, i.e., it removes the assumption that the performance of each design is normally distributed. He et al. (2007) derived the budget allocation rule using the opportunity cost as the performance measure instead of the correct selection probability. Fu et al. (2007) considered the optimal budget allocation when the system performances are sampled in the presence of correlation. A more generalized version of correlated sampling can be found in Peng et al. (2013). Jia et al. (2013b) formulated a new version of OCBA in order to find the simplest good designs and the asymptotically optimal allocation rule which has been shown to be effective. Jia (2013a) futher quantified the relationship between simulation time and the performance estimation accuracy, which generalized the OCBA rule when the simulation time is stochastic. Some other research works related with OCBA include using the OCBA framework for rare event simulation (Shortle et al., 2012), deriving the adaptive sampling algorithm for simulation-based optimization with descriptive complexity preference (Jia, 2011), and the work is generalized in Yan et al. (2012), and OCBA for discrete event simulation experiments (Chen et al., 2010).

Concurrent with the theoretical development and extension of OCBA, a large number of research papers discussed the applications of the OCBA framework to various search algorithms and some practical simulation problems. Examples of applying the OCBA framework to search algorithms can be summarized as follows. He et al. (2008) used the OCBA framework in simulation optimization using the cross entropy method. Chen et al. (2013) incorporated OCBA into the partitioned base random search algorithm. Other examples include nested partition search with OCBA (Chew et al., 2009), multi-objective evolutionary algorithm (Lee et al., 2008), population-based incremental leaning algorithm and neighborhood random search algorithm (Chen et al., 2008), and particle swarm optimization (Zhang et al., 2011). OCBA can also be applied in many practical simulation problems. For example, OCBA has been applied to the manufacturing scheduling of a semiconductor factory, where the computation efficiency is one of the major challenges (Hsieh et al., 2001; 2007). Trailovic and Pao (2004) applied the OCBA framework to the target tracking algorithm aiming to find the best design with the minimum variance. Lee et al. (2007; 2008) and Chew et al. (2009) integrated multi-objective OCBA framework with a search algorthm to solve flight scheduing and invertory management problems. Other examples include using OCBA to improve the energy management in commercial office building (Jia et al., 2012), minimizing the processing cost for top *k* queries (Farley et al., 2012) and data envelopment analysis (Wong et al., 2011).

In summary, the research in the area of discrete simulation optimization in finite search space can be categorized as follows. Firstly, there is a large amount of literature on selecting the best system, where OCBA, VIP and IZ are the three major approaches towards this problem. Secondly, OCBA has been extended to subset selection, constrained optimization and multi-objective simulation optimization problems. However, no previous research has used the OCBA framework to attempt the problem of ranking the alternatives. Lastly, many extension works and application papers exist in the literature, but little research has incorporated the response surface methodology with OCBA.

2.3. Optimal Design of Experiment

Design of experiment (DOE) is commonly used for gathering information when variation is present. DOE can be categorized into different categories based on different criteria (Melas, 2006), but the problem considered in our research is most related to that of the optimal design of experiments. Optimal design is a class of experiment design that determines the design locations and allocates samples in order to optimize the experiment process with respect to certain statistical criterion. It allows the parameters estimated to be unbiased and with minimum variance. On the other hand, the non-optimal designs will need more experimental runs in order to reach the same precision of estimation. In general, optimal design reduces the number of experiment runs, and thus the experiment cost. Barton (2005) discussed various optimal criteria for regression models in estimating simulation output. For example, A-optimality seeks to minimize the trace of the inverse of the information matrix. C-optimality aims to minimize the variance of a best linear unbiased estimator. D-optimality minimizes the determinant of the information matrix and G-optimality maximizes the maximal entry in the diagonal of the hat matrix. Cheng and Kleijnen (1999) first applied the DOE method to simulation optimization and developed a criterion to provide the best fitting regression equation for queuing models.

Unlike the traditional optimal experiment design, the optimal simulation design (OSD) method proposed by Brantley et al. (2013a) aims to maximize the probability of correctly selecting the best design location in a set of pre-determined design locations. Unlike all previous OCBA problems that require conducting simulation at all design locations, OSD only needs to conduct simulation at a subset of the alternative design locations. The simulation output across the domain is then modeled by a quadratic regression function. The performance value at each design location is estimated from the regression function. The objective of the OSD method is to determine which design locations should be selected for simulation and how the simulation samples should be allocated among the selected design locations. OSD has been shown to be very efficient when the underlying performance structure of all design points is quadratic and approximately quadratic. However, OSD assumes a common quadratic equation for all design locations and common noise across the entire domain. It is natural to think that the two assumptions used in OSD may not hold when the number of design locations is large. One way to resolve the problem is to divide the entire domain into many partitions, and assume a quadratic equation and different common noise for each partition. This problem is studied in Brantley et al. (2013b) where a simulation budget allocation problem is formulated for the scenario when the domain is partitioned into various sub-regions. The resulting simulation budget allocation rule is named as POSD. However, the POSD method proposed in

Brantley et al. (2013b) approximated the correct selection probability by a convenient lower bound. The simulation budget allocation is derived via analyzing the different scenarios of the approximated probability of correct selection. Therefore, the heuristic allocation rule is suboptimal. As mentioned in Brantley et al. (2013b), better allocation rules can be derived using optimal formulation or tighter probability bound.

2.4. Summary of Research Gaps

Based on the literature survey, some research gaps can be identified in the area of optimal computing budget allocation. Firstly, no previous research has used the OCBA framework to determine the optimal simulation budget allocation for the problem of ranking all alternatives completely. The only work we found is based on inference-zone formulation (Bishop, 1978; Beirlant et al., 1982) which tries to find a feasible solution to achieve a pre-specified probability of correct ranking. The procedure is conservative and inefficient. Secondly, a more general problem than complete ranking is to rank the top *m* designs out of *k* alternatives. The top *m* ranking problem becomes complete ranking if *m* is equal to *k*, and it can be reduced to the original OCBA problem of selecting a single best design if *m* is equal to 1. In addition, top *m* ranking can be applied to population-based search algorithms in noisy environment to enhance the search efficiency by reducing the number of simulation replications needed for performance evaluation. Lastly, little research has been done to determine the simulation budget allocation when response surface methodology is used to model the simulation output, in particular when quadratic regression functions are used to model the simulation output. The work done by Brantley et al. (2013a; 2013b) is heuristic in nature and non-optimal. We can reformulate the problem using large deviation theory to characterize the optimal allocation strategy, and to analyze the asymptotical behaviors of the allocation rule.

Given the research motivations and research gaps summarized above, we will study two categories of OCBA problems in this thesis, i.e., the ranking-based OCBA and regression-based OCBA. Ranking-based OCBA aims to develop efficient simulation budget allocation rule for ranking all alternatives completely, and selecting and ranking top *m* designs simultaneously. Regression-based OCBA tries to determine the simulation budget allocation when the simulation output can be modeled by a quadratic regression line in each partitioned domain. Table 2.1 below summarizes the major existing research works and identifies the research problems that will be studied in this thesis.

Table 2.1. Summary of existing works and research gaps

	Existing Works	Problem Considered
Problem Setting	OCBA1: Select the single best	Complete ranking: Rank all alternatives completely
	OCBA-m/OCBA-m+ : Select top *m* designs	
	MOCBA: Select a Pareto set	Top *m* ranking: Simultaneously select and rank top *m* designs
	OCBA-CO: Select the single best subject to constraints	
Output Modeling	Sample mean	Quadratic regression line

Chapter 3. Simulation Budget Allocation for Complete Ranking

We present the first research problem of ranking all designs completely in this chapter. The simulation budget allocation rules derived asymptotically maximize the probability of correct ranking. The organization of this chapter is as follows. Section 3.1 provides an overview of the whole chapter. In Section 3.2, we provide the formulation of the complete ranking problem. Section 3.3 states the necessary assumptions needed in deriving the optimal allocation rule. In Section 3.4, we provide the derivation of asymptotically optimal allocation rule. An easily implementable closed-form approximated allocation rule is presented in Section 3.5. We propose a heuristic sequential allocation algorithm to implement our allocation rules, and prove that the estimators are consistent in Sections 3.6 and 3.7 respectively. Numerical experiments are conducted in Section 3.8. Finally, we conclude in Section 3.9. The work of Chapter 3 has been published in Xiao et al. (2013).

3.1. Overview

Previous research in ranking and selection focused on selecting the best design and subset selection. Little research has been done for ranking all designs completely. In deterministic analysis, it is straightforward to identify the ranking of the solutions based on their performance values. However, in stochastic environment, a large number of simulation runs or experiments are needed in order to obtain a steady mean performance value. The ranking information is usually needed when secondary or other performance measurements are considered. For example, when several lift systems are available for selection for a multi-storey warehouse, we could rank the lift systems quantitatively based on the cost or efficiency. However, other qualitative requirements such as space utility, safety issue and environmental issue also need to be considered. Decision makers can use the complete ranking information and make the tradeoff among the designs based on the qualitative requirements. Some other examples of using ranking information in the literature are as follows. In reliability and life testing, experiments are conducted in order to rank several population items. The difficulty is to decide how many devices of each type should be tested (Bishop and Dudewicz, 1977). Zhao et al. (2004) proposed using a simulation approach for ranking of fire safety attributes of existing buildings. The complete ranking approach can be applied in their study in order to reduce the number of simulation runs. In data envelopment analysis (DEA), much work has been devoted to find the complete ranking of the decision-making units (DMU). As demonstrated in Wong et al. (2011), efficient budget allocation will reduce the data needed to estimate the performance when the efficiency measurement is in a stochastic environment. In addition, the complete ranking procedure could also be incorporated into population-based searching algorithm to enhance the search efficiency. For example, in a genetic algorithm (GA), better candidates are usually given higher probability to be selected as the parents to produce the offspring. Therefore, the ranking of the solutions is important in determining the search direction of GA. Examples of using ranking information in evolutionary algorithms can be found in (Blickle and Thiele, 1995;Schmidt et al., 2007). In the literature, the complete ranking problem has been approached with the IZ formulation and a two-stage allocation algorithm is proposed to guarantee that the pre-specified correct ranking probability is achieved (Bishop, 1978; Beirlant et al., 1982). This result has been applied to decide the number of simulation replications for design of experiment (Bishop, 1978) and to rank the random number generators (Levendovszky et al., 1996).

The IZ formulation of the complete ranking problem aims to find a feasible way to achieve the pre-specified probability of correct ranking, and the efficiency can be improved. In this chapter, we will formulate the complete ranking as an optimal computing budget allocation problem and derive an efficient budget allocation procedure to maximize the probability of correctly ranking all designs with a fixed limited computing budget. We will first formulate the correct ranking probability directly and derive its asymptotical optimal allocation rule based on the formulation. An approximated formulation of the correct ranking probability based on a lower bound is also presented, and an approximated allocation rule is derived based on this formulation. We will compare the asymptotical optimal allocation rule, approximated allocation rule, the existing two-stage IZ rule and equal allocation in the numerical experiments. The numerical comparison shows that our approximated allocation rule outperforms the other rules in terms of correct ranking probability in finite budget.

3.2. Problem Formulation

Consider the problem of ranking k designs according to their performance values. The performance value can only be estimated with noise via simulation. The mean performance is used as the measurement for comparison. In order to have a statistically significant steady estimation, a large number of simulation replications are needed because of the inherent uncertainty within simulation. Given that there is a total of *n* simulation replications available, our objective is to find the best allocation strategy such that we could rank *k* designs as correctly as possible based on the mean performance value estimated from the simulation output.

Without loss of generality, we assume that the mean performance values of designs $1,...,k$ are $\mu_1,...,\mu_k$ respectively with $\mu_1 < \mu_2 < ... < \mu_k$. We assume that there exists $\delta > 0$ such that $\mu_{i+1} - \mu_i > \delta, \forall i = 1,...,k-1$. In other words, designs with performance value difference smaller than δ are not considered for comparison in our research. Let $\boldsymbol{\alpha} = (\alpha_1,...,\alpha_k)$ represent the proportion of total budget to be allocated to each design with $\sum_{i=1}^{k} \alpha_i = 1$. Let $\bar{X}_i(\alpha_i n) = (\alpha_i n)^{-1} \sum_{j=1}^{\alpha_i n} X_{ij}$ denote the sample mean performance of design i, where $(X_{i1},...,X_{i,\alpha_i n})$ are the samples generated from design i. Our objective is to find the optimal allocation strategy $\boldsymbol{\alpha}^* = (\alpha_1^*,...,\alpha_k^*)$ such that the probability of correctly ranking the *k* designs can be maximized with fixed limited computing budget *n*.

Under the assumptions that $\mu_1 < ... < \mu_i < ... < \mu_k$ and $\mu_{i+1} - \mu_i > \delta$, $\forall i = 1,...,k-1$, correct ranking occurs if we have $\bar{X}_i(\alpha_i n) \le \bar{X}_{i+1}(\alpha_{i+1} n)$ for all $i = 1,..,k-1$. Therefore, the probability of correct ranking $P(CR)$ can be written as follows:

$$P(CR)=\left\{\bigcap_{i=1,\dots,k-1}\left(\bar{X}_i(\alpha_i n)\le \bar{X}_{i+1}(\alpha_{i+1}n)\right)\right\} \tag{3.1}$$

To maximize the probability of correct ranking, we can formulate this problem as an optimal computing budget allocation model as follows.

$$\begin{aligned}&\max_{\alpha_1,\dots,\alpha_k} P(CR)\\ &s.t.\ \ \alpha_1+\alpha_2+\dots+\alpha_k=1\ ,\alpha_i\ge 0, i=1,\dots,k\end{aligned} \tag{3.2}$$

The objective of maximizing the probability of correct ranking is equivalent to minimizing the probability of false ranking. We will explore large deviation theory to asymptotically minimize the probability of false ranking.

3.3. Assumptions

Assumption 3.1: The performance of every design is independent.

The independence of each design ensures that the samples $(X_{i1},\dots,X_{i,\alpha_i n})$ for $i=1,\dots,k$ that we generated are independent. Thus, the results we obtained will not be affected by the correlations among different designs.

Define the cumulant generating function of sample mean $\bar{X}_i(n)$ to be $\Lambda_i^{(n)}(\theta)=\ln E(e^{\theta\bar{X}_i(n)})$. The effective domain of any function $f:D_f\to R^*$ is the set $\{x\in D_f : f(x)<\infty\}$, while the range is $R^*=R\bigcup\{+\infty\}$. Let $D_{\Lambda_i}=\{\theta\in R:\Lambda_i(\theta)<\infty\}$ and $F_i=\{\Lambda_i^{'}(\theta):\theta\in D_{\Lambda_i}^0\}$. For any set A, A^o denotes its interior and $\bar{A}$ denotes its closure.

Assumption 3.2 For every $i=1,...,k$,

(1) The limit $\Lambda_i(\theta)=\lim_{n\to\infty}\frac{1}{n}\Lambda_i^{(n)}(n\theta)$ is well defined as an extended real number for all θ.

(2) The origin belongs to $D_{\Lambda_i}^0$.

(3) $\Lambda_i(.)$ is strictly convex and steep, i.e., $\lim_{n\to\infty}|\Lambda_i^{'}(\theta_n)|=\infty$, where $\{\theta_n\}$ is a sequence converging to the boundary point of D_{Λ_i}.

(4) $[\mu_1,\mu_k]\subset\bigcap_{i=1}^{k}F_i^0$.

Assumption 3.2 implies that $\mu_i=\Lambda_i^{'}(0)$ with $\bar{X}_i(n)\to\mu_i$ a.s. when $n\to\infty$. Furthermore, it indicates that the sample mean $\bar{X}_i(\alpha_i n)$ will satisfy the large deviation principle. The last condition in assumption 3.2 ensures that the sample means of every design can take any value between μ_1 and μ_k, and $P\left(\bar{X}_i(\alpha_i n)\geq\bar{X}_{i+1}(\alpha_{i+1}n)\right)>0$.

3.4. Asymptotically Optimal Allocation Strategy

The correct ranking event happens when every design is at its correct position. For a minimization problem, this can be denoted as $\bar{X}_1(\alpha_1 n)<...<\bar{X}_i(\alpha_i n)<...<\bar{X}_k(\alpha_k n)$ since $\mu_1<...<\mu_i<...<\mu_k$ is assumed. The false ranking event happens when $\bar{X}_i(\alpha_i n)\geq\bar{X}_{i+1}(\alpha_{i+1}n)$ for any $i=1,...,k-1$. Mathematically, we can denote the false ranking probability as

$$P(FR) = P\left\{ \bigcup_{i=1,\dots,k-1} \left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\} \tag{3.3}$$

Note that $P(FR) = P\left\{ \bigcup_{i=1,\dots,k-1} \left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\}$ is bounded from below by

$$\max_{1\leq i\leq k-1} P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right)$$

and bounded from above by

$$(k-1)\max_{1\leq i\leq k-1} P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right).$$

Thus, for $i = 1,2,\dots,k-1$,

$$\lim_{n\to\infty} \frac{1}{n} \ln P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) = -G_i(\alpha_i, \alpha_{i+1}),$$

Hence,

$$\lim_{n\to\infty} \frac{1}{n} \ln P(FR) = \lim_{n\to\infty} \frac{1}{n} \ln P\left\{ \bigcup_{i=1,\dots,k-1} \left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\}$$

Since the function $\ln(\bullet)$ is strictly increasing, it is easy to see that

$$\begin{aligned} &\lim_{n\to\infty} \frac{1}{n} \ln \max_{1\leq i\leq k-1} P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) \\ &\leq \lim_{n\to\infty} \frac{1}{n} \ln P(FR) \leq \lim_{n\to\infty} \frac{1}{n} \ln\, (k-1)\max_{1\leq i\leq k-1} P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) \end{aligned}$$

$$\begin{aligned}
&\lim_{n\to\infty}\frac{1}{n}\ln\,(k-1)\max_{1\le i\le k-1}P\left(\bar{X}_i(\alpha_i n)\ge\bar{X}_{i+1}(\alpha_{i+1}n)\right)\\
&=\lim_{n\to\infty}\frac{1}{n}\left(\ln\,(k-1)+\ln\,\max_{1\le i\le k-1}P\left(\bar{X}_i(\alpha_i n)\ge\bar{X}_{i+1}(\alpha_{i+1}n)\right)\right)\\
&=\lim_{n\to\infty}\frac{1}{n}\ln\,(k-1)+\lim_{n\to\infty}\frac{1}{n}\ln\max_{1\le i\le k-1}P\left(\bar{X}_i(\alpha_i n)\ge\bar{X}_{i+1}(\alpha_{i+1}n)\right)\\
&=\lim_{n\to\infty}\frac{1}{n}\ln\,\max_{1\le i\le k-1}P\left(\bar{X}_i(\alpha_i n)\ge\bar{X}_{i+1}(\alpha_{i+1}n)\right)
\end{aligned}$$

Therefore, we have,

$$\lim_{n\to\infty}\frac{1}{n}\ln P(FR)=-\min_{1\le i\le k-1}G_i(\alpha_i,\alpha_{i+1}) \qquad (3.4)$$

Lemma 3.1 The rate function of false ranking probability of k designs is

$$-\lim_{n\to\infty}\frac{1}{n}\ln P(FR)=\min_{1\le i\le k-1}\inf_x\left(\alpha_i I_i(x)+\alpha_{i+1}I_{i+1}(x)\right) \qquad (3.5)$$

where $I(x)=\sup_{\theta\in R}\left(\theta x-\Lambda(\theta)\right)$ is the Fenchel-Legendre transform and $\Lambda(\theta)=\ln E(e^{\theta X})$.

Proof: Let $Y_n=\left(\bar{X}_i(\alpha_i n),\bar{X}_{i+1}(\alpha_{i+1}n)\right), n=1,2,...$ .The cumulant moment generating function of Y_n can be written as

$$\Lambda_n\left(\theta_i,\theta_{i+1}\right)=\ln E\left(e^{\theta_i\bar{X}_i(\alpha_i n)+\theta_{i+1}\bar{X}_{i+1}(\alpha_{i+1}n)}\right)=\Lambda_i^{(\alpha_i n)}\left(\theta_i/\alpha_i n\right)+\Lambda_{i+1}^{(\alpha_{i+1}n)}\left(\theta_{i+1}/\alpha_{i+1}n\right)$$

Under assumption 3.2, we know that

$$\lim_{n\to\infty}\frac{1}{n}\Lambda_n(n\theta_i,n\theta_{i+1})=\alpha_i\Lambda_i(\theta_i/\alpha_i)+\alpha_{i+1}\Lambda_{i+1}(\theta_{i+1}/\alpha_{i+1}).$$

From Crammer's Theorem, $\{Y_n, n=1,2,...\}$ satisfies large deviation principle with good rate function

$$
\begin{aligned}
& I(x_i, x_{i+1}) \\
& = \sup_{\theta_i, \theta_{i+1}} \left(\theta_i x_i + \theta_{i+1} x_{i+1} - \alpha_i \Lambda_i(\theta_i / \alpha_i) - \alpha_{i+1} \Lambda_{i+1}(\theta_{i+1} / \alpha_{i+1}) \right) \\
& = \sup_{\theta_i} \left(\theta_i x_i - \alpha_i \Lambda_i(\theta_i / \alpha_i) \right) + \sup_{\theta_{i+1}} \left(\theta_{i+1} x_{i+1} - \alpha_{i+1} \Lambda_{i+1}(\theta_{i+1} / \alpha_{i+1}) \right) \\
& = \alpha_i \sup_{\theta_i / \alpha_i} \left((\theta_i / \alpha_i) x_i - \Lambda_i(\theta_i / \alpha_i) \right) + \alpha_{i+1} \sup_{\theta_{i+1} / \alpha_{i+1}} \left((\theta_{i+1} / \alpha_{i+1}) x_{i+1} - \Lambda_{i+1}(\theta_{i+1} / \alpha_{i+1}) \right) \\
& = \alpha_i I_i(x_i) + \alpha_{i+1} I_{i+1}(x_{i+1})
\end{aligned}
$$

Hence, from large deviation principle, we know that

$$\lim_{n\to\infty} \frac{1}{n} \ln P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1} n) \right) = -G_i(\alpha_i, \alpha_{i+1}) = -\inf_x \left(\alpha_i I_i(x) + \alpha_{i+1} I_{i+1}(x) \right)$$

Therefore,

$$\lim_{n\to\infty} \frac{1}{n} \ln P(FR) = -\min_{1\leq i\leq k-1} G_i(\alpha_i, \alpha_{i+1}) = -\min_{1\leq i\leq k-1} \inf_x \left(\alpha_i I_i(x) + \alpha_{i+1} I_{i+1}(x) \right) . \square$$

To maximize the probability of correct ranking is equivalent to minimize the false ranking probability. From Lemma 3.1, the asymptotically optimal allocation strategy will result from maximizing the rate at which *P*(*FR*) goes to zero as a function of $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_k)$. Thus, we wish to find the best $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_k)$ that solves the following optimization problem:

$$
\begin{aligned}
& \max \min_{1\leq i\leq k-1} G_i(\alpha_i, \alpha_{i+1}) \\
& s.t. \quad \sum_{i=1}^{k} \alpha_i = 1, \alpha_i \geq 0, \forall i = 1,..,k
\end{aligned}
$$

It can be re-expressed as follows:

$$\begin{aligned} &\max \quad z \\ &s.t. \quad G_i(\alpha_i, \alpha_{i+1}) - z \geq 0, i = 1,...,k-1 \\ &\sum_{i=1}^{k} \alpha_i = 1, \alpha_i \geq 0, \ \forall i = 1,..,k \end{aligned} \tag{3.6}$$

$G_i(\alpha_i, \alpha_{i+1}) = \inf_x \left(\alpha_i I_i(x) + \alpha_{i+1} I_{i+1}(x) \right)$ is a concave and strictly increasing function of α_i and α_{i+1} as shown in (Glynn and Juneja, 2004). Therefore, the optimization problem (3.6) is a concave programming problem. Thus, the first order condition is also the optimality condition. We use this to determine the optimal allocation strategy in the following theorem.

Theorem 3.1 Under assumptions 3.1 and 3.2, if the optimal allocation $\boldsymbol{\alpha}^* > 0, \sum_{i=1}^{k} \alpha_i = 1$ minimizes the probability of false ranking asymptotically, then,

$$\begin{cases} G_1(\alpha_1^*, \alpha_2^*) = ... = \min\left(G_{i-1}(\alpha_{i-1}^*, \alpha_i^*), G_i(\alpha_i^*, \alpha_{i+1}^*) \right) \\ \qquad = ... = G_{k-1}(\alpha_{k-1}^*, \alpha_k^*), i = 2,..,k-2 \\ \sum_{i=1}^{k} \alpha_i^* = 1, \alpha_i^* > 0 \ \ \forall i = 1,...,k \end{cases} \tag{3.7}$$

where $G_i(\alpha_i, \alpha_{i+1}) = \inf_x (\alpha_i I_i(x) + \alpha_{i+1} I_{i+1}(x))$.

Proof: Since the optimization problem (3.6) is concave and $\boldsymbol{\alpha}^*$ is strictly positive, the first order condition is also the optimality condition. From the Karush–Kuhn–Tucker conditions, there exist $\lambda_i \geq 0, i = 1,...,k-1$ and $\gamma > 0$ such that,

$$1 - \sum\nolimits_{i=1}^{k-1} \lambda_i = 0 \tag{3.8}$$

$$\lambda_1 \frac{\partial G_1(\alpha_1^*, \alpha_2^*)}{\partial \alpha_1} = \gamma \tag{3.9}$$

$$\lambda_{k-1} \frac{\partial G_{k-1}(\alpha_{k-1}^*, \alpha_k^*)}{\partial \alpha_k} = \gamma \tag{3.10}$$

$$\lambda_i \frac{\partial G_i(\alpha_i^*, \alpha_{i+1}^*)}{\partial \alpha_i} + \lambda_{i-1} \frac{\partial G_{i-1}(\alpha_{i-1}^*, \alpha_i^*)}{\partial \alpha_i} = \gamma, 2 \le i \le k-2 \tag{3.11}$$

$$\lambda_i (z - G_i(\alpha_i^*, \alpha_{i+1}^*)) = 0 \quad 1 \le i \le k-1 \tag{3.12}$$

$$\lambda_i \ge 0 \quad 1 \le i \le k-1 \tag{3.13}$$

From (3.8) and (3.13), we could conclude that $\lambda_i > 0$ for some i. From (3.9), we see that if $\lambda_1 = 0$, then $\gamma = 0$. Since $\partial G_i(\alpha_i^*, \alpha_{i+1}^*)/\partial \alpha_i > 0$, it will result in $\lambda_{k-1} = 0$ if we substitute $\gamma = 0$ into (3.10). Substituting $\lambda_1 = 0, \gamma = 0$ into (3.11) results in $\lambda_1 = \lambda_2 = ... = \lambda_{k-1} = 0$, which contradicts with (3.8). Thus, $\lambda_1 > 0$ and $\gamma > 0$. Therefore, $\lambda_{k-1} > 0$ and $\max(\lambda_i, \lambda_{i-1}) > 0, i = 2,..,k-2$, which means that λ_i and λ_{i-1} cannot be zero at the same time for $2 \le i \le k-2$. From the constraint $G_i(\alpha_i, \alpha_{i+1}) - z \ge 0$, we know that for any i such that λ_i is zero, we have $G_i(\alpha_i^*, \alpha_{i+1}^*) \ge z$. For any $\lambda_i > 0$, we have $G_i(\alpha_i^*, \alpha_{i+1}^*) = z$ because of the complementary slackness condition in equation (3.12). Since we know that $\lambda_1, \lambda_{k-1} > 0$ and $\max(\lambda_i, \lambda_{i-1}) > 0$, for $i = 2,..,k-2$, therefore, $G_1(\alpha_1^*, \alpha_2^*) = G_{k-1}(\alpha_{k-1}^*, \alpha_k^*) = \min(G_{i-1}(\alpha_{i-1}^*, \alpha_i^*), G_i(\alpha_i^*, \alpha_{i+1}^*)), i = 2,..,k-2$.□

In simulation literature, most research works assume that the simulation output is normally distributed since the noise of simulation is normally distributed. We will demonstrate the asymptotically optimal allocation rule in the case of normally distributed design performance.

Suppose the performance of the design follows the normal distribution $X_i \sim N(\mu_i, \sigma_i^2),\ i = 1, ..., k$. Since the rate function for normal distribution is $I_i(x) = (x - \mu_i)^2 / 2\sigma_i^2$, we will obtain

$$G_i(\alpha_i, \alpha_{i+1}) = \inf_x \left(\alpha_i (x - \mu_i)^2 / 2\sigma_i^2 + \alpha_{i+1}(x - \mu_{i+1})^2 / 2\sigma_{i+1}^2 \right).$$

Since $I_i(x)$ is strictly convex, the sum of convex functions is convex, and the infimum of a convex function is also convex, therefore, the infimum can be achieved by differentiation with respect to x. We have,

$$x_i^* = \left(\frac{\alpha_i / \sigma_i^2}{\alpha_i / \sigma_i^2 + \alpha_{i+1} / \sigma_{i+1}^2} \right) \mu_i + \left(\frac{\alpha_{i+1} / \sigma_{i+1}^2}{\alpha_i / \sigma_i^2 + \alpha_{i+1} / \sigma_{i+1}^2} \right) \mu_{i+1}.$$

Therefore,

$$G_i(\alpha_i, \alpha_{i+1}) = \frac{(\mu_i - \mu_{i+1})^2}{2(\sigma_i^2 / \alpha_i + \sigma_{i+1}^2 / \alpha_{i+1})}. \tag{3.14}$$

The optimal allocation rule from Theorem 3.1 is such that,

$$\begin{cases} \dfrac{(\mu_1 - \mu_2)^2}{\sigma_1^2 / \alpha_1^* + \sigma_2^2 / \alpha_2^*} = ... = \min(\dfrac{(\mu_{i-1} - \mu_i)^2}{\sigma_{i-1}^2 / \alpha_{i-1}^* + \sigma_i^2 / \alpha_i^*}, \dfrac{(\mu_i - \mu_{i+1})^2}{\sigma_i^2 / \alpha_i^* + \sigma_{i+1}^2 / \alpha_{i+1}^*}) \\ = ... = \dfrac{(\mu_k - \mu_{k-1})^2}{\sigma_k^2 / \alpha_k^* + \sigma_{k-1}^2 / \alpha_{k-1}^*}, i = 2, .., k-2 \\ \sum_{i=1}^{k} \alpha_i^* = 1, \alpha_i^* > 0 \quad \forall i = 1, ..., k \end{cases} \tag{3.15}$$

3.5. Approximated Allocation Strategy

The allocation rule obtained from the previous section is asymptotically optimal in terms of maximizing the false ranking convergence rate. However, as shown in equations (3.7) and (3.15), there is no closed-form allocation rule even for the simple normal distribution. In order to derive an easily implementable allocation rule, we propose an approximated allocation strategy derived from minimizing an upper bound of false ranking probability.

Define a strictly increasing sequence $\{c_i, i = 0,1,...,k\}$ such that $c_i = (\mu_i + \mu_{i+1})/2,\ \mu_0 = -\infty, \mu_{k+1} = +\infty$. A lower bound for the probability of correct ranking can be approximated as

$$\begin{aligned} P(CR) &= P\left\{ \bigcap_{i=1,...,k-1} \left(\bar{X}_i(\alpha_i n) \le \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\} \\ &\ge P\left\{ \bigcap_{i=1,...,k} \left(c_{i-1} \le \bar{X}_i(\alpha_i n) \le c_i \right) \right\} \end{aligned} \tag{3.16}$$

We could establish the corresponding upper bound for the probability of false ranking as

$$\begin{aligned} P(FR) &= P\left\{ \bigcup_{i=1,...,k-1} \left(\bar{X}_i(\alpha_i n) \ge \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\} \\ &\le P\left\{ \bigcup_{i=1,...,k} \left(\left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \cup \left(\bar{X}_i(\alpha_i n) \ge c_i \right) \right) \right\} \\ &= \sum_{i=1}^{k} P\left\{ \left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \cup \left(\bar{X}_i(\alpha_i n) \ge c_i \right) \right\} \\ &= APFR \end{aligned} \tag{3.17}$$

To derive the large deviation rate function of the approximated probability of false ranking $(APFR)$, we should first prove that the large deviation principle is satisfied. It is easy to see from equation (3.17) that $APFR$ is bounded from below by

$$\max_i \left\{ P\left((\bar{X}_i(\alpha_i n) \le c_{i-1}) \cup (\bar{X}_i(\alpha_i n) \ge c_i) \right) \right\}$$

and bounded from above by

$$k \max_i \left\{ P\left((\bar{X}_i(\alpha_i n) \le c_{i-1}) \cup (\bar{X}_i(\alpha_i n) \ge c_i) \right) \right\}.$$

Therefore, for $i = 1, ..., k$,

$$\lim_{n\to\infty} \frac{1}{n} \ln P\left\{ \left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \right\} = -R_1(\alpha_i, c_{i-1})$$

$$\lim_{n\to\infty} \frac{1}{n} \ln P\left\{ \left(\bar{X}_i(\alpha_i n) \ge c_i \right) \right\} = -R_2(\alpha_i, c_i)$$

for some rate functions $R_1(\alpha_i, c_{i-1})$ and $R_2(\alpha_i, c_i)$. Then by the principle of largest term (Ganesh et al., 2004)

$$\begin{aligned} &\lim_{n\to\infty} \frac{1}{n} \ln P\left\{ \left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \cup \left(\bar{X}_i(\alpha_i n) \ge c_i \right) \right\} \\ &= \lim_{n\to\infty} \frac{1}{n} \ln \left\{ P\left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) + P\left(\bar{X}_i(\alpha_i n) \ge c_i \right) \right\} \\ &= -\min\left\{ R_1(\alpha_i, c_{i-1}), R_2(\alpha_i, c_i) \right\} \end{aligned} \tag{3.18}$$

Similarly,

$$\begin{aligned} &\lim_{n\to\infty} \frac{1}{n} \ln APFR \\ &= \lim_{n\to\infty} \frac{1}{n} \ln \sum_{i=1}^{k} P\left\{ \left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \cup \left(\bar{X}_i(\alpha_i n) \ge c_i \right) \right\} \\ &= -\min_i \left(\min\left\{ R_1(\alpha_i, c_{i-1}), R_2(\alpha_i, c_i) \right\} \right) \end{aligned} \tag{3.19}$$

Lemma 3.2 The rate functions $R_1(\alpha_i, c_{i-1})$ and $R_2(\alpha_i, c_i)$ are given as follows:

$$\begin{aligned} R_1(\alpha_i, c_{i-1}) &= \alpha_i I_i(c_{i-1}) \\ R_2(\alpha_i, c_i) &= \alpha_i I_i(c_i) \end{aligned} \qquad (3.20)$$

where $I(x) = \sup_{\theta \in R} \left(\theta x - \Lambda(\theta)\right)$ is the Fenchel-Legendre transform and $\Lambda(\theta) = \ln E(e^{\theta X})$.

Proof: Under assumption 3.2, we have

$$\lim_{n\to\infty} \frac{1}{n} \ln E\left(e^{\theta \bar{X}_i(\alpha_i n)}\right) = \alpha_i \ln \Lambda_i\left(\theta / \alpha_i\right).$$

Moreover,

$$\sup_{\theta\in R}\left\{c_{i-1}\theta - \alpha_i \Lambda_i(\theta/\alpha_i)\right\} = \alpha_i \sup_{\theta/\alpha_i}\left\{c_{i-1}\theta/\alpha_i - \Lambda_i(\theta/\alpha_i)\right\} = \alpha_i I_i(c_{i-1})$$

$$\sup_{\theta\in R}\left\{c_i\theta - \alpha_i \Lambda_i(\theta/\alpha_i)\right\} = \alpha_i \sup_{\theta/\alpha_i}\left\{c_i\theta/\alpha_i - \Lambda_i(\theta/\alpha_i)\right\} = \alpha_i I_i(c_i)$$

From Crammer's Theorem, we know that $\bar{X}_i(\alpha_i n)$ satisfies the large deviation principle. Thus,

$$\begin{aligned} &\lim_{n\to\infty} \frac{1}{n} \ln P\{(\bar{X}_i(\alpha_i n) \le c_{i-1})\} = -\alpha_i I_i(c_{i-1}) \\ &\lim_{n\to\infty} \frac{1}{n} \ln P\{(\bar{X}_i(\alpha_i n) \ge c_i)\} = -\alpha_i I_i(c_i). \quad \square \end{aligned}$$

Therefore, the rate function of the approximated probability of false ranking is

$$\begin{aligned}-\lim_{n\to\infty}\frac{1}{n}\ln P(APFR) &= \min_i\left(\min\left\{R_1(\alpha_i, c_{i-1}), R_2(\alpha_i, c_i)\right\}\right)\\ &= \min_i\left(\min\left\{\alpha_i I_i(c_{i-1}), \alpha_i I_i(c_i)\right\}\right).\end{aligned} \tag{3.21}$$

Our objective is to maximize the correct ranking probability. It is the same as minimizing the upper bound of false ranking probability. The asymptotically optimal allocation of minimizing the APFR is to solve for the best $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_k)$ from the following optimization problem.

$$\begin{aligned}&\max \min_i\left(\min\left\{\alpha_i I_i(c_{i-1}), \alpha_i I_i(c_i)\right\}\right)\\ s.t.\quad &\textstyle\sum_{i=1}^{k}\alpha_i = 1, \alpha_i \geq 0, \forall i = 1,..,k\end{aligned}$$

This can be re-expressed as,

$$\begin{aligned}\max\quad & z\\ s.t.\quad & \min\left\{\alpha_i I_i(c_{i-1}), \alpha_i I_i(c_i)\right\} - z \geq 0, 1 \leq i \leq k\\ & \sum_{i=1}^{k}\alpha_i = 1, \alpha_i \geq 0, 1 \leq i \leq k\end{aligned} \tag{3.22}$$

Theorem 3.2 below gives the optimal solution for (3.22).

Theorem 3.2 Under assumptions 3.1 and 3.2, if the optimal allocation $\boldsymbol{\alpha}^* > 0, \sum_{i=1}^{k}\alpha_i = 1$ minimizes the approximated probability of false ranking asymptotically, then,

$$\alpha_i^* \min(I_i(c_{i-1}), I_i(c_i)) = \alpha_j^* \min(I_j(c_{j-1}), I_j(c_j)), i, j \in \{1, ..., k\}.$$

Proof: Since (3.22) is a concave programming problem, the first order condition is the optimality condition. Therefore, from the Karush–Kuhn–Tucker conditions, there exist λ_i and $\gamma > 0$ such that,

$$1-\sum_{i=1}^{k}\lambda_i = 0 \tag{3.23}$$

$$\lambda_i \min(I_i(c_{i-1}), I_i(c_i)) = \gamma, \quad 1 \le i \le k \tag{3.24}$$

$$\lambda_i[z-\alpha_i^* \min(I_i(c_{i-1}), I_i(c_i))] = 0, \ 1 \le i \le k \tag{3.25}$$

$$\lambda_i \ge 0 \quad 1 \le i \le k \tag{3.26}$$

From equation (3.23) we know that λ_i must be positive for some i. However, since $\min(I_i(c_{i-1}), I_i(c_i))$ is always positive, any $\lambda_i = 0$ will lead to all $\lambda_i = 0$. Therefore, we have $\lambda_i > 0$ for all i. As a result of (3.25), we conclude that

$$\alpha_i^* \min(I_i(c_{i-1}), I_i(c_i)) = \alpha_j^* \min(I_j(c_{j-1}), I_j(c_j)), \forall i, j \in \{1,..,k\}. \ \square$$

In the case of the normal distribution, the performance of design $i, i = 1,...,k$ is normally distributed with $X_i \sim N(\mu_i, \sigma_i^2)$. The rate function of normal distribution is known as $I_i(x_i) = (\mu_i - x_i)^2 / 2\sigma_i^2$. Therefore,

$$\begin{aligned}\min(I_i(c_{i-1}), I_i(c_i)) &= \frac{\min\left\{(\mu_i/2-\mu_{i+1}/2)^2, (\mu_i/2-\mu_{i-1}/2)^2\right\}}{2\sigma_i^2} \\ &= \frac{\min\left\{(\mu_i-\mu_{i+1})^2, (\mu_i-\mu_{i-1})^2\right\}}{8\sigma_i^2}\end{aligned}$$

As a result of Theorem 3.2, the resulting simulation budget allocation rule can be computed as

$$\frac{\alpha_i^*}{\alpha_j^*} = \frac{\sigma_i^2 / \min\{(\mu_i - \mu_{i+1})^2, (\mu_i - \mu_{i-1})^2\}}{\sigma_j^2 / \min\{(\mu_j - \mu_{j+1})^2, (\mu_j - \mu_{j-1})^2\}}, \forall i, j \in \{1,...,k\}. \qquad (3.27)$$

3.6. Sequential Allocation Procedure

From Theorems 3.1 and 3.2, we know that the allocation rule or the value of $\boldsymbol{\alpha}$ can only be determined after we know the distribution of the design performance. In actual implementation, the distribution of the design performance is unknown. We will propose a sequential allocation rule and use the sampling distribution to estimate the allocation step by step. We have assumed $\mu_{i+1} - \mu_i > \delta, \forall i = 1,...,k-1$, i.e., the mean performance of every design is different. We further assume that the variance of the performance is finite. Together with assumptions 3.1 and 3.2, the sequential allocation strategy for complete ranking of *k* designs is proposed as follows.

Define *l* to be the iteration number and $N_i^l, i = 1,...,k$ to be the total number of simulation replications that have been allocated to design *i* up to iteration *l*. *n* is the total number of simulation replications available. Δ is the number of incremental simulation replications for each iteration.

Step 0: Perform n_0 simulation replications for every design.

$$l \leftarrow 0, N_1^l = ... = N_k^l = n_0$$

Step 1: If $\sum_{i=1}^{k} N_i^l \geq n$, stop.

Step 2: Increase the computation budget by Δ and compute the new budget allocation using Theorem 3.1 or Theorem 3.2.

Step 3: Perform additional $\max(0, N_i^{l+1} - N_i^l)$ simulation runs for design

$i, i = 1,..,k$, $l \leftarrow l+1$. Go back to Step 1.

As the simulation continues, design *i* will be ranked number *i* for all *i*. However, the ranking of each design may change from iteration to iteration, although it will converge to the true ranking when the total computation budget goes to infinity. When ranking changes, the budget allocation in Step 2 will be applied immediately. Therefore, the actual proportion of budget for every system will converge to the optimal proportion when the number of iterations is sufficiently large.

Furthermore, we need to take note of n_0, the initial number of replications for every system. n_0 cannot be too small because the estimation of the rate function can be poor especially when the variance of the performance is large. On the other hand, if n_0 is too large, some portions of the system will be allocated excessively compared with its optimal allocation number. When the total budget is very limited, those portions that need more replications may suffer from large n_0 and this would eventually affect the simulation results. Other than the initial number of replications, the incremental budget Δ is also important in the implementation procedure. Large Δ may result in the wasting of budget, while small Δ will lead to expensive computation in Step 2.

3.7. Consistent Estimator

It is natural to think that there will be significant variation in terms of performance because of the variability in the estimation. However, we will demonstrate in this section that the estimated optimal allocation strategy will converge to the true optimal strategy as the total budget $n \to \infty$. To simplify notations, we assume that each system is allocated n samples when we prove the consistency. Recall that the samples generated for design i are denoted by $(X_{i1},...,X_{in})$. The empirical cumulant generating function of system i could be written as $\Lambda_i^{(n)}(\theta) = \ln((1/n)\sum_{j=1}^{n} e^{\theta \bar{X}_{ij}})$ and the empirical rate function is $I_i^{(n)}(x) = \sup_{\theta}\{\theta x - \Lambda_i^{(n)}(\theta)\}$. Therefore, we are using $I_i^{(n)}(x)$ to estimate $I_i(x)$ for the approximated allocation strategy, and using $G_i^{(n)}(\alpha_i, \alpha_{i+1})$ to estimate $G_i(\alpha_i, \alpha_{i+1})$ for the optimal allocation strategy.

Theorem 3.3 below explains why $G_i^{(n)}(\alpha_i, \alpha_{i+1})$ is a consistent estimator of $G_i(\alpha_i, \alpha_{i+1})$.

Theorem 3.3 The empirical estimation of the optimal allocation is consistent, i.e.,

$$\alpha_i^*(n) \to \alpha_i^*, \forall i = 1,..,k \text{ a.s. when } n \to +\infty .$$

Proof: It has been argued that $I_i^{(n)}(x) \to I_i(x)$ almost surely in (Glynn and Juneja, 2004). By a similar argument, we could conclude that the estimator $\alpha_i^*(n) \to \alpha_i^*, \forall i = 1,..,k$ as $n \to +\infty$ almost surely.

It has also been shown in Theorem 2 of Glynn and Juneja (2004) that $G_i^{(n)}(\alpha_1,\alpha_i) \to G_i(\alpha_1,\alpha_i), \forall i = 2,...,k$. Replacing $G_i^{(n)}(\alpha_1,\alpha_i)$ by $G_i^{(n)}(\alpha_i,\alpha_{i+1})$ will not affect the proof. Thus, we conclude that $G_i^{(n)}(\alpha_i,\alpha_{i+1}) \to G_i(\alpha_i,\alpha_{i+1})$ almost surely.□

3.8. Numerical Experiments

In this section, we will test our proposed algorithms with a series of numerical experiments. We will compare with the asymptotically optimal allocation (AOA) in Section 3.4, approximated allocation (AA) in Section 3.5 and equal allocation (EA). Comparisons with equal allocation and indifference zone (IZ) formulation (Bishop, 1978; Beirlant et al., 1982) are also performed. Our numerical experiments have all assumed that the performance of every design follows a normal distribution.

3.8.1. Comparing AOA and AA with EA

The AOA provided in Section 3.4 is the asymptotically optimal allocation rule in terms of maximizing the convergence rate of false ranking probability. We also present an approximated formulation in Section 3.5 which maximizes the convergence rate of approximated probability of false ranking. We will illustrate the difference of the two allocation rules and compare them through numerical experiments based on normally distributed design performance.

For any $i \neq j$, the allocation from AOA is such that,

$$
\begin{aligned}
&\min(\frac{(\mu_{i-1}-\mu_i)^2}{\sigma_{i-1}^2/\alpha_{i-1}^*+\sigma_i^2/\alpha_i^*},\frac{(\mu_i-\mu_{i+1})^2}{\sigma_i^2/\alpha_i^*+\sigma_{i+1}^2/\alpha_{i+1}^*}) \\
&=\min(\frac{(\mu_{j-1}-\mu_j)^2}{\sigma_{j-1}^2/\alpha_{j-1}^*+\sigma_j^2/\alpha_j^*},\frac{(\mu_j-\mu_{j+1})^2}{\sigma_j^2/\alpha_j^*+\sigma_{j+1}^2/\alpha_{j+1}^*})
\end{aligned}
$$

where $\mu_0 = -\infty$ and $\mu_{k+1} = +\infty$.

Therefore, the percentage of allocation is positively correlated with the variance and inversely correlated with the square of the difference with its neighbors. From Section 3.5, we also know that the approximated allocation rule is such that,

$$
\frac{\alpha_i^*}{\alpha_j^*}=\frac{\sigma_i^2/\min\{(\mu_i-\mu_{i+1})^2,(\mu_i-\mu_{i-1})^2\}}{\sigma_j^2/\min\{(\mu_j-\mu_{j+1})^2,(\mu_j-\mu_{j-1})^2\}},\forall i,j\in\{1,...,k\},
$$

which also shows that the percentage of allocation is positively correlated with the variance and inversely correlated with the square of difference with its neighbors. The difference between them is that the AOA considers the allocation of neighbors simultaneously, while AA considers the allocation of neighbors separately.

In the case of three designs, AOA yields

$$
\begin{cases}
\dfrac{(\mu_1-\mu_2)^2}{\sigma_1^2/\alpha_1^*+\sigma_2^2/\alpha_2^*} & =\dfrac{(\mu_2-\mu_3)^2}{\sigma_2^2/\alpha_2^*+\sigma_3^2/\alpha_3^*} \\
\alpha_1^*+\alpha_2^*+\alpha_3^*=1 & \alpha_1^*,\alpha_2^*,\alpha_3^*>0
\end{cases}
\tag{3.28}
$$

while AA yields

$$\begin{cases} \dfrac{\alpha_1^*}{\alpha_2^*} = \dfrac{\sigma_1^2 / (\mu_1 - \mu_2)^2}{\sigma_2^2 / \min\left\{(\mu_1 - \mu_2)^2, (\mu_2 - \mu_3)^2\right\}} \\ \alpha_1^* + \alpha_2^* + \alpha_3^* = 1 \qquad \alpha_1^*, \alpha_2^*, \alpha_3^* > 0 \\ \dfrac{\alpha_1^*}{\alpha_3^*} = \dfrac{\sigma_1^2 / (\mu_1 - \mu_2)^2}{\sigma_3^2 / (\mu_2 - \mu_3)^2} \end{cases} \tag{3.29}$$

If we use the results in (3.29) and substitute them into equation (3.28), we have the following relationships of (3.28):

$$\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 / \alpha_1^* + \sigma_2^2 / \alpha_2^*} = \frac{(\mu_1 - \mu_2)^2}{\sigma_2^2 / \alpha_2^* (1 + \dfrac{(\mu_1 - \mu_2)^2}{\min\{(\mu_1 - \mu_2)^2, (\mu_2 - \mu_3)^2\}})}$$

$$\frac{(\mu_2 - \mu_3)^2}{\sigma_2^2 / \alpha_2^* + \sigma_3^2 / \alpha_3^*} = \frac{(\mu_2 - \mu_3)^2}{\sigma_2^2 / \alpha_2^* (1 + \dfrac{(\mu_2 - \mu_3)^2}{\min\{(\mu_1 - \mu_2)^2, (\mu_2 - \mu_3)^2\}})}$$

Without loss of generality, we could assume that $(\mu_1 - \mu_2)^2 \geq (\mu_2 - \mu_3)^2$, leading to

$$\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 / \alpha_1^* + \sigma_2^2 / \alpha_2^*} = \frac{\sigma_2^2}{\alpha_2^*} \frac{(\mu_1 - \mu_2)^2}{(1 + \dfrac{(\mu_1 - \mu_2)^2}{(\mu_2 - \mu_3)^2})}$$

$$\frac{(\mu_2 - \mu_3)^2}{\sigma_2^2 / \alpha_2^* + \sigma_3^2 / \alpha_3^*} = \frac{\sigma_2^2}{\alpha_2^*} \frac{(\mu_2 - \mu_3)^2}{2}.$$

Therefore, as long as $(\mu_1 - \mu_2)^2$ and $(\mu_2 - \mu_3)^2$ are close to each other, AA will result in a similar allocation rule with AOA. Although this derivation is not accurate for more than 3 designs, a similar conclusion can be reached, i.e., AA and AOA rules will be close to each other when the mean differences between consecutive designs are the same or close.

3.8.1.1. Comparison of Convergence Rate of False Ranking Probability

The convergence rate of false ranking probability based on AOA should be the largest since it is the asymptotically optimal result. Given the mean and variance, we are able to calculate the allocation rules, i.e., the value of $\boldsymbol{\alpha}$, through Theorems 3.1 and 3.2 for AOA and AA respectively. Then we could use $\boldsymbol{\alpha}$ as the input to Lemma 3.1 to compute the false ranking rates under the three scenarios for AOA, AA and EA. The value of $\boldsymbol{\alpha}$ for EA is simply $(1/k,...,1/k)$.

In this numerical experiment, we will randomly generate 100 sets of designs. Each set has 10 designs with different means and variances. The convergence rates of false ranking probability of AOA, AA and EA are computed. We will compare these rates of the three allocation rules. Figure 3.1 below provides the box plot of the convergence rates of AA and EA, where the rate from AOA is used as a benchmark of 100%.

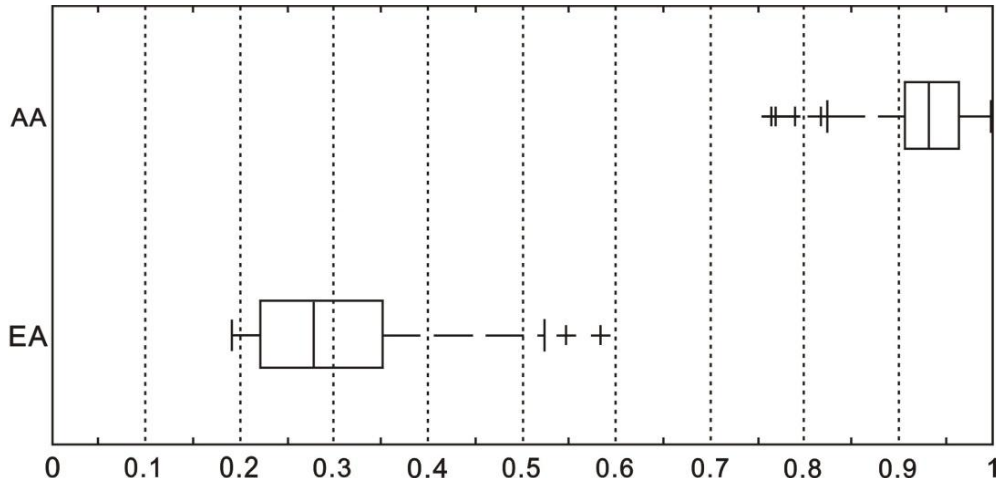

Fig.3.1. Boxplot for AA and EA.

The numerical results show that the convergence rate of false ranking probability under AA rule is very close to AOA in most of the scenarios. Most of the convergence rates computed are above 90% of the optimal rates. The convergence rate of false ranking probability under EA is much smaller than AOA and AA. This suggests that significant improvement can be made when compared with EA.

3.8.1.2. Comparison of Correct Ranking Probability

Although AOA performs better than AA in terms of convergence rates of false ranking probability as shown above, the empirical probability performance in finite budget may be different. Firstly, AOA is optimal under asymptotical analysis and this may not be true in the finite horizon situation. Secondly, it is optimal only when the convergence rate is used as the performance measurement. The result may be different when the correct ranking probability is used as the performance measurement.

In the following numerical experiments, we will test three different sets of designs. The performance of every design is assumed to be normally distributed. We will compare the performance of AA, AOA and EA in terms of correct ranking probability with one-time allocation of the total available computing budget.

The three numerical experiments are performed as follows. Experiment 1 sets equal spacing between consecutive designs and different variances for each design. Experiment 2 sets equal variances for each design but the spacing between consecutive designs is different. Experiment 3 sets both the spacing and the variances as being different. The mean and variance of each experiment are summarized in Table 3.1.

Table 3.1. Mean and variance of three experiments in chapter 3

	Equal Spacing		Equal Variance		Increasing Spacing Decresing Variance	
Design	Mean	Variance	Mean	Variance	Mean	Variance
I	2	1	1	100	1	100
II	4	4	2	100	2	81
III	6	9	4	100	4	64
IV	8	16	7	100	7	49
V	10	25	11	100	11	36
VI	12	36	16	100	16	25
VII	14	49	22	100	22	16
VIII	16	64	29	100	29	9
IX	18	81	37	100	37	4
X	20	100	46	100	46	1

The finite budget performances of the three allocation rules are simulated assuming known mean and variance. We vary the total budget *T* from 200 to 6000 for each allocation rule, and the probability of correct ranking (*PCR*) is estimated from 10,000 runs of simulation. The performances for all the three scenarios are shown in Figure 3.2.

From all the three experiments, it is clear that AA performs the best in finite budget in terms of correct ranking probability. However, the performance of AOA will catch up with AA when the amount of budget becomes large. The performance of AA and AOA is closest under equal spacing scenario. This matches our prediction above. It is also interesting to note that EA will have similar or even better performance than AA and AOA when the amount of budget is very small.

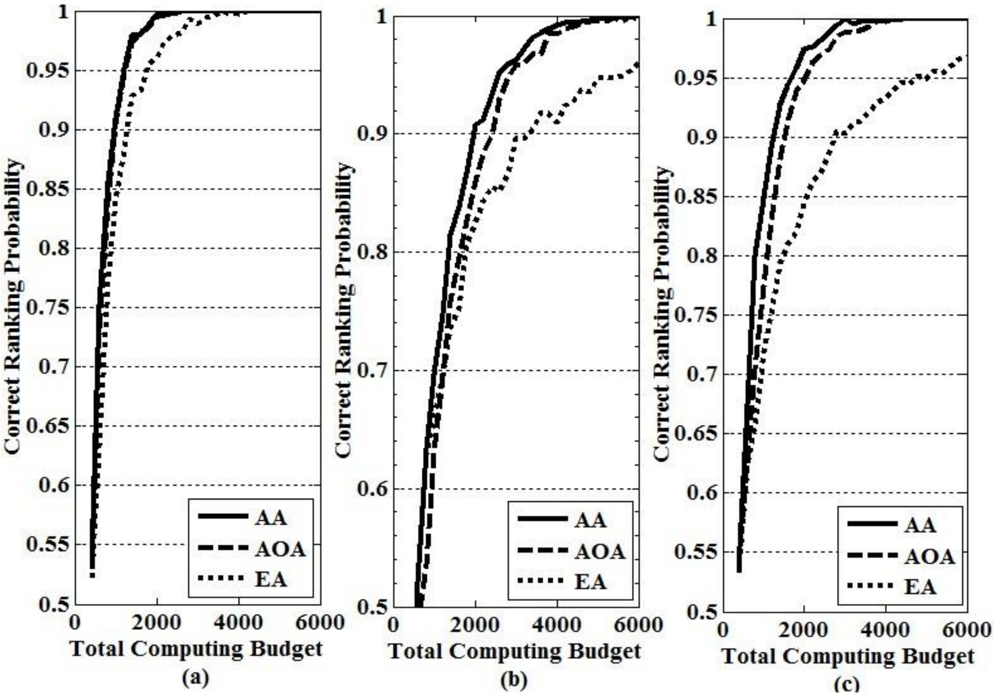

Fig.3.2. Correct ranking probability comparison for AA,AOA and EA. (a) is for equal spacing scenario; (b) is for equal variance scenario; (c) is for increasing spacing but decreasing variance scenario.

When the total computing budget is very small, designs with small percentage of allocation (non-critical designs) will receive extremely small number of replications. Although non-critical designs are easy to distinguish, too few replications would make them variable and hard to distinguish. However, if a few more replications are allocated to non-critical designs, they can be distinguished very easily. In other words, we can conclude that the marginal increase of correct ranking probability will be much larger if we allocate a few more replications to non-critical designs when the total budget is small. This explains why EA allocation is good when the total budget is very small. Although AOA is the asymptotically optimal allocation in terms of convergence rate, AA performs better in terms of correct ranking probability in finite horizon. AA rule is derived based on the upper bound of the false ranking probability, and the numerical experiments show that AA always allocates slightly less budget to critical designs and slightly more to non-critical designs compared with AOA. Following the same explanation, we could see that the reason why AA performs better in terms of correct ranking probability in finite budget than AOA is because AA gives a few more simulation replications to non-critical designs at an early stage. When the total budget is relatively large, the probability of correct ranking under either AOA or AA will go to 100%; therefore, little difference can be observed when total budget becomes very large.

3.8.2. Comparing AA and IZ with EA

The numerical experiments above are conducted assuming known mean and variance. In the case when the expected means and variances are not available, we need to use the sequential algorithm proposed in Section 3.6 to find the correct ranking. We have shown that AA will perform better than AOA in terms of probability when the amount of budget is limited. In addition, implementing AOA requires solving a nonlinear programming problem at every iteration. In actual simulation, we may need to solve it as many as millions of times. It not only increases the computational cost but also brings about instability because of the process of solving the nonlinear programming problem. Most importantly, we only have finite computation budget when faced with a practical simulation problem. Therefore, AA is preferred over AOA. We will use the AA rule when the sequential allocation algorithm is used, and compare it with the existing IZ procedure and EA rule.

In this numerical experiment, we will use the same set of data as shown in Table 3.1 and observe the performance when the sequential algorithm is used with AA, IZ procedure and EA rule. We summarize the allocation algorithms as follows:

AA procedure: Use the sequential algorithm proposed in Section 3.6. In Step 2, use the sample mean and sample variance as the estimation of the population mean and population variance to calculate the allocation rule in the next round based on Theorem 3.2.

IZ procedure: Step 1: Decide the pre-specified probability to achieve the indifference zone δ^* and check the value of h. Step 2: Simulate each design with n_0 replications and compute the sample variance s_i of design *i*. Step 3: Decide the second stage allocation for design i to be $n_i = \max\{n_0 + 1, (s_i h / \delta^*)^2\}$, and simulate $n_i - n_0$ additional replications.

EA: Equally allocate the computing budget for every design.

It should be noted that the pre-specified probability we decide at Step 1 of IZ procedure is the lower bound probability we could achieve, and the actual probability based on the allocation can be much higher than the specified probability. Therefore, we will use the IZ procedure and run the simulation to obtain the actual probability.

The data that we use is summarized in Table 3.1. We use the IZ formulation to decide how many simulation replications are needed for every pre-specified probability of 0.75, 0.8, 0.85, 0.9, 0.95, 0.975, and 0.99, and we run the simulation 10,000 times to obtain the actual probability of correct ranking. The performance of EA and AA is obtained through simulating the design 10,000 times by varying the total budget from 200 to 20,000. Figure

3.3 shows the results for the numerical experiments. The correct ranking probabilities in the horizontal axis are the actual probabilities based on the simulation experiments.

It is easy to see that our proposed algorithm AA performs the best in all scenarios. However, it is of interest that that the EA performs better than the IZ rule under the equal variance scenario. This is not a random event. Based on the IZ procedure, we could easily see that the budget for every design will be approximately equal because they have the same variances. Therefore, they will perform slightly worse than the EA because of the inherent variability within simulation. The feature of increasing spacing is not captured by the IZ procedure. Therefore, its performance is not better than EA.

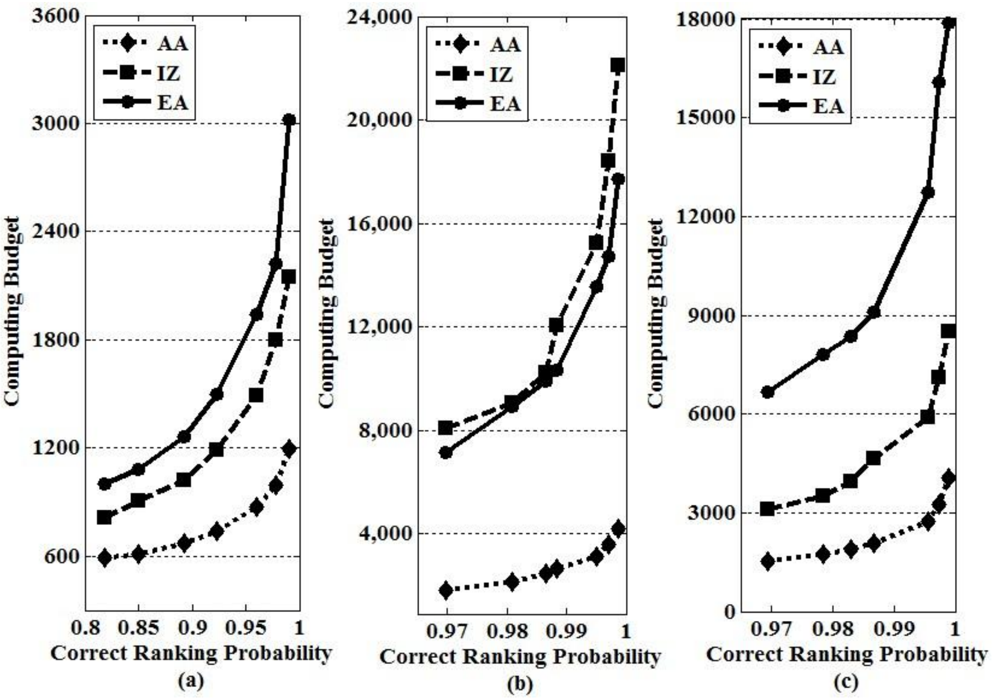

Fig.3.3. Computing budget comparison for AA,AOA and EA. (a) is for equal spacing scenario; (b) is for equal variance scenario; (c) is for increasing spacing but decreasing variance scenario.

3.9. Conclusion

In this chapter, we present the characterization of optimal allocation strategy for ranking of finite alternatives whose performance can only be estimated via simulation. Using the large deviation framework, we have presented two formulations and derived two allocation rules. The AOA rule is derived based on exact formulation and it is the optimal allocation rule under asymptotical analysis when convergence rate of false ranking probability is used as performance measurement. The AA rule performs better than the AOA rule in terms of correct ranking probability under finite budget. A sequential allocation algorithm is proposed and used together with the AA rule. This algorithm is easy to implement when the underlying distribution governing the performance value is unknown or assumed. We then compare our proposed sequential allocation algorithm with IZ formulation and EA rules. Our proposed algorithm performs the best in every situation and it shows that significant budget can be saved by using this algorithm.

Chapter 4. Efficient Simulation Budget Allocation for Ranking an Optimal Subset with an Application to Genetic Algorithms

Motivated by the idea of integrating ranking and selection procedure into evolutionary algorithms, we consider the problem of ranking top *m* designs. Top *m* ranking is also an important problem in statistical ranking and selection. It can be used to improve the simulation efficiency when the design performance can only be estimated with noise via simulation. On the other hand, top *m* ranking can be regarded as a generalization of complete ranking as presented in Chapter 3 and the OCBA problem of selecting a single best as in (Chen et al., 2000). The organization of this chapter is as follows. Section 4.1 provides the background and overview of the problem considered in this chapter. We will formulate an optimal computing budget model to solve the problem of ranking top *m* designs in Section 4.2. The asymptotically optimal allocation strategy is derived in Section 4.3. In Section 4.4, we propose an upper bound of the probability of correct ranking and derive a simple closed-form allocation rule. We provide a sequential allocation algorithm and prove the consistency of the estimators in Section 4.5. In Section 4.6, numerical experiments are conducted to compare different allocation rules. Furthermore, the allocation rules are integrated into genetic algorithms to show how our proposed allocation rule can enhance the search efficiency in solving stochastic simulation optimization problems. Lastly, we conclude this chapter in Section 4.7.

4.1. Overview

Genetic algorithm (GA) is a heuristic adaptive search method that mimics the process of natural evolution. The probabilistic search algorithm was first introduced by John Holland in 1970s. A higher probability of being selected to produce offspring will be given to better candidate solutions so that the solutions in the next generation will be better than that of the previous generation on average. GA has been widely used to solve deterministic optimization problems because it does not require knowing the problem structure. A comprehensive review of GA and its application can be found in books such as (Gen and Cheng, 2000), (Haupt and Haupt, 2004) and (Sivanandam and Deepa, 2010).

While it has been successfully applied to deterministic optimization problems, GA becomes computationally expensive when the evaluation of candidate solution is subject to noise. In particular, the computational burden becomes extremely heavy if the performance can only be estimated via simulation. This is due to the fact that we must simulate each solution a large number of times in order to obtain a steady mean fitness value. The accuracy of the estimator cannot be improved faster than $O(1/\sqrt{N})$, where N is the number of simulation replications. Schmidt et al. (2007) has proposed the idea of integrating statistical ranking into evolutionary algorithms when the ranking of candidate solutions is used as the selection criterion. In order to reduce the number of samples to be simulated, we must focus the sampling on those solutions that are critical to the evolutionary algorithms.

A typical process of a genetic algorithm includes selection, reproduction and replacement. In the selection step, better fitness solutions are selected and assigned a probability of being selected to produce the offspring. In the replacement step, the set of new and old solutions is reduced to the usual population size. There exist various selection schemes for GA, and they can be classified as proportionate selection, ranking selection and tournament selection. In this thesis, we consider the GA type in which ranking information is used as the selection criterion. In particular, we are using GA with exponential ranking selection scheme in our numerical experiments. Ranking-based selection scheme overcomes the scaling problem of the direct fitness-based approach (Gen and Cheng, 2000). In a deterministic scenario, GA has access to the complete ranking information. In a stochastic situation, it requires a huge number of simulation replications in order to obtain the ranking information.

Motivated by the idea of integrating statistical ranking procedure into evolutionary algorithms, we propose an efficient ranking procedure such that the selection and reproduction can be done simultaneously while a higher probability will be given to better fitness value solutions. This motivates us to develop an efficient ranking procedure which can select and rank the top *m* candidates out of a total population of *k* candidates simultaneously, where *k* is greater than *m*. Our objective is to determine the optimal allocation of simulation budget among the *k* designs in order to maximize the probability of correctly ranking the top *m* designs. The problem of ranking the top *m* designs falls into a research area called ranking and selection in statistics (Bechhofer, 1995). In recent years, ranking and selection procedures have been successfully applied in simulation (Swisher, 2003; Andradóttir, 2005). Three major approaches are summarized and compared in Branke et al. (2007).

Besides the application to genetic algorithms, top *m* ranking itself can be useful in many other aspects. If we want to determine the best *m* experiment parameter settings and their ranking, it is important to consider how to minimize the number of experimental runs, especially when the cost of the experiment is high. When the evaluation of a project is subject to multiple attributes including some qualitative measure, it can be useful to identify a few top choices and their relative ranking by considering one quantitative attribute. The best choice can be selected after considering all the attributes. In some scenarios, selecting the optimal subset without identifying their relative ranking may not be useful enough for the decision makers. Therefore, we propose the study of the problem of ranking the optimal subset.

In this chapter, we will formulate the problem of top *m* ranking using the OCBA framework, and derive the efficient simulation budget allocation rule. In addition, we will also show how the proposed allocation can be integrated with genetic algorithms to enhance the search efficiency for simulation optimization problem using GA in noisy environments.

4.2. Problem Formulation

Consider the problem of ranking the top *m* designs out of *k* alternatives. The performance of every design can only be estimated through simulation. The mean performance is used as the ranking criterion. In order to have a steady mean performance value, a large number of simulation replications are needed because of the randomness of individual samples. Given that we only have a total of n simulation replications available, our objective is to decide the best allocation of the total n replications to the k designs in order to maximize the probability of correctly ranking the top *m* designs.

Without loss of generality, we assume the mean performances of designs to be $\mu_1, \mu_2, \cdots, \mu_k$ respectively. The mean performance is such that $\mu_1 < ... < \mu_i < ... < \mu_k$ and $\mu_{i+1} - \mu_i \geq \delta, i = 1, ..., k-1$, where δ is a positive number. Let $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_k)$ be the proportion of the total computing budget n to be allocated to each design such that $\sum_{i=1}^{k} \alpha_i = 1, i = 1, ..., k.$ Let $\bar{X}_i(\alpha_i n) = (\alpha_i n)^{-1} \sum_{j=1}^{\alpha_i n} X_{ij}$ denote the sample mean performance of design i, where ($X_{i1}, ..., X_{i,\alpha_i n}$) denotes the samples from population i. We ignored the case when $\alpha_i n$ is not an integer because we could let $\alpha_i n$ be $\lfloor \alpha_i n \rfloor$, the greatest integer less than $\alpha_i n$. The analysis will remain unaffected. Our objective is to find the optimal allocation strategy $\boldsymbol{\alpha}^* = (\alpha_1^*, ..., \alpha_k^*)$ such that the probability of correctly ranking the top *m* designs can be maximized with a fixed limited computing budget *n*.

Under the assumption that $\mu_1 < ... < \mu_i < ... < \mu_k$, the correct ranking of the top m designs happens if $\bar{X}_i(\alpha_i n) \leq \bar{X}_{i+1}(\alpha_{i+1} n)$ for all $i = 1, .., m-1$ and $\bar{X}_m(\alpha_m n) \leq \bar{X}_j(\alpha_j n)$ for all $j = m+1, ..., k.$ Mathematically, we can write the probability of correctly ranking the top *m* designs as

$$P(CR_m) = P\left(\left\{ \bigcap_{i=1}^{m-1} \left(\bar{X}_i(\alpha_i n) \le \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\} \bigcap \left\{ \bigcap_{j=m+1}^{k} \left(\bar{X}_m(\alpha_m n) \le \bar{X}_j(\alpha_j n) \right) \right\} \right). \tag{4.1}$$

We can use this to formulate an optimal computing budget allocation problem that maximizes the probability of correctly ranking top *m* designs as follows:

$$\begin{aligned} &\max_{\alpha_1,\dots,\alpha_k} P(CR_m) \\ s.t.\ \ &\alpha_1 + \dots + \alpha_i + \dots + \alpha_k = 1, \alpha_i \ge 0, i = 1,\dots,k \end{aligned} \tag{4.2}$$

The objective of maximizing the probability of correctly ranking top *m* designs is equivalent to minimize the probability of falsely ranking top *m* designs. It is the same as maximizing the convergent rate at which the false ranking probability goes to zero. In this chapter, we will explore large deviation theory to derive this rate function. Therefore, the original OCBA problem can be reformulated as the problem of maximizing the convergent rate function. The assumptions we used here are the same as the assumptions in Section 3.3.

4.3. Asymptotically Optimal Allocation

The probability of correctly ranking top *m* designs is defined above in Section 4.2. The probability of falsely ranking top *m* designs is just its complement.

$$
\begin{aligned}
P(FR_m) &= 1 - P(CR_m) \\
&= P\left\{ \left(\left\{ \bigcap_{i=1}^{m-1} \left(\bar{X}_i(\alpha_i n) \le \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right\} \bigcap \left\{ \bigcap_{j=m+1}^{k} \left(\bar{X}_m(\alpha_m n) \le \bar{X}_j(\alpha_j n) \right) \right\} \right)^C \right\} \\
&= P\left\{ \left(\bigcup_{i=1}^{m-1} \left(\bar{X}_i(\alpha_i n) \ge \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right) \bigcup \left(\bigcup_{j=m+1}^{k} \left(\bar{X}_m(\alpha_m n) \ge \bar{X}_j(\alpha_j n) \right) \right) \right\}
\end{aligned}
\tag{4.3}
$$

where $(\bullet)^C$ represents its complement. Lemma 4.1 shows the convergent rate of the probability of falsely ranking top *m* designs.

Lemma 4.1 The rate function of $P(FR_m)$ is given by

$$
\begin{aligned}
&-\lim_{n\to\infty} \frac{1}{n} \ln P(FR_m) \\
&= \min\left\{ \min_{1\le i\le m-1} \inf_x \left(\alpha_i I_i(x) + \alpha_{i+1} I_{i+1}(x) \right),\ \min_{j=m+1,\dots,k} \inf_x \left(\alpha_j I_j(x) + \alpha_m I_m(x) \right) \right\}
\end{aligned}
$$

where $I(x) = \sup_{\theta\in R} \left(\theta x - \Lambda(\theta) \right)$ is the Fenchel-Legendre transform and $\Lambda(\theta) = \ln E(e^{\theta X})$.

Proof: $P(FR_m)$ is bounded below by

$$
\max\left\{ \max_{1\le i\le m-1} P\left(\bar{X}_i(\alpha_i n) \ge \bar{X}_{i+1}(\alpha_{i+1} n) \right),\ \max_{j=m+1,\dots,k} P\left(\bar{X}_m(\alpha_m n) \ge \bar{X}_j(\alpha_j n) \right) \right\}
$$

and bounded above by

$$
(k-1) * \max\left\{ \max_{1\le i\le m-1} P\left(\bar{X}_i(\alpha_i n) \ge \bar{X}_{i+1}(\alpha_{i+1} n) \right),\ \max_{j=m+1,\dots,k} P\left(\bar{X}_m(\alpha_m n) \ge \bar{X}_j(\alpha_j n) \right) \right\}.
$$

Thus, for $i = 1, 2, \dots, m-1$,

$$\lim_{n\to\infty}\frac{1}{n}\ln P\{\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1}n)\} = -G_i(\alpha_i, \alpha_{i+1}) .$$

For $j = m+1,...,k$,

$$\lim_{n\to\infty}\frac{1}{n}\ln P\{\bar{X}_m(\alpha_m n) \geq \bar{X}_j(\alpha_j n)\} = -G_j(\alpha_j, \alpha_m) .$$

We then have

$$\lim_{n\to\infty}\frac{1}{n}\ln P(FR_m) = -\min\left\{\min_{1\leq i\leq m-1} G_i(\alpha_i, \alpha_{i+1}), \min_{j=m+1,...,k} G_j(\alpha_j, \alpha_m)\right\} \quad (4.4)$$

From Lemma 3.1, it can be concluded that

$$\lim_{n\to\infty}\frac{1}{n}\ln P\left(\bar{X}_i(\alpha_i n) \geq \bar{X}_{i+1}(\alpha_{i+1}n)\right) = -\inf_x\left(\alpha_i I_i(x) + \alpha_{i+1}I_{i+1}(x)\right)$$
$$\lim_{n\to\infty}\frac{1}{n}\ln P\left(\bar{X}_m(\alpha_m n) \geq \bar{X}_j(\alpha_j n)\right) = -\inf_x\left(\alpha_j I_j(x) + \alpha_m I_m(x)\right)$$

$$\begin{aligned}&\lim_{n\to\infty}\frac{1}{n}\ln P(FR_m)\\&= -\min\left\{\min_{1\leq i\leq m-1}\inf_x\left(\alpha_i I_i(x) + \alpha_{i+1}I_{i+1}(x)\right), \min_{j=m+1,...,k}\inf_x\left(\alpha_j I_j(x) + \alpha_m I_m(x)\right)\right\}.\square\end{aligned}$$

Our objective is to maximize the probability of correctly ranking top *m* designs. This can be achieved by minimizing the false ranking probability. It is also the same as maximizing the convergent rate of $P(FR_m)$ subject to $\sum_{i=1}^{k}\alpha_i = 1$ and $\alpha_i \geq 0, \forall i = 1,..,k$. The original OCBA optimization model is equivalent to the following:

$$\begin{aligned} &\max\ \min\left\{\min_{1\le i\le m-1}\inf_x(\alpha_i I_i(x)+\alpha_{i+1}I_{i+1}(x)),\ \min_{j=m+1,\dots,k}\inf_x\left(\alpha_j I_j(x)+\alpha_m I_m(x)\right)\right\} \\ &s.t.\quad \sum_{i=1}^{k}\alpha_i=1,\alpha_i\ge 0,\quad i\in\{1,..,k\} \end{aligned} \tag{4.5}$$

By Glynn and Juneja (2004), $\alpha_i I_i(x)+\alpha_{i+1}I_{i+1}(x)$ is a strictly increasing concave function. The infimum of concave functions is also concave. Likewise, the minimum of concave functions is a concave function too. Define $x(\alpha_i,\alpha_{i+1})=\arg\inf_x(\alpha_i I_i(x)+\alpha_{i+1}I_{i+1}(x))$. As shown in Glynn and Juneja (2004), $x(\alpha_i,\alpha_{i+1})$ is the solution to $\alpha_i I_i'(x)+\alpha_{i+1}I_{i+1}'(x)=0$ and

$$\partial\left(\alpha_i I_i\left(x(\alpha_i,\alpha_{i+1})\right)+\alpha_{i+1}I_{i+1}\left(x(\alpha_i,\alpha_{i+1})\right)\right)/\partial\alpha_i=I_i\left(x(\alpha_i,\alpha_{i+1})\right)$$

$$\partial\left(\alpha_i I_i\left(x(\alpha_i,\alpha_{i+1})\right)+\alpha_{i+1}I_{i+1}\left(x(\alpha_i,\alpha_{i+1})\right)\right)/\partial\alpha_{i+1}=I_{i+1}\left(x(\alpha_i,\alpha_{i+1})\right).$$

The result for $\alpha_j I_j(x)+\alpha_m I_m(x)$ follows similarly.

Therefore, the optimization model (4.5) is a concave maximization problem and it can be re-expressed as follows:

$$\begin{aligned} &\max\ z \\ s.t.\quad &\alpha_i I_i\left(x(\alpha_i,\alpha_{i+1})\right)+\alpha_{i+1}I_{i+1}\left(x(\alpha_i,\alpha_{i+1})\right)\ge z, i\in\{1,..,m-1\} \\ &\alpha_j I_j\left(x(\alpha_j,\alpha_m)\right)+\alpha_m I_m\left(x(\alpha_j,\alpha_m)\right)\ge z,\ j\in\{m+1,..,k\} \\ &\sum_{i=1}^{k}\alpha_i=1,\alpha_i\ge 0,i\in\{1,..,k\} \end{aligned} \tag{4.6}$$

Since model (4.6) is strictly concave and the functions of $\boldsymbol{\alpha}$ are continuous, a unique optimal solution must exist and the KKT conditions are necessary and sufficient for global optimality.

From the KKT conditions on problem (4.6), we define a new problem (4.7) by replacing some inequality signs and forcing α_i to be strictly positive.

$$\begin{aligned}
&\max\ z \\
s.t.\ \ &\alpha_1 I_1\left(x(\alpha_1,\alpha_2)\right)+\alpha_2 I_2\left(x(\alpha_1,\alpha_2)\right)=z \\
&\min\begin{Bmatrix}\alpha_i I_i\left(x(\alpha_i,\alpha_{i+1})\right)+\alpha_{i+1}I_{i+1}\left(x(\alpha_i,\alpha_{i+1})\right), \\ \alpha_{i-1}I_{i-1}\left(x(\alpha_{i-1},\alpha_i)\right)+\alpha_i I_i\left(x(\alpha_{i-1},\alpha_i)\right)\end{Bmatrix}=z, i\in\{2,..,m-1\} \qquad (4.7) \\
&\alpha_j I_j\left(x(\alpha_j,\alpha_m)\right)+\alpha_m I_m\left(x(\alpha_j,\alpha_m)\right)=z,\ j\in\{m+1,..,k\} \\
&\sum_{i=1}^{k}\alpha_i=1,\alpha_i>0, i\in\{1,..,k\}
\end{aligned}$$

Theorem 4.1 Under the assumptions 3.1 and 3.2 in Section 3.3, problems (4.6) and (4.7) are equivalent, i.e., a solution $\boldsymbol{\alpha}^*=(\alpha_1^*,...,\alpha_k^*)$ is the optimal solution to (4.6) if and only if it is also an optimal solution to (4.7).

Proof: We assume that a point satisfying the KKT condition of (4.6) is also feasible to (4.7). We first prove the forward and backward assertions. We then prove that the assumption that a point satisfying the KKT condition of (4.6) is also feasible to (4.7) is indeed correct.

(=>) Suppose $\boldsymbol{\alpha}^*$ is the optimal solution to (4.7). Since the feasible region of (4.7) is a subset of that of (4.6), if the optimal solution to (4.6) is feasible to (4.7), it must be optimal to (4.7). Since the KKT conditions are necessary and sufficient for optimality in (4.6) and it is feasible to (4.7), therefore, if a point satisfies the KKT condition in (4.6), it is must be optimal to (4.7).

(<=) Suppose the optimal solution to (4.6) is $\boldsymbol{\alpha}^*$ and the optimal solution to (4.7) is $\tilde{\boldsymbol{\alpha}}^*$, and $\boldsymbol{\alpha}^*\neq\tilde{\boldsymbol{\alpha}}^*$. Since the KKT conditions are necessary and sufficient condition to (4.6), thus $\boldsymbol{\alpha}^*$ must satisfy the KKT conditions. Furthermore, the objective function of (4.7) is the same as that of (4.6), and the feasible region of (4.7) is a subset of that of (4.6). Therefore, $\boldsymbol{\alpha}^*$ must be infeasible to (4.7). However, we assumed that a point satisfying the KKT conditions of (4.6) must be feasible to (4.7). We have thus reached a contradiction. So we must have $\boldsymbol{\alpha}^* = \tilde{\boldsymbol{\alpha}}^*$.

We are now in the position to prove that a point satisfying the KKT conditions of (4.6) must be feasible to (4.7). If we let $\alpha_i = 1/k$, we can have $z > 0$ for problem (4.6). However, any $\alpha_i = 0$ for $i = 1,...,m-1$ will lead to $\alpha_{i+1} \inf_x I_{i+1}(x) = \alpha_{i+1} I_{i+1}(\mu_{i+1}) = 0$. If $\alpha_j = 0$ for $j = m+1,...,k$, we will have $\alpha_m \inf_x I_m(x) = \alpha_m I_m(\mu_m) = 0$ .Therefore, the optimal solution must satisfy $\alpha_i > 0$ for every $i = 1,...,k.$

Since the problem (4.6) is a concave optimization problem, the first order condition is also the optimality condition. According to the KKT conditions, there exist $\lambda_i \ge 0, \lambda_j \ge 0, i \in \{1,..,m-1\},\ j \in \{m+1,..,k\}$ and $\gamma > 0$ such that,

$$\sum_{i=1}^{m-1} \lambda_i + \sum_{j=m+1}^{k} \lambda_j = 1 \tag{4.8}$$

$$\lambda_1 I_1 \left(x(\alpha_i^*, \alpha_{i+1}^*) \right) = \gamma \tag{4.9}$$

$$\lambda_{i-1} I_i \left(x(\alpha_{i-1}^*, \alpha_i^*) \right) + \lambda_i I_i \left(x(\alpha_i^*, \alpha_{i+1}^*) \right) = \gamma, i \in \{2,...,m-1\} \tag{4.10}$$

$$\lambda_{m-1} I_m\left(x(\alpha_{m-1}^*, \alpha_m^*)\right) + \sum_{j=m+1}^{k} \lambda_j I_m\left(x(\alpha_j^*, \alpha_m^*)\right) = \gamma \tag{4.11}$$

$$\lambda_j I_j\left(x(\alpha_j^*, \alpha_m^*)\right) = \gamma,\ j \in \{m+1, ..., k\} \tag{4.12}$$

$$\lambda_i\left(\alpha_i^* I_i\left(x(\alpha_i^*, \alpha_{i+1}^*)\right) + \alpha_{i+1}^* I_{i+1}\left(x(\alpha_i^*, \alpha_{i+1}^*)\right) - z\right) = 0, i \in \{1, ..., m-1\} \tag{4.13}$$

$$\lambda_j\left(\alpha_j^* I_j\left(x(\alpha_j^*, \alpha_m^*)\right) + \alpha_m^* I_m\left(x(\alpha_j^*, \alpha_m^*)\right) - z\right) = 0,\ j \in \{m+1, ..., k\} \tag{4.14}$$

If $\lambda_1 = 0$, γ will be zero. Thus, we could conclude that all $\lambda_i, \lambda_j = 0, j = m+1, ..., k$ by putting $\gamma = 0$ into equations (4.9) to (4.12). Substituting $\lambda_1 = 0$ and $\gamma = 0$ into equations (4.10) and (4.11) will result in $\lambda_i = 0, i = 2, ..., m-1$. However, this contradicts with equation (4.8) which requires at least one $\lambda_i, \lambda_j > 0$. Thus, we conclude that $\lambda_1 > 0$ and $\gamma > 0$ because $I_i\left(x(\alpha_i^*, \alpha_{i+1}^*)\right), i = 1, .., k-1$ is strictly positive. Similarly, we could conclude that $\lambda_j > 0, j = m+1, ..., k$ from equation (4.12) and $\max\{\lambda_i, \lambda_{i+1}\} > 0,\ i = 1, ..., m-1$ from equation (4.10).

Based on the results that $\lambda_1 > 0$, $\lambda_j > 0, j \in \{m+1, ..., k\}$, $\max\{\lambda_i, \lambda_{i+1}\} > 0$, $i = 2, ..., m-1$ and constraints of (4.6), we have the following equality from the complementary slackness condition in equations (4.13) and (4.14). For $i \in \{2, ..., m-1\}$, $j \in \{m+1, ..., k\}$,

$$
\begin{aligned}
z &= \alpha_1^* I_1\left(x(\alpha_1^*,\alpha_2^*)\right)+\alpha_2^* I_2\left(x(\alpha_1^*,\alpha_2^*)\right)\\
&= \alpha_j^* I_j\left(x(\alpha_j^*,\alpha_m^*)\right)+\alpha_m^* I_m\left(x(\alpha_j^*,\alpha_m^*)\right)\\
&= \min\left\{\begin{matrix}\alpha_i^* I_i\left(x(\alpha_i^*,\alpha_{i+1}^*)\right)+\alpha_{i+1}^* I_{i+1}\left(x(\alpha_i^*,\alpha_{i+1}^*)\right),\\ \alpha_{i-1}^* I_{i-1}\left(x(\alpha_{i-1}^*,\alpha_i^*)\right)+\alpha_i^* I_i\left(x(\alpha_{i-1}^*,\alpha_i^*)\right)\end{matrix}\right\}
\end{aligned}
\tag{4.15}
$$

So we have proved the assertion that a point satisfying the KKT conditions of (4.6) must be feasible to (4.7).□

Therefore, we could conclude that the optimal allocation rule to rank the top *m* designs $\boldsymbol{\alpha}^* = (\alpha_1^*,...,\alpha_k^*)$ solves the equations E1 and E2 below.

$$
\begin{aligned}
&\mathbf{E1}: \sum_{i=1}^{k}\alpha_i = 1, \alpha_i > 0,\\
&\mathbf{E2}: \min\left\{\begin{matrix}\alpha_i^* I_i\left(x(\alpha_i^*,\alpha_{i+1}^*)\right)+\alpha_{i+1}^* I_{i+1}\left(x(\alpha_i^*,\alpha_{i+1}^*)\right),\\ \alpha_{i-1}^* I_{i-1}\left(x(\alpha_{i-1}^*,\alpha_i^*)\right)+\alpha_i^* I_i\left(x(\alpha_{i-1}^*,\alpha_i^*)\right)\end{matrix}\right\}\\
&\quad= \alpha_1^* I_1\left(x(\alpha_1^*,\alpha_2^*)\right)+\alpha_2^* I_2\left(x(\alpha_1^*,\alpha_2^*)\right)\\
&\quad= \alpha_j^* I_j\left(x(\alpha_j^*,\alpha_m^*)\right)+\alpha_m^* I_m\left(x(\alpha_j^*,\alpha_m^*)\right),\ i = 2,...,m-1; j = m+1,...,k
\end{aligned}
$$

Suppose the performance of each design follows a normal distribution $X_i \sim N(\mu_i, \sigma_i^2),\ i = 1,...,k.$ Equations E1 and E2 can be re-written as follows in the case of normally distributed performance:

$$
\begin{aligned}
&\mathbf{E3}: \sum_{i=1}^{k}\alpha_i = 1, \alpha_i > 0,\\
&\mathbf{E4}: \min\left\{\frac{(\mu_i-\mu_{i+1})^2}{2(\sigma_i^2/\alpha_i^*+\sigma_{i+1}^2/\alpha_{i+1}^*)}, \frac{(\mu_i-\mu_{i-1})^2}{2(\sigma_i^2/\alpha_i^*+\sigma_{i-1}^2/\alpha_{i-1}^*)}\right\}\\
&\quad= \frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2/\alpha_1^*+\sigma_2^2/\alpha_2^*)}\\
&\quad= \frac{(\mu_j-\mu_m)^2}{2(\sigma_j^2/\alpha_j^*+\sigma_m^2/\alpha_m^*)},\quad i = 2,...,m-1; j = m+1,...,k.
\end{aligned}
$$

4.4. Approximated Asymptotically Optimal Allocation

We have derived the asymptotically optimal allocation strategy in Section 4.3 above; however, the optimal allocation involves solving the nonlinear equations each time. In this section, we first propose an upper bound of the probability of false ranking. We derive the rate function of this upper bound probability, and obtain the optimal allocation strategy by maximizing this convergent rate at which the false ranking probability goes to zero. A simple closed-form allocation rule can be obtained based on this upper bound of the false ranking probability.

Recall from the previous section that the probability of false ranking is defined as follows:

$$
\begin{aligned}
&P(FR_m) \\
&= P\left\{ \left(\bigcup_{i=1}^{m-1} \left(\bar{X}_i(\alpha_i n) \ge \bar{X}_{i+1}(\alpha_{i+1} n) \right) \right) \bigcup \left(\bigcup_{j=m+1}^{k} \left(\bar{X}_m(\alpha_m n) \ge \bar{X}_j(\alpha_j n) \right) \right) \right\}
\end{aligned} \tag{4.16}
$$

Define a strictly increasing sequence $\{c_i, i = 0,1,...,m+1\}$ such that $c_0 < ... < c_i < ... < c_{m+1}$ with $c_i = \mu_i, i = 1,...,m, c_0 = -\infty$, where μ_i is the mean performance of design i. We could approximate $P(FR_m)$ as

$$
\begin{aligned}
&P(FR_m) \\
&\le P\left\{ \left(\bigcup_{i=1,...,m} \left(\left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \cup \left(\bar{X}_i(\alpha_i n) \ge c_{i+1} \right) \right) \right) \bigcup \left(\bigcup_{j=m+1,...,k} \left(\bar{X}_j(\alpha_j n) \le c_m \right) \right) \right\} \\
&= \sum_{i=1}^{m} P\left\{ \left(\bar{X}_i(\alpha_i n) \le c_{i-1} \right) \cup \left(\bar{X}_i(\alpha_i n) \ge c_{i+1} \right) \right\} + \sum_{j=m+1}^{k} P\left\{ \bar{X}_j(\alpha_i n) \le c_m \right\} \\
&= P(AFR_m)
\end{aligned}
$$

where the second equality follows from the fact that every design is mutually independent .

To derive the large deviation rate function for $P(AFR_m)$, we should first prove that the large deviation principle is satisfied. It is easy to see that $P(AFR_m)$ is bounded below by

$$\max\left\{\max_{1\le i\le m} P\left\{\left(\bar{X}_i(\alpha_i n)\le c_{i-1}\right)\bigcup\left(\bar{X}_i(\alpha_i n)\ge c_{i+1}\right)\right\},\ \max_{j=m+1,\dots,k} P\left\{\bar{X}_j(\alpha_j n)\le c_m\right\}\right\}$$

and bounded above by

$$k*\max\left\{\max_{1\le i\le m} P\left\{\left(\bar{X}_i(\alpha_i n)\le c_{i-1}\right)\bigcup\left(\bar{X}_i(\alpha_i n)\ge c_{i+1}\right)\right\},\ \max_{j=m+1,\dots,k} P\left\{\bar{X}_j(\alpha_j n)\le c_m\right\}\right\}.$$

Therefore, for $i=1,\dots,m$,

$$\lim_{n\to\infty}\frac{1}{n}\ln P\{(\bar{X}_i(\alpha_i n)\le c_{i-1})\}=-R_1(\alpha_i,c_{i-1})$$

$$\lim_{n\to\infty}\frac{1}{n}\ln P\{(\bar{X}_i(\alpha_i n)\ge c_{i+1})\}=-R_2(\alpha_i,c_{i+1})$$

and for $j=m+1,\dots,k$,

$$\lim_{n\to\infty}\frac{1}{n}\ln P\{(\bar{X}_j(\alpha_j n)\le c_m)\}=-R_1(\alpha_j,c_m)$$

for some rate functions $R_1(\alpha_i,c_{i-1})$, $R_2(\alpha_i,c_{i+1})$ and $R_1(\alpha_j,c_m)$. Then, by the principle of largest term (Ganesh et al., 2004),

$$\begin{aligned}
&\lim_{n\to\infty}\frac{1}{n}\ln P\{(\bar{X}_i(\alpha_i n)\le c_{i-1})\bigcup(\bar{X}_i(\alpha_i n)\ge c_{i+1})\}\\
&=\lim_{n\to\infty}\frac{1}{n}\ln\{P(\bar{X}_i(\alpha_i n)\le c_{i-1})+P(\bar{X}_i(\alpha_i n)\ge c_{i+1})\}\\
&=-\min\{R_1(\alpha_i,c_{i-1}),R_2(\alpha_i,c_{i+1})\}
\end{aligned}$$

Similarly, by the principle of largest term, the rate function of $P(AFR_m)$ can be denoted as

$$\begin{aligned}
&\lim_{n\to\infty}\frac{1}{n}\ln P(AFR_m)\\
&=\sum\nolimits_{i=1}^{m}P\left\{\left(\bar{X}_i(\alpha_i n)\le c_{i-1}\right)\bigcup\left(\bar{X}_i(\alpha_i n)\ge c_{i+1}\right)\right\}+\sum\nolimits_{j=m+1}^{k}P\left\{\bar{X}_j(\alpha_j n)\le c_m\right\}\\
&=-\min\left\{\min_{1\le i\le m}\left\{\min\left(R_1(\alpha_i,c_{i-1}),R_2(\alpha_i,c_{i+1})\right)\right\},\min_{j=m+1,\dots,k}R_1(\alpha_j,c_m)\right\}
\end{aligned}$$

From Lemma 3.2, the rate function for the approximated probability of false ranking is given by

$$\begin{aligned}
&-\lim_{n\to\infty}\frac{1}{n}\ln P(AFR_m)\\
&=\min_i(\min\{R_1(\alpha_i,c_{i-1}),R_2(\alpha_i,c_{i+1})\})\\
&=\min\left\{\min_{1\le i\le m}\left\{\min\left(\alpha_i I_i(c_{i-1}),\alpha_i I_i(c_{i+1})\right)\right\},\min_{j=m+1,\dots,k}\alpha_j I_j(c_m)\right\}
\end{aligned}\qquad(4.17)$$

Our objective is to maximize the probability of correctly ranking the top *m* designs. This can be achieved by minimizing the upper bound of the false ranking probability. It is also the same as maximizing the convergent rate at which $P(AFR_m)$ goes to zero, i.e.,

$$\begin{aligned}
&\max\min\left\{\min_{1\le i\le m}\left\{\min\left(\alpha_i I_i(c_{i-1}),\alpha_i I_i(c_{i+1})\right)\right\},\min_{j=m+1,\dots,k}\alpha_j I_j(c_m)\right\}\\
&s.t.\ \sum_{i=1}^{k}\alpha_i=1,\alpha_i\ge 0,\forall i=1,..,k
\end{aligned}$$

This can be re-expressed as follows:

$$\begin{aligned} &\max \quad z \\ s.t.\ \ &\min\left\{\min_{1\le i\le m}\left\{\min\left(\alpha_i I_i(c_{i-1}), \alpha_i I_i(c_{i+1})\right)\right\}, \min_{j=m+1,\dots,k} \alpha_j I_j(c_m)\right\} - z \ge 0 \\ &\sum_{i=1}^{k} \alpha_i = 1, \alpha_i \ge 0 \quad (1 \le i \le k) \end{aligned} \tag{4.18}$$

Theorem 4.2 Under assumptions 3.1 and 3.2 in Section 3.3, if the optimal allocation $\boldsymbol{\alpha}^* > 0, \sum_{i=1}^{k} \alpha_i^* = 1$ minimizes the approximated probability of false ranking asymptotically, then

$$\begin{aligned} &\alpha_p^* \min\left(I_p(c_{p-1}), I_p(c_{p+1})\right) = \alpha_q^* \min\left(I_q(c_{q-1}), I_q(c_{q+1})\right) \\ &= \alpha_j^* I_j(c_m), \text{where } p, q \in \{1,\dots,m\}, \ j \in \{m+1,\dots,k\}. \end{aligned}$$

Proof: We first re-write the optimization model (4.18) as follows:

$$\begin{aligned} &\max \quad z \\ s.t.\ \ &\min\left(\alpha_i I_i(c_{i-1}), \alpha_i I_i(c_{i+1})\right) - z \ge 0, i = 1,\dots,m \\ &\alpha_j I_j(c_m) - z \ge 0, \ j = m+1,\dots.,k \\ &\sum_{i=1}^{k} \alpha_i = 1, \alpha_i \ge 0 \quad (1 \le i \le k) \end{aligned} \tag{4.19}$$

By Glynn and Juneja (2004), we know that (4.19) is a concave programming problem, and hence the first order condition is also the optimality condition. Therefore, under the KKT conditions, there exist λ_i and $\gamma > 0$ such that,

$$1 - \sum_{i=1}^{k} \lambda_i = 0 \tag{4.20}$$

$$\lambda_i \min(I_i(c_{i-1}), I_i(c_{i+1})) = \gamma, \forall i = 1,..,m \tag{4.21}$$

$$\lambda_j I_j(c_m) = \gamma, \forall j = m+1,...,k \tag{4.22}$$

$$\lambda_i[z - \alpha^*_i \min(I_i(c_{i-1}), I_i(c_{i+1}))] = 0, 1 \le i \le m \tag{4.23}$$

$$\lambda_j[z - \alpha^*_j I_j(c_m)] = 0, j = m+1,....,k \tag{4.24}$$

$$\lambda_i \ge 0, 1 \le i \le k \tag{4.25}$$

From equation (4.20) we know that λ_i must be positive for some $i, i = 1,...,k$. However, the rate function $I_i(c_i), i = 1,...,k$ is always positive. Therefore, any $\lambda_i = 0$ will lead to all other $\lambda_i = 0$. Therefore, we have $\lambda_i > 0$ for all *i*. As a result of equation (4.23), we conclude that $z = \alpha^*_p \min\left(I_p(c_{p-1}), I_p(c_{p+1})\right) = \alpha^*_j I_j(c_m) = \alpha^*_q \min\left(I_q(c_{q-1}), I_q(c_{q+1})\right)$, where $p, q \in [1,...,m], \ j \in [m+1,...,k]$.□

In the case of normal distribution, the performance of design *i* is then denoted as $X_i \sim N(\mu_i, \sigma_i^2)$. Since $c_i = \mu_i, i = 1,...,m, c_0 = -\infty$ and $I_i(x_i) = (\mu_i - x_i)^2 / 2\sigma_i^2$, we have $I_j(c_m) = (\mu_j - \mu_m)^2 / 2\sigma_j^2$. As a result,

$$\min(I_p(c_{p-1}), I_p(c_{p+1})) = \min\left\{\left(\mu_p - \mu_{p-1}\right)^2, \left(\mu_p - \mu_{p-1}\right)^2\right\} / 2\sigma_p^2$$

and the optimal allocation is such that,

$$\begin{cases} \dfrac{\alpha^*_p}{\alpha^*_q} = \dfrac{\sigma_p^2 / \min\{(\mu_p - \mu_{p+1})^2, (\mu_p - \mu_{p-1})^2\}}{\sigma_q^2 / \min\{(\mu_q - \mu_{q+1})^2, (\mu_q - \mu_{q-1})^2\}} \\ \dfrac{\alpha^*_q}{\alpha^*_j} = \dfrac{\sigma_q^2 / \min\{(\mu_q - \mu_{q+1})^2, (\mu_q - \mu_{q-1})^2\}}{\sigma_j^2 / (\mu_j - \mu_m)^2} \end{cases} \tag{4.26}$$

where $p, q \in [1,...,m], j \in [m+1,...,k]$

4.5. Sequential Allocation and Consistent Estimator

In order to implement the proposed algorithm, we propose a similar sequential allocation algorithm as in Section 3.6. The proof of the estimator being consistent will also be similar to that in Section 3.7. Therefore, only a short summary of the results will be provided in this section.

Step 0: Perform n_0 simulation replications for every design.

$l \leftarrow 0, N_1^l = ... = N_k^l = n_0$

Step 1: If $\sum_{i=1}^{k} N_i^l \geq n$, stop.

Step 2: Increase the computation budget by Δ and compute the new budget allocation using E1 and E2 or Theorem 4.2.

Step 3: Perform additional $\max(0, N_i^{l+1} - N_i^l)$ simulation runs for design $i, i = 1, .., k$.

Theorem 4.3 The empirical estimation of the optimal allocation is consistent, i.e.,

$$\alpha_i^*(n) \rightarrow \alpha_i^*, \forall i = 1, .., k \text{ as } n \rightarrow +\infty \text{ almost surely.}$$

Proof: See the proof of Theorem 3.3.□

4.6. Numerical Experiments

In this section, we will conduct several numerical experiments by comparing our proposed algorithm with different allocation procedures. The performance of every design is assumed to be normally distributed in all experiments. Therefore, the allocation rule derived in Section 4.3 can be obtained by solving equations E3 and E4. The allocation rule from Section 4.4 is obtained through equation (4.26).

4.6.1. Probability of Correct Ranking

The first set of numerical experiments we will conduct is the comparison of the empirical probability of correct ranking for different allocation procedures. We will describe the different allocation procedures as follows.

Equal Allocation (EA): The simulation budget allocated is such that every design has an equal number of replications, i.e., $\alpha_i = 1/k, i = 1,...,k$. This is the simplest allocation rule and it can serve as a benchmark for other allocation procedures.

Asymptotically Optimal Allocation (AOA-m): This is the allocation rule we obtain by solving the equations E3 and E4 in Section 4.3. This allocation rule optimizes the convergent rate of false ranking probability asymptotically.

Approximated Allocation (AA-m): This is the closed-form allocation rule we derived in Section 4.4.

Table 4.1. Mean and variance of three experiments in chapter 4

	Equal Spacing		Equal Variance		Increasing Spacing Decresing Variance	
Design	Mean	Variance	Mean	Variance	Mean	Variance
I	1	400	1	100	1	400
II	2	361	2	100	2	361
III	3	324	4	100	4	324
IV	4	289	7	100	7	289
V	5	256	11	100	11	256
VI	6	225	16	100	16	225
VII	7	196	22	100	22	196
VIII	8	169	29	100	29	169
IX	9	144	37	100	37	144
X	10	121	46	100	46	121
XI	11	100	56	100	56	100
XII	12	81	67	100	67	81
XIII	13	64	79	100	79	64
XIV	14	49	92	100	92	49
XV	15	36	106	100	106	36
XVI	16	25	121	100	121	25
XVII	17	16	137	100	137	16
XVIII	18	9	154	100	154	9
XIX	19	4	172	100	172	4
XX	20	1	191	100	191	1

The experimental setting is summarized in Table 4.1. The objective of this experiment is to rank the top 5 designs out of 20 alternatives. Three different scenarios are tested as follows: (a) Equal spacing scenario refers to the situation when the mean differences between consecutive designs are the same but the variance of each design is different. (b) Equal variance scenario refers to the situation when the variance of each design is the same but the mean differences between consecutive designs are different. (c) increasing spacing but decreasing variance scenario refers to the situation when the variance of each design is different and the mean differences between consecutive designs are also different.

The experiments are conducted in two ways. Firstly, we assume that the mean and variance of each design are known. Secondly, we assume the mean and variance of each design is unknown. The simulation procedure is summarized below for each method.

Known Mean and Variance:

Step 0: Perform n_0 simulation replications for all designs.

Step 1: Determine the budget allocation rules for AA-m using equation (4.26). Determine the budget allocation rule for AOA-m by solving E3 and E4. The number of simulation replications for design i is N_i, $i = 1, ..., k$.

Step 2: Perform additional $\max(0, N_i - n_0)$ simulation runs for design $i, i = 1, .., k$.

Unknown Mean and Variance:

Step 0: Perform n_0 simulation replications for all designs. $l \leftarrow 0$, $N_1^l = N_k^l = ... = N_k^l = n_0$.

Step 1: If $\sum_{i=1}^{k} N_i^l \geq n$, stop.

Step 2: Increase the computation budget by Δ and compute the new budget allocation for AA-m using equation (4.26), and determine the budget allocation rule for AOA-m by solving E3 and E4, where the sample mean and sample variance are used as an estimation of the population mean and population variance.

Step 3: Perform additional $\max(0, N_i^{l+1} - N_i^l)$ simulation runs for design

$i, i = 1,..,k$, $l \leftarrow l + 1$. Go back to step 1.

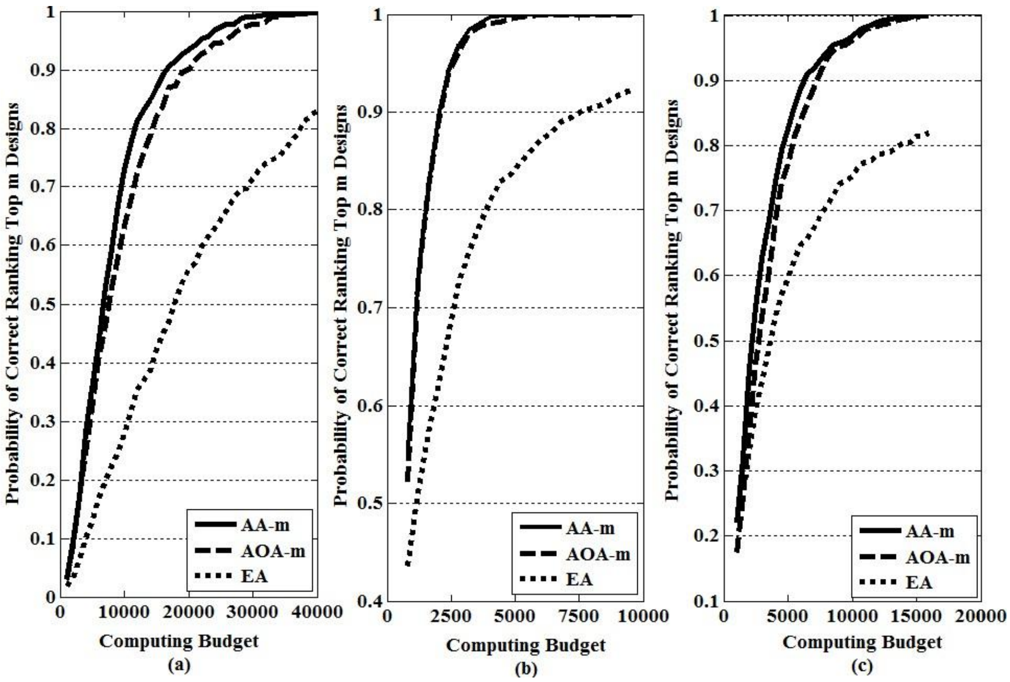

Fig.4.1. Comparison of correct ranking probability of top *m* designs for AA-m, AOA-m and EA with expected mean and variance. (a) is for equal spacing scenario; (b) is for equal variance scenario; (c) is for increasing spacing but decreasing variance scenario.

The probability of correct ranking is estimated as the number of times correct ranking occurs out of the total number of simulation runs we have conducted. We conducted 10,000 simulation runs for each experiment. Figure 4.1 summarizes the results of the experiments in the case of known mean and variance, while Figure 4.2 shows the results in the case of unknown mean and variance.

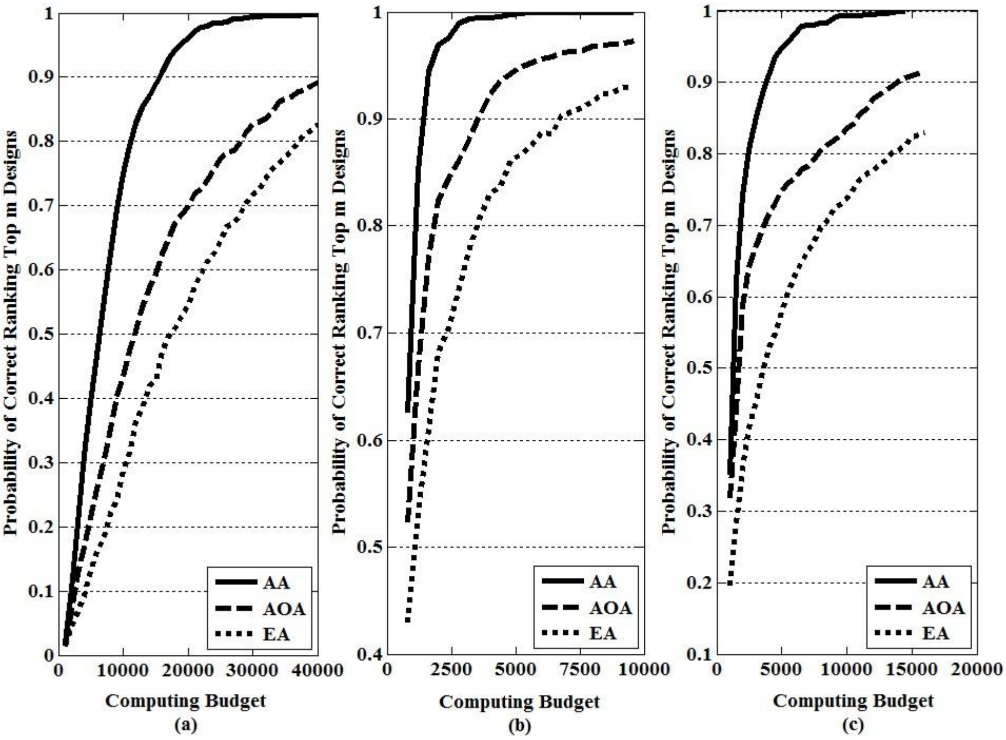

Fig.4.2. Comparison of correct ranking probability of top *m* designs for AA-m, AOA-m and EA under sequential allocation strategy. (a) is for equal spacing scenario; (b) is for equal variance scenario; (c) is for increasing spacing but decreasing variance scenario.

Given known mean and variance, AA-m performs the best among the three allocation rules in all three scenarios in terms of the probability of correct ranking in finite budget. However, AOA-m catches up with AA-m quickly when the total computing budget becomes large. In addition, the performance difference of AA-m and AOA-m is very small.

On the other hand, AA-m performs the best among the three allocation rules in all three scenarios in terms of the probability of correct ranking when the simulation budget is sequentially allocated. However, the performance difference of AA-m and AOA-m is much larger than when the mean and variance are given.

AOA-m rule is derived by optimizing the convergent rate of false ranking probability. Thus, it must be the rule which yields the largest convergent rate compared with AA-m and EA. However, it may not be the best allocation rule when the probability of correct ranking is used as the performance measure.

The convergent rate of false ranking probability, which can be computed by substituting the value of $\boldsymbol{\alpha}$ into Lemma 4.1, is well defined if we know the parameters of the underlying distribution. In the case of the normal distribution, the parameters are the mean and variance. However, as the mean and variance are unknown, they can only be accurately estimated when the number of simulation replications is infinite for every design. In the experiments, the simulation replications are allocated sequentially. At an earlier stage, the estimation of the true mean and variance can be poor because of small budget. The AOA-m rule involves solving the nonlinear programming problem in each iteration. This process of solving the nonlinear programming problem brings instability into the process. A small change of the mean or variance can bring about a large change of the allocation, i.e., the value of $\boldsymbol{\alpha}$, by using the nonlinear programming solver. However, the change is minimal in the AA-m rule. This is one of the reasons why AOA-m is consistently worse in performance than AA-m in finite budget. However, when the simulation budget increases, AOA-m can eventually catch up with AA-m.

On the other hand, the probability of correct ranking estimated from simulation depends heavily on the number of samples simulated for each design. When the total computing budget is small, designs with small percentage of allocation (non-critical designs) will receive extremely small number of replications. Although non-critical deigns are easy to distinguish, too few replications makes them variable and hard to distinguish. However, if a few more replications are allocated to non-critical designs, they can be distinguished very easily. In other words, we can conclude that the marginal increase of correct ranking probability will be much larger if we allocate a few more replications to non-critical designs when the total budget is small. This is why AA-m is not the asymptotically optimal allocation rule, but it performs better in finite budget especially when computing budget is small. When the simulation budget becomes relatively large, we could see that the performance of AOA-m and AA-m is very close to each other in the numerical experiments with given mean and variance.

To summarize, our proposed allocation rule in Theorem 4.2 performs best under finite budget when the probability of correct ranking is used as the performance measure although we made some approximation to the probability of false ranking and selection. Moreover, our objective is to provide a simple allocation rule that can be used easily and efficiently. Therefore, we will use the AA-m rule when integrating with genetic algorithms in Section 4.6.2 below.

4.6.2. Numerical Experiments for Simulation Optimization

In the following numerical examples, we will integrate AA-m with the genetic algorithm to solve the simulation optimization problem under noisy environment. Genetic algorithm is a search heuristic for global optimization problems. The selection scheme is the key step of the genetic algorithm. Blickle and Thiele (1995) summarized the different selection schemes used in genetic algorithms. In general, the selection schemes are based either on the performance value of the candidates or on the ranking of the performance of the candidates. In our numerical experiments, we use the ranking information as the selection criterion and show how our proposed budget allocation rule can enhance the search efficiency for the genetic algorithm.

The ranking-based selection scheme we use will give more opportunity to higher ranking candidates to be selected as parents to produce the children. The three allocation rules we will compare are AA-m, EA and OCBA-m (Chen et al., 2008). OCBA-m is the allocation rule that maximizes the probability of correctly selecting the top *m* designs, but it does not aim to distinguish the relative ranking among the top *m* designs.

In simulation optimization, evaluating a solution is subject to noise. A large number of replications are needed in order to have a steady mean performance value. In every iteration of the genetic algorithm, we will simulate a large number of samples for each candidate solution and rank them according to their mean performance values. EA will simply allocate the available simulation budget equally to each candidate, while AA-m and OCBA-m will sequentially allocate the simulation budget based on their respective allocation rules.

A general framework of the genetic algorithm in stochastic simulation optimization can be written as follows:

Step 1: Initialize a population.

Step 2: Use sequential allocation algorithm to find the ranking of the top *m* elite solutions.

Step 3: Reproduce next generation based on the ranking information from Step 2.

Step 4: Mutate the solution to avoid trapping in the local optima.

Step 5: Repeat Steps 2-4 until the termination condition is met.

Using the proposed genetic algorithm framework, we will conduct three numerical experiments for three well-known continuous deterministic optimization problems. The noise for all experiments is assumed to be normally distributed with mean 0 and standard deviation of 50. The population size of the GA is set to be 20, and the top 10 solutions will be ranked as they will be selected as the parents to produce the offspring. Exponential ranking selection scheme is used in the numerical experiments. For ranked solution 1 to solution 10, the probability of being selected to produce offspring is set to be $p_i = (c-1)c^{10-i}/(c^{10}-1)$. The parameter c is set to be 0.7 in all experiments. 1000 simulation replications are available for each iteration, and GA terminates when the total number of iterations reaches 1000.

Experiment 1: Goldstein-Price function

$$\begin{aligned} S(X) = &\left(1+(x_1+x_2+1)^2(19-14x_1+3x_2^2-14x_2+6x_1x_2+3x_2^2)\right) \\ &*\left(30+(2x_1-3x_2)^2(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)\right) \end{aligned}$$

where $X=(x_1,x_2), -3 \le x_i \le 3, i=1,2$.

Goldstein-Price function has a unique optimal solution at (0,-1) with the objective value of 3. However, 4 local optima exist in the given feasible region.

Experiment 2: Griewank function

$$S(X) = \frac{1}{40}(x_1^2 + x_2^2) - \cos(x_1)\cos(\frac{x_2}{\sqrt{2}}) + 2$$

where $X = (x_1, x_2), -10 \le x_i \le 10, i = 1, 2$.

The unique optimal solution is at (0, 0) with objective value of 1. Many local optima exist in the given region.

Experiment 3: Spherical function

$$S(X) = \sum_{i=1}^{5}(X_i^2 - c)$$

where $c = 5, -5 \le x_i \le 15, i = 1, 2, 3, 4, 5$.

The value of *c* can be arbitrary. We use *c* = 5 in our experiment. This function has the optimal value of zero with $X = (5, 5, 5, 5, 5)$.

Figure 4.3 summarizes the numerical results of using AA-m, OCBA-m and EA in the selection process of the genetic algorithm respectively for Goldstein Price Function, Giewank Function and Spherical Function.

It is easy to see that the number of iterations can be reduced significantly by using the AA-m procedure in the selection process. As defined earlier, a simulation budget of 1000 is used for one iteration. Therefore, the total amount of budget saved is extremely large.

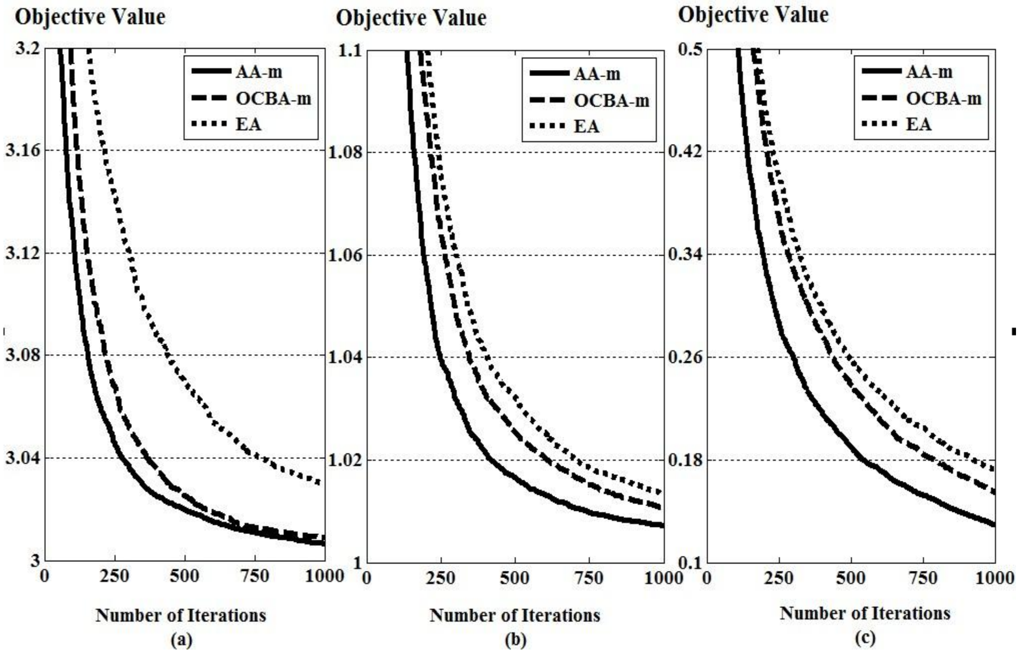

Fig.4.3. Numerical results of simulation optimization using genetic algorithm integrated with simulation budget allocation rules: AA-m, OCBA-m and EA. (a) is for Goldstein price function; (b) is for Giewank function; (c) is for Spherical function.

4.7. Conclusion

Motivated by the idea of integrating statistical ranking procedure into genetic algorithms, we have proposed an optimal computing budget allocation strategy of ranking top *m* designs out of *k* alternatives in this chapter. Based on the large deviation framework, we have derived the optimal allocation rule. Together with the heuristic sequential allocation algorithm, our method can be used to rank top *m* designs when the performance of the design can only be estimated from simulation. The proposed method is integrated with genetic algorithms and used to solve simulation optimization problems. The numerical experiments have shown that significant simulation budget can be saved by using our proposed method, and thus the searching efficiency is enhanced by integrating our simulation budget allocation rule with genetic algorithms.

Chapter 5. Simulation Optimization Using Regression in Partitioned Domains

We consider the problem of determining the simulation budget allocation when the simulation output can be modeled by quadratic equations. The budget allocation rule proposed in Brantley et al. (2013b) is heuristic and non-optimal. We will reformulate this budget allocation prolem based on large deviation theory and present its optimal characterization. We further analyze the limiting behaviour of the allocation rule. The rest of the chapter is organized as follows. Section 5.1 provides an overview of the whole chapter. In Section 5.2, we provide the detailed description and mathematical formulation of the problem in consideration. A Bayesian regression framework is used to estimate the performance distribution in Section 5.3. In Section 5.4, we show that only three points are needed to be simulated in order to obtain a quadratic line. We characterize the optimal allocation in Section 5.5. The limiting behavior of the allocation rule is discussed in Section 5.6. We present a sequential allocation algorithm for implementation in Section 5.7. In Section 5.8, we conduct the numerical experiments to compare our proposed allocation rule with some of the existing allocation rules. Finally, we conclude this chapter in Section 5.9.

5.1. Overview

The optimal simulation design (OSD) method proposed by Brantley et al. (2013a) has been shown to be an efficient simulation budget allocation rule when the underlying performance structure of all design points is quadratic or approximately quadratic. OSD assumes a common quadratic equation for all design locations and a common normally distributed noise across the entire domain. It is natural to think that the two assumptions in OSD may not be satisfied. Brantley et al. (2013b) further developed a heuristic allocation rule when the entire domain is divided into many partitions.

We will approach a similar problem differently and focus more on the theoretical derivation of the allocation rule. Firstly, our derivation is based on the large deviation rate of false selection probability, which is the speed at which the false selection probability goes to zero. Secondly, we provide the optimal characterization of the allocation strategy while Brantley et al. (2013b) only gives a heuristic way to obtain the allocation rule. Lastly, we analyze the limiting allocation rule when the number of partitions goes to infinity in order to obtain an easily implementable budget allocation. The resulted simulation budget allocation rule is very intuitive. The cross partition allocation rule is similar to the original OCBA rule (Chen et al., 2000) and the allocation rule within a partition is simply the OSD for the best partition which contained the best design location and a feasibility check problem with quadratic regression for other partitions.

5.2. Problem Formulation

We assume that the entire domain can be divided into m partitions, and there are k design locations in each partition, i.e., there are mk design locations in total. Without loss of generality, we assume that this is a minimization problem, and our objective is to find the design location which has the minimum performance value among the mk locations. Let $y(x_{hi})$ denote the performance at design location x_{hi}. This minimization problem can be mathematically written as follows:

$$\min_{x_{hi}} y(x_{hi}) = E[f(x_{hi})]; h = 1,...,m; i = 1,...,k . \tag{5.1}$$

The performance value $y(x_{hi})$ is unknown, and it can only be estimated with noise via simulation. Suppose that the performance of every design location in each partition can be modeled as a quadratic regression equation. The performance at design location x_{hi} can be written as follows:

$$y(x_{hi}) = W_{h0} + W_{h1}x_{hi} + W_{h2}x_{hi}^2; h = 1,...,m; i = 1,...,k . \tag{5.2}$$

Define a vector $\mathbf{W_h} = [W_{h0}\ W_{h1}\ W_{h2}], h = 1,...,m$. $\mathbf{W_h}$ provides the coefficients of the quadratic function. It can only be estimated based on the simulation output. The performance value of each design location can be estimated by simulation output with a normally distributed noise ε_h. The normally distributed noise is the same for every design location in the same partition. Let $f(x_{hi})$ denote the simulation output at design location x_{hi}. It can be represented as follows:

$$f(x_{hi}) = y(x_{hi}) + \varepsilon_h; \varepsilon_h \sim N(0, \sigma_h^2) . \tag{5.3}$$

Although both $\mathbf{W_h}$ and $y(x_{hi})$ are unknown, $y(x_{hi})$ can be estimated by the simulation output $f(x_{hi})$ and $\mathbf{W_h}$ can be derived using the least squares estimation method. In order to avoid singularity, we must choose at least three design locations in each partition. Let $\mathbf{\hat{W}_h} = [\hat{W}_{h0}\ \hat{W}_{h1}\ \hat{W}_{h2}]$ denote the estimation of $\mathbf{W_h}$ based on the simulation output. Let $\hat{y}(x_{hi})$ denote the estimated value of $y(x_{hi})$ based on the estimated quadratic function. Thus, the estimated quadratic regression formula can be written as the following expression:

$$\hat{y}(x_{hi}) = \hat{W}_{h0} + \hat{W}_{h1}x_{hi} + \hat{W}_{h2}x_{hi}^2; h = 1,...,m; i = 1,...,k . \tag{5.4}$$

Define $\mathbf{F_h}$ to be a vector with n entries of simulation output $f(x_{hi})$. Define $\mathbf{X_h}$ to be an $n \times 3$ matrix with each row $[1\ x_{hi}\ x_{hi}^2]$ corresponding to each entry $f(x_{hi})$ of $\mathbf{F_h}$. As shown in many design of experiment (DOE) literature, the least squares method minimizes the sum of the squared errors $\omega_h = \mathbf{(F_h - X_h W_h)^t (F_h - X_h W_h)}$, where the superscript *t* refers to the transpose of the matrix.

To derive $\mathbf{W_h}$, we expand ω_h as $\omega_h = \mathbf{F_h{}^t F_h - 2W_h X_h^t F_h - W_h X_h^t X_h W_h}$. Differentiating ω_h with respect to $\mathbf{W_h}$ results in $\partial \omega_h / \partial \mathbf{W_h} = \mathbf{X_h^t F_h - X_h^t X_h W_h}$. The minimum value of ω_h is achieved when the partial derivative is equal to 0, i.e., $\mathbf{X_h^t F_h - X_h^t X_h W_h} = 0$. We could solve for the estimated value of $\mathbf{W_h}$. Therefore, $\hat{\mathbf{W}}_\mathbf{h}$ can be represented as follows:

$$\hat{\mathbf{W}}_\mathbf{h} = \mathbf{(X_h^t X_h)^{-1} X_h^t F_h} \tag{5.5}$$

where $\mathbf{X_h}$ refers to the design locations given beforehand and $\mathbf{F_h}$ is the performance value from the simulation. $\mathbf{X_h^t X_h}$ is called the information matrix in the DOE literature. In the context of a quadratic regression model,

$$\mathbf{X_h^t X_h} = \begin{pmatrix} N & \sum x_{hi} & \sum x_{hi}^2 \\ \sum x_{hi} & \sum x_{hi}^2 & \sum x_{hi}^3 \\ \sum x_{hi}^2 & \sum x_{hi}^3 & \sum x_{hi}^4 \end{pmatrix}.$$

In the literature of DOE, there are many criteria used to decide the optimal allocation of simulation replications among the different design locations for each partition. For example, A-optimality aims to minimize the trace of the inverse of the information matrix. It results in minimizing the average of the estimation of the regression coefficients. D-optimality seeks to minimize $\left|\left(\mathbf{X_h^t X_h}\right)^{-1}\right|$, i.e., maximizing the determinant of the information matrix. This criterion will lead to maximizing the differential Shannon information content of the parameter estimation. Another popular criterion called G-optimality tries to minimize the maximum variance of the predicated value. Furthermore, other criteria such as E-optimality, I-optimality and V-optimality have also been well studied.

The problem considered here is different from all the optimality criteria discussed above. Our objective is to maximize the probability of selecting the best design location with the minimum performance value among all the mk design locations with a fixed simulation budget *T*. Based on the estimation of the coefficients $\mathbf{\hat{W}_h = (X_h^t X_h)^{-1} X_h^t F_h}$, we could compute the performance value at every design location $\hat{y}(x_{hi}) = \hat{W}_{h0} + \hat{W}_{h1} x_{hi} + \hat{W}_{h2} x_{hi}^2; h = 1,...,m; i = 1,...,k$. Let x_{Bb} be the design location with the performance value $y(x_{Bb})$ being the smallest. Therefore, the probability of correct selection is the probability that $\hat{y}(x_{Bb})$ is indeed the smallest performance value. Let N_{hi} be the number of simulation replications allocated to design location x_{hi}. Since the computing budget is always limited, our objective is to find the best way to allocate the total budget such that the probability of correct selection can be maximized. Mathematically, we can write this optimal computing budget allocation problem as follows:

$$\begin{aligned} &\max \quad P\left\{\hat{y}(x_{Bb}) < \hat{y}(x_{hi})\right\}, \forall h = 1,...,m; i = 1,...,k; \\ &s.t. \quad \sum_{h=1}^{m}\sum_{i=1}^{k} N_{hi} = T \end{aligned} \tag{5.6}$$

where T is the total number of computing budget available.

The nature of the optimization model (5.6) makes it very difficult to solve. Firstly, the distribution of $\hat{y}(x_{hi})$ is unknown, and we must conduct simulation to estimate the mean performance value of $f(x_{hi})$. We can only have a good estimation when the total computing budget *T* is exhausted. In the next two sections, we will simplify the optimization model (5.6) and make it easier to solve.

5.3. Bayesian Regression Framework

The first step to solve the optimization model (5.6) is to obtain the distribution of $\hat{y}(x_{hi})$. To do so, we must estimate the coefficients $\mathbf{W_h}$ in the quadratic equations (5.4). We assume that the simulated performance value $\mathbf{F_h}$ follows multi-variate normal distribution with mean $\mathbf{X_h W_h}$ and a covariance matrix $\sigma_h^2 \boldsymbol{I}$, where $\mathbf{I}$ is the identity matrix and σ_h^2 is the variance in equation (5.3). Given that $\mathbf{W_h}$ and σ_h^2 are known, we can derive the probability density function of $\mathbf{F_h}$ as follows:

$$p(\mathbf{F_h} \mid \mathbf{W_h}, \sigma_h^2) = \frac{1}{\sqrt{(2\pi\sigma_h^2)^n}} \exp[\frac{1}{2\sigma_h^2}(\mathbf{F_h} - \mathbf{X_h W_h})^t (\mathbf{F_h} - \mathbf{X_h W_h})] \qquad (5.7)$$

where $\mathbf{W_h}$ is unknown, $\mathbf{F_h}$ is known from the simulation output and σ_h^2 can estimated based on the simulation output.

Our objective is to derive the distribution of $\mathbf{W_h}$. We could use the idea of conditional probability to express $\mathbf{W_h}$ as a function of $\mathbf{F_h}$ as follows.

Firstly, we can decompose the condition probability of equation (5.7) to be equation (5.8) below:

$$p(\mathbf{F_h} \mid \mathbf{W_h}, \sigma_h^2) = \frac{p(\mathbf{F_h}, \mathbf{W_h}, \sigma_h^2)}{p(\mathbf{W_h}, \sigma_h^2)} = \frac{1}{p(\mathbf{W_h} \mid \sigma_h^2)} \frac{p(\mathbf{F_h}, \mathbf{W_h}, \sigma_h^2)}{p(\sigma_h^2)} . \qquad (5.8)$$

Secondly, given the simulation output vector $\mathbf{F_h}$, the condition probability density function of $\mathbf{W_h}$ can be written as

$$p(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2) = \frac{p(\mathbf{F_h}, \mathbf{W_h}, \sigma_h^2)}{p(\mathbf{F_h}, \sigma_h^2)} = \frac{1}{p(\mathbf{F_h} \mid \sigma_h^2)} \frac{p(\mathbf{F_h}, \mathbf{W_h}, \sigma_h^2)}{p(\sigma_h^2)}. \qquad (5.9)$$

Combining equations (5.8) and (5.9), we have the following equation:

$$p(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2) = \frac{p(\mathbf{F_h} \mid \mathbf{W_h}, \sigma_h^2)\, p(\mathbf{W_h} \mid \sigma_h^2)}{p(\mathbf{F_h} \mid \sigma_h^2)} \qquad (5.10)$$

In this equation, we know that the term $p(\mathbf{F_h} \mid \sigma_h^2)$ does not provide any information on the estimation of the parameter $\mathbf{W_h}$ by Gill (2002). It is a normalization term to make sure that the probabilities sum up to one. Therefore, we could write the conditional probability $p(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2)$ as being parameterized by $p(\mathbf{F_h} \mid \mathbf{W_h}, \sigma_h^2)\, p(\mathbf{W_h} \mid \sigma_h^2)$,

$$p(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2) \propto p(\mathbf{F_h} \mid \mathbf{W_h}, \sigma_h^2)\, p(\mathbf{W_h} \mid \sigma_h^2). \qquad (5.11)$$

A noninformative improper prior distribution commonly used is

$$p(\mathbf{W_h}, \sigma_h^2) \propto \frac{1}{\sigma_h^2}.$$

Since σ_h^2 is assumed to be known in our problem, the conditional distribution of $\mathbf{W_h}$ would be $p(\mathbf{W_h} \mid \sigma_h^2) \propto \frac{1}{\sigma_h^2}$. We could write the conditional probability density function of $p(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2)$ as following the below expression:

$$p(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2) \propto \frac{1}{\sigma_h^{n+2}} \exp[\frac{1}{-2\sigma_h^2} (\mathbf{F_h} - \mathbf{X_h W_h})^t (\mathbf{F_h} - \mathbf{X_h W_h})]. \qquad (5.12)$$

Hence, the mean and variance of $\mathbf{W_h}$ are as follows:

$$\begin{aligned} E(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2) &= \mathbf{(X_h^t X_h)^{-1} X_h^t F_h} \\ \mathrm{cov}(\mathbf{W_h} \mid \mathbf{F_h}, \sigma_h^2) &= \sigma_h^2 \mathbf{(X_h^t X_h)^{-1}} \end{aligned} \tag{5.13}$$

Given that $\mathbf{F_h}$ is the performance value from simulation, the mean and variance of $\mathbf{W_h}$ from the previous equation are in fact the mean and variance for the estimation of the coefficients $\mathbf{\hat{W}_h}$. As $\hat{y}(x_{hi}) = \hat{W}_{h0} + \hat{W}_{h1} x_{hi} + \hat{W}_{h2} x_{hi}^2$; $h = 1,...,m;\ i = 1,...,k$ is a linear combination of $\mathbf{\hat{W}_h}$, it can be concluded that $\hat{y}(x_{hi})$ follows a multi-variate normal distribution:

$$\hat{y}(x_{hi}) \sim N\left(\mathbf{X_{hi} (X_h^t X_h)^{-1} X_h^t F_h}, \sigma_h^2 \mathbf{X_{hi}^t (X_h^t X_h)^{-1} X_{hi}}\right). \tag{5.14}$$

5.4. Required Number of Support Points

In the literature of the design of experiment, the design locations which are used to simulate the performance are called the support points for the regression. We have mentioned previously that we need at least three design locations in order to avoid singularity for the quadratic regression problems. The theorem below formally states that we only need three support points for the quadratic regression.

Theorem 5.1 Given that we assume the expectation of our underlying function is quadratic within each partition, we require only three support points on each partition and two of these support points will be at the extreme design locations, i.e., x_{h1} and x_{hk} for partition h.

Proof: The result that we only need three support points comes from de la Garza (1954), which states that we only need *m*+1 support points for a polynomial of degree *m*. In addition, Kiefer (1959) concluded that two of the support points will be at the extreme regardless of what optimality criterion is used. □

From the results of Theorem 5.1, we assume that the support points are $\{x_{h1}, x_{hs}, x_{hk}\}, 1 < s < k$ for partition *h*, where x_{hs} can be different in different partitions. Let $N_{h\bullet}$ be the number of computing budget allocated to partition *h*. We also define $\alpha_{hi} = N_{hi} / N_{h\bullet}, i = 1, s, k$ to be the proportion of computing budget allocated to design location *i* in partition *h*. Therefore, the optimal computing budget problem (5.6) can be re-written as follows:

$$\begin{aligned} &\max \quad P\left\{\hat{y}(x_{Bb}) < \hat{y}(x_{hi})\right\}, \forall h = 1, ..., m; i = 1, ..., k; \\ &s.t. \quad \sum_{h=1}^{m} N_{h\bullet}(\alpha_{h1} + \alpha_{hs} + \alpha_{hk}) = T \\ &\qquad\ \ \alpha_{h1}, \alpha_{hs}, \alpha_{hk} \geq 0 \end{aligned} \tag{5.15}$$

5.5. Characterization of Optimal Allocation Rule

Section 5.3 concluded that $y(x_{hi}), h = 1, ..., m; i = 1, ..., k$ is normally distributed with mean $\mathbf{X_{hi}(X_h^t X_h)^{-1} X_h^t F_h}$ and variance $\sigma_h^2 \mathbf{X_{hi}^t (X_h^t X_h)^{-1} X_{hi}}$. Hence, the large deviation principle is satisfied for these random variables. In this section, the rate function of false selection will be derived using large deviation theory.

Since x_{Bb} is the best design location with performance value $y(x_{Bb}) < y(x_{hi}); h=1,...,m; i=1,...,k$ and $\hat{y}(x_{hi})$ is the estimated value of $y(x_{hi})$, false selection will occur if $\hat{y}(x_{Bb})$ is not the smallest value. More specifically, false selection occurs if $\hat{y}(x_{Bb}) \geq \min_{\substack{i\neq b \\ i=1,...,k}} \hat{y}(x_{Bi})$ or $\hat{y}(x_{Bb}) \geq \min_{\substack{h\neq B, h=1,...,m \\ i=1,...,k}} \hat{y}(x_{hi})$. The probability of false selection is

$$P\{FS\} = P\left(\hat{y}(x_{Bb}) \geq \min\left\{ \min_{\substack{i\neq b \\ i=1,...,k}} \hat{y}(x_{Bi}), \min_{\substack{h\neq B, h=1,...,m \\ i=1,...,k}} \hat{y}(x_{hi}) \right\} \right). \qquad (5.16)$$

It is easy to see that this probability is bounded below by

$$\max\left\{ \max_{i\neq b} P\left(\hat{y}(x_{Bb}) \geq \hat{y}(x_{Bi})\right), \max_{\substack{h\neq B, h=1,...,m \\ i=1,...,k}} P\left(\hat{y}(x_{Bb}) \geq \hat{y}(x_{hi})\right) \right\}$$

and bounded above by

$$\left(km-1\right)\times\max\left\{ \max_{i\neq b} P\left(\hat{y}(x_{Bb}) \geq \hat{y}(x_{Bi})\right), \max_{\substack{h\neq B, h=1,...,m \\ i=1,...,k}} P\left(\hat{y}(x_{Bb}) \geq \hat{y}(x_{hi})\right) \right\}.$$

Define $\beta_h = N_{h\bullet}/T,\ h=1,...,m$ to be the proportion of total budget allocated to partition *h*, where β_B is the proportion of budget allocation to the best partition, i.e., the partition which contains the best design location x_{Bb}. By definition, it can be concluded that $\sum_{h=1}^{m} \beta_h = 1$. For partition *h*, define $\alpha_{hi} = N_{hi}/N_{h\bullet}, i=1,s,k$ and $\alpha_{h1}+\alpha_{hs}+\alpha_{hk}=1$. $\boldsymbol{\alpha}_\mathbf{h} = (\alpha_{h1}, \alpha_{hs}, \alpha_{hk})$ is the proportion of budget $N_{h\bullet}$ allocated to design locations x_{h1}, x_{hs} and x_{hk} within partition *h*, where $\boldsymbol{\alpha}_{\mathbf{B}} = (\alpha_{B1}, \alpha_{Bs}, \alpha_{Bk})$ refers to the allocation within the best partition.

Thus, for $1 < h < m, 1 < i < k$,

$$\begin{aligned}
&\lim_{T\to\infty} \frac{1}{n} \ln P\left\{\hat{y}(x_{Bb}) \geq \hat{y}(x_{hi})\right\} = -R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}}), h \neq B \\
&\lim_{T\to\infty} \frac{1}{n} \ln P\left\{\hat{y}(x_{Bb}) \geq \hat{y}(x_{Bi})\right\} = -R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}), i \neq b
\end{aligned}$$

for some rate function $R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}}), R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}})$. Then

$$-\lim_{T\to\infty} \frac{1}{T} \ln P\left\{FS\right\} = \min\left\{ \min_{i \neq b} R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}), \min_{\substack{h \neq B, h=1,\ldots,m \\ i=1,\ldots,k}} R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}}) \right\} \tag{5.17}$$

Let the scaled cumulant generating function of $\left(\hat{y}(x_{Bb}), \hat{y}(x_{hi})\right)$ and $\left(\hat{y}(x_{Bb}), \hat{y}(x_{Bi})\right)$ be denoted as

$$\begin{aligned}
&\lim_{T\to\infty} \frac{1}{T} \Lambda^{\left(\hat{y}(x_{Bb}), \hat{y}(x_{hi})\right)}(T\beta_B, T\beta_h) = \lim_{T\to\infty} \frac{1}{T} \ln E\left(e^{T\beta_B \hat{y}(x_{Bb}) + T\beta_h \hat{y}(x_{hi})}\right) \\
&\lim_{T\to\infty} \frac{1}{T} \Lambda^{\left(\hat{y}(x_{Bb}), \hat{y}(x_{Bi})\right)}(T\beta_B) = \lim_{T\to\infty} \frac{1}{T} \ln E\left(e^{T\beta_B \hat{y}(x_{Bb}) + T\beta_B \hat{y}(x_{hi})}\right)
\end{aligned}$$

By the Gärtner-Ellis Theorem (Dembo and Zeitouni, 1998), $\left(\hat{y}(x_{Bb}), \hat{y}(x_{hi})\right)$ and $\left(\hat{y}(x_{Bb}), \hat{y}(x_{Bi})\right)$ satisfy the large deviation principle with good rate function as follows:

$$\begin{aligned}
&R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}}) = \inf_{v} \left(\beta_B I_{Bb}(v) + \beta_h I_{hi}(v)\right) \\
&R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}) = \inf_{v} \left(\beta_B I_{Bb}(v) + \beta_B I_{Bi}(v)\right)
\end{aligned} \tag{5.18}$$

where $I(x) = \sup_{\theta \in R}\left(\theta x - \ln E(e^{\theta x})\right)$. In the case of normal distribution, $I_{hi}(v) = \left(v - y(x_{hi})\right)^2 / 2\xi_{hi}^2$ if $y(x_{hi}) \sim N\left(y(x_{hi}), \xi_{hi}^2\right)$.

Lemma 5.1 The rate function of probability of false selection can be explicitly expressed as follows:

$$-\lim_{T\to\infty}\frac{1}{T}\ln P\left\{FS\right\} = \min\left\{\min_{i\neq b} R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}), \min_{\substack{h\neq B, h=1,\ldots,m \\ i=1,\ldots,k}} R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}})\right\}$$

$$R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}})$$
$$= \frac{\delta_{hi}^2 / 2}{\frac{\sigma_B^2}{\beta_B}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right) + \frac{\sigma_h^2}{\beta_h}\left(\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}}\right)}, \forall h \neq B, 1 < i < k$$

$$R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}})$$
$$= \frac{\delta_{Bi}^2 / 2}{\frac{\sigma_B^2}{\beta_B}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right) + \frac{\sigma_B^2}{\beta_B}\left(\frac{E_{Bi,1}^2}{\alpha_{B1}} + \frac{E_{Bi,s}^2}{\alpha_{Bs}} + \frac{E_{Bi,k}^2}{\alpha_{Bk}}\right)}, \forall i \neq b$$

$$E_{hi,1} = \left\{\frac{(x_{hs} - x_{hi})(x_{hk} - x_{hi})}{(x_{h1} - x_{hs})(x_{h1} - x_{hk})}\right\}, E_{hi,s} = \left\{\frac{(x_{h1} - x_{hi})(x_{hk} - x_{hi})}{(x_{hs} - x_{h1})(x_{hs} - x_{hk})}\right\},$$
$$E_{hi,k} = \left\{\frac{(x_{h1} - x_{hi})(x_{hs} - x_{hi})}{(x_{hk} - x_{h1})(x_{hk} - x_{hs})}\right\}, \delta_{hi}^2 = \left(y(x_{Bb}) - y(x_{hi})\right)^2, \delta_{Bi}^2 = \left(y(x_{Bb}) - y(x_{Bi})\right)^2$$

Proof: See Appendix A.□

Maximizing the probability of correct selection is equivalent to minimizing the false selection probability. The asymptotically optimal allocation strategy will result from maximizing the rate at which the false selection probability goes to zero as a function of $\beta_h, \boldsymbol{\alpha}_{\mathbf{h}}, h = 1, \ldots, m$. Thus, the optimization model (5.15) is equivalent to finding the best $\beta_h, \boldsymbol{\alpha}_\mathbf{h}, h=1,...,m$ that solves the following optimization problem:

$$\begin{aligned}
&\max\min\left\{\min_{i\neq b} R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_\mathbf{B}),\ \min_{\substack{h\neq B, h=1,...,m\\ i=1,...,k}} R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_\mathbf{B}, \boldsymbol{\alpha}_\mathbf{h})\right\}\\
s.t.\quad &\alpha_{h1}+\alpha_{hs}+\alpha_{hk}=1, h=1,...,m \qquad\qquad (5.19)\\
&\sum_{h=1}^{m}\beta_h=1,\ \beta_h, \alpha_{h1}, \alpha_{hs}, \alpha_{hk}\geq 0, h=1,...,m
\end{aligned}$$

The nonlinear optimization model above is highly complex because of the large number of decision variables as well as the complexity of the rate functions. Although we can use nonlinear optimization solver to get the optimal allocation directly, our objective is to derive a simple allocation rule which can be easily implemented in actual simulation studies. In the following Section 5.6, we will analyze the limiting allocation rule when the number of partitions goes to infinity.

5.6. Limiting Approximation to the Optimal Allocation Rule

In order to solve the optimization model (5.19) efficiently, we propose a limiting approximation to the solution of model (5.19). We will explore the problem structure of model (5.19) through asymptotical analysis. The analysis will show that the process of solving $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}_\mathbf{h}, h=1,...,m$ can be decomposed. The results indicate that the budget allocation rule between partitions follows similarly with the OCBA rule (Chen et al., 2000), while the budget allocation rules within the partitions is the OSD for best partition and a feasibility determination problem for other partitions.

We consider the limiting scenario when the number of partitions *m* goes to infinity. To understand what drives *m* to infinity, consider the context when we divide the entire domain into more and more partitions. As *m* becomes large, the number of design locations within each partition tends to be smaller, and simulation noise among design locations within the partition tends to be closer. This justifies our assumption that the simulation noise is an independent identical standard normal random variable for each partition as shown in equation (5.3).

The following assumptions are made before we start to analyze the limiting behaviors of the optimization model (5.19).

Assumption 5.1: The design locations are approximately equally spaced, .i.e.,

$$(x_{hi} - x_{h(i-1)}) = (x_{hj} - x_{h(j-1)}), \forall h \in \{1,...,m\}; \forall i, j \in \{1,...,k\} .$$

The importance of assumption 5.1 is demonstrated by the following Lemma 5.2 and the proofs of Theorem 5.2. This assumption is generally held since it is natural and common to discretize the continuous domain equally. It is not meaningful to make some of the points close to zero while the others are far away.

Assumption 5.2: Assume the following conditions are true.

(1) $0 < V_L < y(x_{Bb}) < \cdots < y(x_{hi}) < \cdots < V_U < \infty$.

(2) There always exists $\delta > 0$ such that $y(x_{hi}) - y(x_{Bb}) > \delta, \forall h, \forall i$.

(3) The simulation noise is such that $0 < \sigma_h^2 < \infty, \forall h$.

Assumption 5.2 (1) states that the mean performance at each design location is finite. The second condition makes sure that the performance difference between any pair of design locations is significant. In other words, the performance at two different design locations is comparable. The last condition guarantees that the noise (variance) is finite.

Lemma 5.2 Under assumption 5.1, the coefficient $\left[\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}}\right]$ is always finite, and there always exists a constant *C* such that

$$\left[\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}}\right] \le C\left[\frac{E_{qj,1}^2}{\alpha_{q1}} + \frac{E_{qj,s}^2}{\alpha_{qs}} + \frac{E_{qj,k}^2}{\alpha_{qk}}\right],$$

where $h, q \in \{1,..,m\}; i, j \in \{1,...,k\}$.

Proof: See Appendix B.□

The results of Lemma 5.2 will be used when we prove Theorem 5.2 below.

Theorem 5.2: Under assumptions 5.1 and 5.2, the following statements are true.

(1) There exists $c < \infty$ such that $\beta_h^* \le c\beta_q^*$ for all $h, q \in \{1,...,m\}; h, q \ne B$ and for all m.

(2) There exists $c < \infty$ such that $\beta_h^* \le c\beta_{\min}^*, \forall h \ne B, \beta_{\min}^* = \min_q \left\{\beta_q^*\right\}$.

(3) $\beta_h^* \to 0, \forall h \ne B$ as $m \to \infty$.

(4) $\beta_h^* / \beta_B^* \to 0, \forall h \ne B$ as $m \to \infty$.

Proof: See Appendix C.□

The main assertion of Theorem 5.2 is that $\beta_h / \beta_B \to 0$ as the number of partitions m goes to infinity. In other words, it says that the fraction of simulation budget allocated to the best partition far exceeds the fraction given to other partitions when the number of partitions goes to infinity. This result is meaningful if we think of each non-best partition as an individual attempting to "beat" the best partition. Far more simulation budget should be allocated to the best partition in order for it to "defeat" all the competing partitions.

The rate function associated with design location x_{hi} for $h = 1,...m; h \ne B$ is as follows:

$$R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_\mathbf{B}, \boldsymbol{\alpha}_\mathbf{h}) = \frac{\delta_{hi}^2 / 2}{\dfrac{\sigma_B^2}{\beta_B}\left(\dfrac{E_{Bb,1}^2}{\alpha_{B1}} + \dfrac{E_{Bb,s}^2}{\alpha_{Bs}} + \dfrac{E_{Bb,k}^2}{\alpha_{Bk}}\right) + \dfrac{\sigma_h^2}{\beta_h}\left(\dfrac{E_{hi,1}^2}{\alpha_{h1}} + \dfrac{E_{hi,s}^2}{\alpha_{hs}} + \dfrac{E_{hi,k}^2}{\alpha_{hk}}\right)}.$$

The rate function above implies that the convergent rate associated with design location x_{hi} depends on the value of $\beta_B, \beta_h, \boldsymbol{\alpha}_\mathbf{B}, \boldsymbol{\alpha}_\mathbf{h}$. As $m \to \infty$, $\beta_h / \beta_B \to 0$, and hence

$$\left(\left(\frac{\sigma_B^2}{\beta_B}\right)\left(\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right)\right) \Bigg/ \left(\left(\frac{\sigma_h^2}{\beta_h}\right)\left(\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}}\right)\right) \to 0.$$

Therefore, the rate function $R_{Bb,hi}(\beta_B,\beta_h,\boldsymbol{\alpha}_\mathbf{B},\boldsymbol{\alpha}_\mathbf{h})$ approaches $\tilde{R}_{Bb,hi}(\beta_h,\boldsymbol{\alpha}_\mathbf{h})$ as $m\to\infty$,

$$R_{Bb,hi}(\beta_B,\beta_h,\boldsymbol{\alpha}_\mathbf{B},\boldsymbol{\alpha}_\mathbf{h})\to\tilde{R}_{Bb,hi}(\beta_h,\boldsymbol{\alpha}_\mathbf{h})=\frac{\delta_{hi}^2/2}{\dfrac{\sigma_h^2}{\beta_h}\left(\dfrac{E_{hi,1}^2}{\alpha_{h1}}+\dfrac{E_{hi,s}^2}{\alpha_{hs}}+\dfrac{E_{hi,k}^2}{\alpha_{hk}}\right)}. \tag{5.20}$$

Hence, the optimization model (5.19) can be written as follows when $m\to\infty$:

$$\begin{aligned}&\max\min\left\{\min_{i\neq b}R_{Bb,Bi}(\beta_B,\boldsymbol{\alpha}_\mathbf{B}),\min_{\substack{h\neq B,h=1,\ldots,m\\ i=1,\ldots,k}}\tilde{R}_{Bb,hi}(\beta_h,\boldsymbol{\alpha}_\mathbf{h})\right\}\\ s.t.\quad&\alpha_{h1}+\alpha_{hs}+\alpha_{hk}=1,h=1,\ldots,m\\ &\sum_{h=1}^{m}\beta_h=1,\beta_h,\alpha_{h1},\alpha_{hs},\alpha_{hk}\geq 0,h=1,\ldots,m\end{aligned} \tag{5.21}$$

Lemma 5.3 For each $h=1,\ldots,m;h\neq B$, suppose $\tilde{\boldsymbol{\alpha}}_\mathbf{h}^*$ is the optimal solution to the problem

$$\begin{aligned}&\max\min_{i=1,\ldots,k}\tilde{R}_{Bb,hi}(\beta_h,\boldsymbol{\alpha}_\mathbf{h})\\ &s.t.\ \alpha_{h1}+\alpha_{hs}+\alpha_{hk}=1,\alpha_{h1},\alpha_{hs},\alpha_{hk}>0\end{aligned} \tag{5.22}$$

and $\tilde{\boldsymbol{\alpha}}_\mathbf{B}^*$ is the optimal solution to

$$\begin{aligned}&\max\min_{i\neq b}R_{Bb,Bi}(\beta_B,\boldsymbol{\alpha}_\mathbf{B})\\ &s.t.\ \alpha_{B1}+\alpha_{Bs}+\alpha_{Bk}=1,\alpha_{B1},\alpha_{Bs},\alpha_{Bk}>0\end{aligned} \tag{5.23}$$

$\tilde{\boldsymbol{\alpha}}_\mathbf{h}^*$ and $\tilde{\boldsymbol{\alpha}}_\mathbf{B}^*$ are also optimal solutions of the model (5.21).

Proof: Optimization model (5.21) can be re-expressed as follows.

$$\begin{aligned} &\max \quad z \\ s.t. \quad & R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}) \geq z, i \neq b \\ & \tilde{R}_{Bb,hi}(\beta_h, \boldsymbol{\alpha}_{\mathbf{h}}) \geq z, h \neq B, h = 1,...,m; i = 1,...,k \\ & \alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1, \forall h = 1,...,m \\ & \sum_{h=1}^{m} \beta_h = 1, \beta_h, \alpha_{h1}, \alpha_{hs}, \alpha_{hk} \geq 0, \forall h = 1,...,m \end{aligned} \tag{5.24}$$

and optimization models (5.23) and (5.24) can be rewritten as

$$\begin{aligned} &\max z \\ s.t. \; & \tilde{R}_{Bb,hi}(\beta_h, \boldsymbol{\alpha}_{\mathbf{h}}) \geq z, i = 1,...,k \\ & \alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1, \alpha_{h1}, \alpha_{hs}, \alpha_{hk} > 0 \end{aligned} \tag{5.25}$$

$$\begin{aligned} &\max \quad z \\ s.t. \; & R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}) \geq z, i \neq b \\ & \alpha_{B1} + \alpha_{Bs} + \alpha_{Bk} = 1, \alpha_{B1}, \alpha_{Bs}, \alpha_{Bk} > 0 \end{aligned} \tag{5.26}$$

It is easy to see that models (5.25) and (5.26) have the same objective function with model (5.24). However, the domain of model (5.24) is a subset of models (5.25) and (5.26). Therefore, if $\tilde{\boldsymbol{\alpha}}_{\mathbf{h}}^{*}$, $h = 1,...,m; h \neq B$ and $\tilde{\boldsymbol{\alpha}}_{\mathbf{B}}^{*}$ are feasible to model (5.24), $\tilde{\boldsymbol{\alpha}}_{\mathbf{h}}^{*}$ and $\tilde{\boldsymbol{\alpha}}_{\mathbf{B}}^{*}$ are optimal to model (5.24). We see that from Lemma 5.1 and equation (5.20),

$$\begin{aligned} & R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}) \\ &= \frac{\delta_{Bi}^2 / 2}{\frac{\sigma_B^2}{\beta_B}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right) + \frac{\sigma_B^2}{\beta_B}\left(\frac{E_{Bi,1}^2}{\alpha_{B1}} + \frac{E_{Bi,s}^2}{\alpha_{Bs}} + \frac{E_{Bi,k}^2}{\alpha_{Bk}}\right)}, \forall h = B, i \neq b \end{aligned}$$

$$\tilde{R}_{Bb,hi}(\beta_h, \boldsymbol{\alpha}_{\mathbf{h}}) = \frac{\delta_{hi}^2 / 2}{\frac{\sigma_h^2}{\beta_h}\left(\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}}\right)}.$$

Therefore, the optimal solution to models (5.25) and (5.26) does not depend on the value of $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_m)$. The optimal solutions $\tilde{\boldsymbol{\alpha}}^*_{\mathbf{h}}$ and $\tilde{\boldsymbol{\alpha}}^*_{\mathbf{B}}$ remain the same even when $\boldsymbol{\beta}$ changes. Let $\boldsymbol{\beta}^* = (\beta^*_1, \beta^*_2, ..., \beta^*_m)$ be the optimal solution of model (5.24). Let any $\boldsymbol{\beta} = \boldsymbol{\beta}^*$, $\tilde{\boldsymbol{\alpha}}^*_{\mathbf{h}}$ and $\tilde{\boldsymbol{\alpha}}^*_{\mathbf{B}}$ remain the same and since $\boldsymbol{\beta} = \boldsymbol{\beta}^*$ is feasible to model (5.24), we conclude that $\tilde{\boldsymbol{\alpha}}^*_{\mathbf{h}}$ and $\tilde{\boldsymbol{\alpha}}^*_{\mathbf{B}}$ are also optimal to model (5.21).□

The main assertion of Lemma 5.3 is that we can solve for $\boldsymbol{\alpha}^*_{\mathbf{h}}$ for each h=1, 2,..., k separately when the number of partitions m goes to infinity. It is an important property since it helps us to decompose the solving process of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, which leads to a possible closed-form solution. Solving the models (5.22) and (5.23) $k-2$ times assuming $s = 2, 3, ..., k-1$ will determine the best location of s for each partition. Lemma 5.4 below further explains that the process of solving for $\boldsymbol{\alpha}$ is essentially a typical research problem found in the literature.

Lemma 5.4 The limiting optimal allocation rules within each partition $h, \forall h$ can be determined as follows:

(1) $\forall h \neq B$, $\boldsymbol{\alpha}^*_{\mathbf{h}} = \left(\alpha^*_{h1}, \alpha^*_{hs}, \alpha^*_{hk}\right)$ can be determined by solving the following feasibility determination problem:

$$\begin{aligned} &\min g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right) \\ s.t.\quad &g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right) = \sum_{i=1}^{k} P\left\{y\left(x_{hi}\right) \leq \mu_{Bb}\right\} \\ &\alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1, \alpha_{h1}, \alpha_{hs}, \alpha_{hk} \geq 0 \end{aligned} \tag{5.27}$$

where μ_{Bb} is the constant mean performance value at best design location x_{Bb}.

(2) The budget allocation rule within the best partition $\boldsymbol{\alpha}_{\mathbf{B}}^{*} = \left(\alpha_{B1}^{*}, \alpha_{Bs}^{*}, \alpha_{Bk}^{*}\right)$ is simply the OSD.

Proof: (1) It is easy to see that $g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right)$ is bounded below and above as follows:

$$\max_i P\left\{y\left(x_{hi}\right) \leq \mu_{Bb}\right\} \leq g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right) \leq k \max_i P\left\{y\left(x_{hi}\right) \leq \mu_{Bb}\right\}. \quad (5.28)$$

Since $y\left(x_{hi}\right)$ is a normally distributed random variable with variance ξ_{hi}^2 as shown above, Gartner-Ellis theorem implies that

$$-\lim_{n\to\infty} \frac{1}{n} \ln P\left\{y(x_{hi}) \leq \mu_{Bb}\right\} = I_{hi}(x). \quad (5.29)$$

Therefore,

$$-\lim_{n\to\infty} \frac{1}{n} \ln g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right) = \min_i I_{hi}(x). \quad (5.30)$$

Furthermore, we can express (5.30) explicitly as

$$\begin{aligned} -\lim_{n\to\infty} \frac{1}{n} \ln g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right) &= \min_i \frac{\left(y\left(x_{hi}\right) - \mu_{Bb}\right)^2 / 2}{\xi_{hi}^2} \\ &= \min_i \frac{\left(y(x_{hi}) - \mu_{Bb}\right)^2 / 2}{\dfrac{\sigma_h^2}{\beta_h}\left(\dfrac{E_{hi,1}^2}{\alpha_{h1}} + \dfrac{E_{hi,s}^2}{\alpha_{hs}} + \dfrac{E_{hi,k}^2}{\alpha_{hk}}\right)} \end{aligned} \quad (5.31)$$

where (5.31) is the large deviation rate of $g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right)$. Minimizing $g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right)$ is equivalent to maximizing this convergent rate at which

$g_h\left(\alpha_{h1}, \alpha_{hs}, \alpha_{hk}\right)$ goes to zero. Therefore, the feasibility determination problem (5.27) is equivalent to the following model.

$$\max \min_i \frac{\left(y(x_{hi}) - \mu_{Bb}\right)^2 / 2}{\frac{\sigma_h^2}{\beta_h}\left(\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}}\right)} \quad s.t.\ \alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1, \alpha_{h1}, \alpha_{hs}, \alpha_{hk} \geq 0\ . \tag{5.32}$$

As a result, the optimization model (5.23) is essentially the same as the feasibility determination problem (5.27).

(2) As shown in Lemma 5.3, the value of β_B does not affect the optimal solution for model (5.23). The only decision variables in model (5.23) are $\alpha_{B1}, \alpha_{Bs}, \alpha_{Bk}$. This is essentially the same problem discussed in Brantley et al. (2013a) for solving the budget allocation problem if the entire domain is treated as one partition. □

The idea of Lemma 5.4 can be graphically shown in Figure 5.1. The comparison between design location $x_{hi}, h \neq B$ and the best design x_{Bb} is simply to determine whether $y(x_{hi})$ is feasible to the range $[\mu_{Bb}, \infty)$, where μ_{Bb} is the mean performance value at the best design location. It is a constant and assumed to be known. However, the comparison within the best design location is the same as the OSD problem (Brantley et al. 2013a).

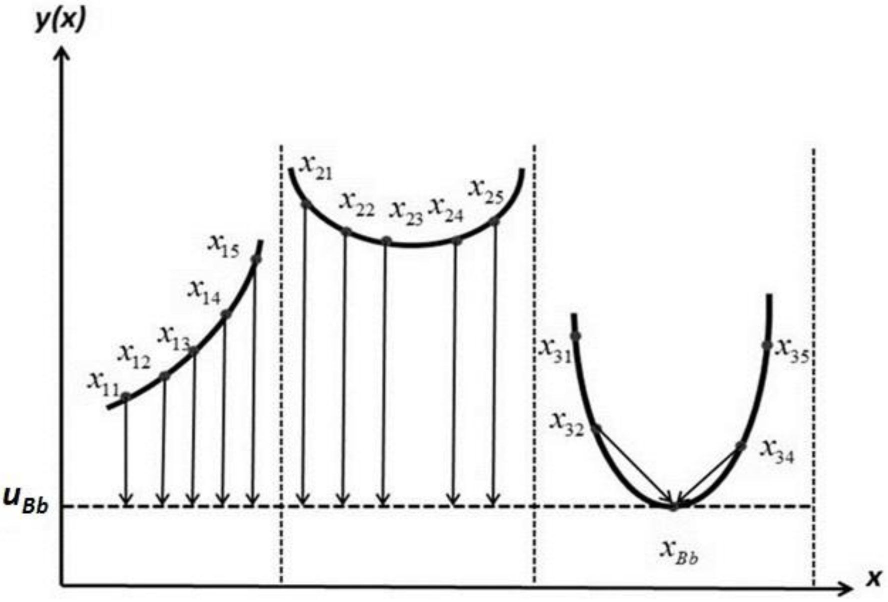

Fig.5.1. Graphical representation of Lemma 5.4

As shown in Lemmas 5.3 and 5.4, the limiting optimal allocation rules within each partition h can be determined by solving a feasibility check problem (5.27) for $h \neq B$, and allocation within the best partition is simply the OSD. The optimization model used to solve for $\boldsymbol{\alpha}_\mathbf{h}, \forall h$ does not depend on the value of $\boldsymbol{\beta}$.

For each $h = 1,...,m, h \neq B$, let $i_h^* = \arg\max_{i=1,...,k} \min\{\tilde{R}_{Bb,hi}(\beta_h, \boldsymbol{\alpha}_\mathbf{h})\}$. Since the optimal model used to solve $\boldsymbol{\alpha}_\mathbf{h}$ does not depend on the value of $\boldsymbol{\beta}$, the optimal solution is always achieved at design location i_h^* for partition h no matter what the value of $\boldsymbol{\beta}$ is. Similarly, we can define $i_B^* = \arg\max_{i \neq b} R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_\mathbf{B})$.

Theorem 5.3 As $m \to \infty$, the limiting asymptotically optimal allocation that asymptotically minimizes the probability of false selection for the problem (5.19)

$$\begin{aligned}
&\max \min \left\{ \min_{i \neq b} R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_{\mathbf{B}}), \min_{\substack{h \neq B, h=1,\ldots,m \\ i=1,\ldots,k}} R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}}) \right\} \\
s.t. \quad & \sum_{h=1}^{m} \beta_h = 1 \\
& \alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1, \forall h = 1,\ldots,m \\
& \beta_h, \alpha_{h1}, \alpha_{hs}, \alpha_{hk} \geq 0, \forall h = 1,\ldots,m
\end{aligned}$$

is such that

$$\begin{cases}
\beta_B^* = \sqrt{\sigma_B^2 \left(\dfrac{E_{Bb,1}^2}{\alpha_{B1}^*} + \dfrac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \dfrac{E_{Bb,k}^2}{\alpha_{Bk}^*} \right) \sum\limits_{\substack{h=1 \\ h \neq B}}^{h} \dfrac{\beta_h^{*2}}{\sigma_h^2 \left(\dfrac{E_{hi_h^*,1}^2}{\alpha_{h1}^*} + \dfrac{E_{hi_h^*,s}^2}{\alpha_{hs}^*} + \dfrac{E_{hi_h^*,k}^2}{\alpha_{hk}^*} \right)}} \\
\dfrac{\beta_h^*}{\beta_q^*} = \dfrac{\sigma_h^2 \left(\dfrac{E_{hi_h^*,1}^2}{\alpha_{h1}^*} + \dfrac{E_{hi_h^*,s}^2}{\alpha_{hs}^*} + \dfrac{E_{hi_h^*,k}^2}{\alpha_{hk}^*} \right) / \delta_{hi_h^*}^2}{\sigma_q^2 \left(\dfrac{E_{qi_q^*,1}^2}{\alpha_{q1}^*} + \dfrac{E_{qi_q^*,s}^2}{\alpha_{qs}^*} + \dfrac{E_{qi_q^*,k}^2}{\alpha_{qk}^*} \right) / \delta_{qi_q^*}^2}, h, q = 1,\ldots,m; h, q \neq B
\end{cases} \tag{5.33}$$

where $\alpha_{h1}^*, \alpha_{hs}^*, \alpha_{hk}^*$ and $\alpha_{B1}^*, \alpha_{Bs}^*, \alpha_{Bk}^*$ are obtained by solving the optimization problem in Lemma 5.4.

Proof: Given that we could solve for i_h^* separately for each partition $h = 1,\ldots,m,$ the convergent rate $\min\limits_{\substack{h \neq B, h=1,\ldots,m \\ i=1,\ldots,k}} R_{Bb,hi}(\beta_B, \beta_h, \boldsymbol{\alpha}_{\mathbf{B}}, \boldsymbol{\alpha}_{\mathbf{h}})$ will be a function of β_B and β_h only. Therefore, we can write it as $\min\limits_{h \neq B, h=1,\ldots,m} R_{Bb,hi_h^*}(\beta_B, \beta_h)$. Similarly, we can use $\min\limits_{i \neq b} R_{Bb,Bi_B^*}(\beta_B)$ to replace

$\min_{i \neq b} R_{Bb,Bi}(\beta_B, \boldsymbol{\alpha}_\mathbf{B})$ for the best partition. Since we know that the minimum rate occurs at design location $x_{hi_h^*}$ for each partition $h = 1,...,m,$ we can re-write the optimization model as

$$\begin{aligned} &\max \min \left\{ R_{Bb,Bi_B^*}(\beta_B), \min_{h \neq B, h=1,...,m} R_{Bb,hi_h^*}(\beta_B, \beta_h) \right\} \\ s.t. \quad &\sum_{h=1}^{m} \beta_h = 1 \\ &\beta_h \geq 0, h = 1,...,m \end{aligned} \tag{5.34}$$

where $R_{Bb,hi_h^*}(\beta_B, \beta_h)$

$$= \frac{\left(\delta_{hi_h^*}^2 / 2\right)}{\left(\sigma_B^2 / \beta_B\right)\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*} + \frac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right) + \left(\sigma_h^2 / \beta_h\right)\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*} + \frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*} + \frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*}\right)} \tag{5.35}$$

$R_{Bb,hi_h^*}(\beta_B, \beta_h)$ and $R_{Bb,Bi_B^*}(\beta_B)$ are concave and strictly increasing functions of β_h and β_B. Therefore, the optimization problem is a concave programming problem. Thus, the first order condition is also the optimality condition. We first re-write the optimization model as follows:

$$\begin{aligned} &\max z \\ s.t. \quad &R_{Bb,hi_h^*}(\beta_B, \beta_h) - z \geq 0, h = 1,....,m, h \neq B \\ &R_{Bb,Bi_B^*}(\beta_B) - z \geq 0 \\ &\sum_{h=1}^{m} \beta_h = 1 \\ &\beta_h \geq 0, h = 1,...,m \end{aligned} \tag{5.36}$$

From the Karush–Kuhn–Tucker conditions, we know that there exist $\lambda_h \geq 0, h = 1,...,m$ and $\gamma > 0$ such that

$$1 - \sum_{h=1}^{m} \lambda_h = 1 \tag{5.37}$$

$$\gamma - \lambda_h \frac{\partial R_{Bb,hi_h^*}(\beta_B^*, \beta_h^*)}{\partial \beta_h} = 0, h = 1,...,m; h \neq B \tag{5.38}$$

$$\gamma - \sum_{h=1,h\neq B}^{m} \lambda_h \frac{\partial R_{Bb,hi_h^*}(\beta_B^*, \beta_h^*)}{\partial \beta_B} - \lambda_B \frac{\partial R_{Bb,Bi_B^*}(\beta_B^*)}{\partial \beta_B} = 0 \tag{5.39}$$

$$\lambda_h \left(z - R_{Bb,hi_h^*}(\beta_B^*, \beta_h^*) \right) = 0, h = 1,...,m; h \neq B \tag{5.40}$$

$$\lambda_B \left(z - R_{Bb,Bi_B^*}(\beta_B^*) \right) = 0 \tag{5.41}$$

Based on equation (5.37), there must exist some $\lambda_h > 0, h = 1,...,m;$ however, if we assume that there is one $\lambda_h = 0, h = 1,...,m; h \neq B$, we could conclude that $\gamma = 0$ from (5.38). $\gamma = 0$ will lead to $\lambda_h = 0$ for all $h = 1,...,m; h \neq B$. Therefore, we conclude that $\lambda_h > 0, h = 1,...,m; h \neq B$. This means we must have $z = R_{Bb,hi_h^*}(\beta_B^*, \beta_h^*), h = 1,...,m; h \neq B$.

On the other hand, $R_{Bb,Bi_B^*}(\beta_B^*) > R_{Bb,hi_h^*}(\beta_B^*, \beta_h^*), h = 1,...,m; h \neq B$ since $\beta_h / \beta_B \to 0$ as m goes to infinity. Based on this result, we can conclude that $\lambda_B = 0$ from equation (5.41). As a result of $\lambda_B = 0$, equation (5.39) can be simplified to $\sum_{h=1,h\neq B}^{m} \lambda_h \frac{\partial R_{Bb,hi_h^*}(\beta_B^*, \beta_h^*)}{\partial \beta_B} = \gamma$.

Substituting equation (5.38) into the simplified equation (5.39), we obtain the following results:

$$\begin{cases} \sum_{\substack{h=1 \\ h\neq B}}^{m} \dfrac{\partial R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*)/\partial\beta_B}{\partial R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*)/\partial\beta_h} = 1 \\ z = R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*), \forall h = 1,\dots,m; h \neq B \end{cases}$$

This is equivalent to the following expression:

$$\delta_{hi_h^*}^2 \Bigg/ \left(\left(\sigma_B^2/\beta_B^*\right)\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*} + \frac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \frac{E_{Bb,k}^2}{\alpha_{Bk}^*} \right) + \left(\sigma_h^2/\beta_h^*\right)\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*} + \frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*} + \frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*} \right) \right)$$
$$= \delta_{qi_q^*}^2 \Bigg/ \left(\left(\sigma_B^2/\beta_B^*\right)\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*} + \frac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \frac{E_{Bb,k}^2}{\alpha_{Bk}^*} \right) + \left(\sigma_q^2/\beta_q^*\right)\left(\frac{E_{qi_q^*,1}^2}{\alpha_{q1}^*} + \frac{E_{qi_q^*,s}^2}{\alpha_{qs}^*} + \frac{E_{qi_q^*,k}^2}{\alpha_{qk}^*} \right) \right)$$

From Theorem 5.2, we know that $\beta_h^*/\beta_B^* \to 0, \forall h \neq B$ as $m \to \infty$ · Therefore, we can simplify the equation above as follows:

$$\delta_{hi_h^*}^2 \Bigg/ \left(\left(\sigma_h^2/\beta_h^*\right)\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*} + \frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*} + \frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*} \right) \right)$$
$$= \delta_{qi_q^*}^2 \Bigg/ \left(\left(\sigma_q^2/\beta_q^*\right)\left(\frac{E_{qi_q^*,1}^2}{\alpha_{q1}^*} + \frac{E_{qi_q^*,s}^2}{\alpha_{qs}^*} + \frac{E_{qi_q^*,k}^2}{\alpha_{qk}^*} \right) \right)$$

where $h,q = 1,\dots,m; h,q \neq B; i_h^*$ and i_q^* are the minimum rate location for partitions h and q. The equations above can be further reduced to

$$\frac{\beta_h^*}{\beta_q^*} = \frac{\sigma_h^2 \left(\dfrac{E_{hi_h^*,1}^2}{\alpha_{h1}^*} + \dfrac{E_{hi_h^*,s}^2}{\alpha_{hs}^*} + \dfrac{E_{hi_h^*,k}^2}{\alpha_{hk}^*} \right) / \delta_{hi_h^*}^2}{\sigma_q^2 \left(\dfrac{E_{qi_q^*,1}^2}{\alpha_{q1}^*} + \dfrac{E_{qi_q^*,s}^2}{\alpha_{qs}^*} + \dfrac{E_{qi_q^*,k}^2}{\alpha_{qk}^*} \right) / \delta_{qi_q^*}^2}, h,q = 1,\dots,m; h,q \neq B\,. \qquad (5.42)$$

The partial derivatives can be expressed explicitly as

$$\frac{\partial R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*)}{\partial\beta_B}$$

$$=\frac{\sigma_B^2\delta_{hi_h^*}^2\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)/\beta_B^{*2}}{\left(\left(\sigma_B^2/\beta_B^*\right)\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)+\left(\sigma_h^2/\beta_h^*\right)\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*}\right)\right)^2}$$

$$\frac{\partial R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*)}{\partial\beta_h}$$

$$=\frac{\sigma_h^2\delta_{hi_h^*}^2\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*}\right)/\beta_h^{*2}}{\left(\left(\sigma_B^2/\beta_B^*\right)\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)+\left(\sigma_h^2/\beta_h^*\right)\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*}\right)\right)^2}.$$

Hence, $\sum_{\substack{h=1\\h\neq B}}^{m}\frac{\partial R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*)/\partial\beta_B}{\partial R_{Bb,hi_h^*}(\beta_B^*,\beta_h^*)/\partial\beta_h}=1$ is equivalent to the following equation:

$$\sum_{\substack{h=1\\h\neq B}}^{m}\frac{\sigma_B^2\delta_{hi_h^*}^2\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)/\beta_B^{*2}}{\sigma_h^2\delta_{hi_h^*}^2\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*}\right)/\beta_h^{*2}}=1 \qquad (5.43)$$

which can be rewritten as

$$\beta_B^*=\sqrt{\sigma_B^2\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)\sum_{\substack{h=1\\h\neq B}}^{h}\frac{\beta_h^{*2}}{\sigma_h^2\left(\frac{E_{hi_h^*,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h^*,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h^*,k}^2}{\alpha_{hk}^*}\right)}}\,. \qquad (5.44)$$

This completes the proof of Theorem 5.3.□

The cross partition allocation rule presented in Theorem 5.3 is similar to the OCBA rule (Chen et al., 2000). The fraction of budget allocated to the non-best partition is proportional to its signal-to-noise ratio that is defined as the variance divided by the squared mean difference. The simulation budget allocated to the best partition is the weighted sum of all other partitions. The results match our intuitions that the best partition should take most of the simulation budget. The non-best partitions will be allocated more if they have larger variance or are closer to the best partition.

5.7. A Sequential Algorithm for Implementation

The allocation rule or the value of $\boldsymbol{\alpha}_{\mathbf{h}}, h = 1, ..., m$ and $\boldsymbol{\beta}$ can only be determined after we know the distribution of $y(x_{hi})$. In actual implementation, the distribution of $y(x_{hi})$ is unknown. We will propose a sequential allocation rule and use sampling distribution to estimate the allocation rule step by step. The quadratic regression-based OCBA (OCBA-QR) procedure can be implemented as follows:

Step 0: Define the input m (the number of partitions), k (the number of design locations), T (the computing budget), x_{hi} (the design locations with partitions pre-determined), n_0 (the number of initial runs), Δ (the increment at each iteration).

Step 1: Perform $\frac{n_0}{3m}$ simulation replications for three design locations in each partition using the D-opt support points with $N_{h1} = N_{h\frac{(k+1)}{2}} = N_{hK} = \frac{n_0}{3m}$.

Step 2: While $\sum_{h=1}^{m}(N_{h1} + N_{hs} + N_{hk}) < T$, do

a. Estimate a quadratic regression equation using the information from all prior simulation runs for each partition.

b. Estimate the mean and variance of each design location using $\hat{y}(x_{hi}) = \hat{\beta}_{h0} + \hat{\beta}_{h1}x_{hi} + \hat{\beta}_{h2}x_{hi}^2; h = 1,...,m; i = 1,...,k$

c. Determine the observed global best design so that $x_{Bb} = \arg\min_{Bi} \hat{y}(x_i)$.

d. Solve the optimization model in Lemma 5.4 to obtain $\boldsymbol{\alpha}_\mathbf{h}$ and *s* for $h = 1,...,m$.

e. Compute $\boldsymbol{\beta}$ using Theorem 5.3.

Step 3: Increase the computing budget by Δ and calculate the new budget allocations using $\boldsymbol{\alpha}_\mathbf{h}, h = 1,...,m$ and $\boldsymbol{\beta}$ from step 2.

Step 4: Perform $\max\{N_{h,i+1}, N_{hi}\}, h = 1,...,m; i = 1, s, k$ runs of simulation replications, and go to step 2.

5.8. Numerical Experiments

In this section, we will conduct several numerical experiments to test our proposed simulation budget allocation rule and compare it with some of the existing allocation procedures. Different allocation procedures are used to solve the same simulation optimization problem with identical experimental settings. We start our description by introducing other allocation procedures.

5.8.1 Allocation Procedures

The most commonly used and simplest method is to allocate the simulation budget equally to each of the design locations. The number of simulation replications received by each design location is T/mk in equal allocation (EA). The best design location is the point x_b such that

$$x_b = \arg\min_{q=1,\ldots,mk} \frac{\sum_{j=1}^{T/mk} f(x_j)}{T/mk}$$

A better way of finding the best design location is to use the OCBA rule proposed by Chen et al. (2000). The OCBA rule allocates the computing budget sequentially with the number of simulation replications allocated to each design location being determined by the signal-to-noise ratio. Let b be the best design location among the total mk design locations, N_i be the number of simulation replications allocated to design location i, $i \in \{1, 2, ..., mk\}, i \neq b$

$$\begin{cases} \dfrac{N_i}{N_j} = \dfrac{\sigma_i^2 / (\mu_i - \mu_b)^2}{\sigma_j^2 / (\mu_j - \mu_b)^2}, i, j \in \{1, 2, ..., mk\}, i, j \neq b \\ N_b = \sigma_b \sqrt{\sum_{i=1, i \neq b}^{mk} \dfrac{N_i^2}{\sigma_i^2}} \end{cases}$$

The best design location is the point x_b such that,

$$x_b = \arg\min_{i=1,...,mk} \frac{\sum_{j=1}^{N_j} f(x_j)}{N_j}.$$

Both EA and OCBA use the mean performance value at each design location for comparison, and the mean performance value is computed directly from the simulation output. They do not rely on any response surface to estimate the performance value at any design location. In the experiments, we will also compare the other procedures that use quadratic equations as the response surface to estimate the performance value.

A typical simulation budget allocation rule in design of experiment is called D-optimal when it tries to maximize the determinant of the information matrix. According to the D-optimal rule, the simulation budget should be equally allocated to design locations 1, (1+*k*)/2 and *k*. In the case of partitioned domains, we equally allocate the simulation budget to each partition.

In addition, we want to compare with the POSD method proposed by Brantley et al. (2013b). POSD is the improved allocation rule of OSD when the entire domain is divided into various partitions.

The last method used in the comparison is the allocation rule we proposed in Theorem 5.3 and implemented using the sequential allocation algorithm in Section 5.7. We name our proposed quadratic regression-based OCBA allocation rule as OCBA-QR.

5.8.2. Experiments

In our experiments, the probability of correct selection (PCS) is used as the performance measure. PCS is estimated by counting the number of times we successfully find the best design location out of 10,000 independent simulation runs for each of the allocation procedure. In order to have a fair comparison, we have set the initial number of simulation replications n_0 to be the same for different allocation procedures. The number of partitions and the number of design locations in each partition are exactly the same for each allocation procedure.

The experiments are conducted with 10,000 independent simulation runs each. Under exactly identical experimental settings, the performance of each allocation procedure is presented below.

Experiment 5.1

The first experiment we conduct is taken from the classical experiment (T örn and Žilinskas,1989). The function we use for this experiment is

$$f(x_i) = \sin(x_i) + \sin(10x_i / 3) + \ln(x_i) - 0.84x_i + 3 .$$

The noise of simulation is assumed to be a standard normal random variable. As shown in Figure 5.2 below, we discretize the domain of the function into 60 evenly spaced points from 3 to 8. There are three minimum points within the domain [3, 8], but the global minimum point occur at $x = 5.2$ with $y(x) = -1.6$.

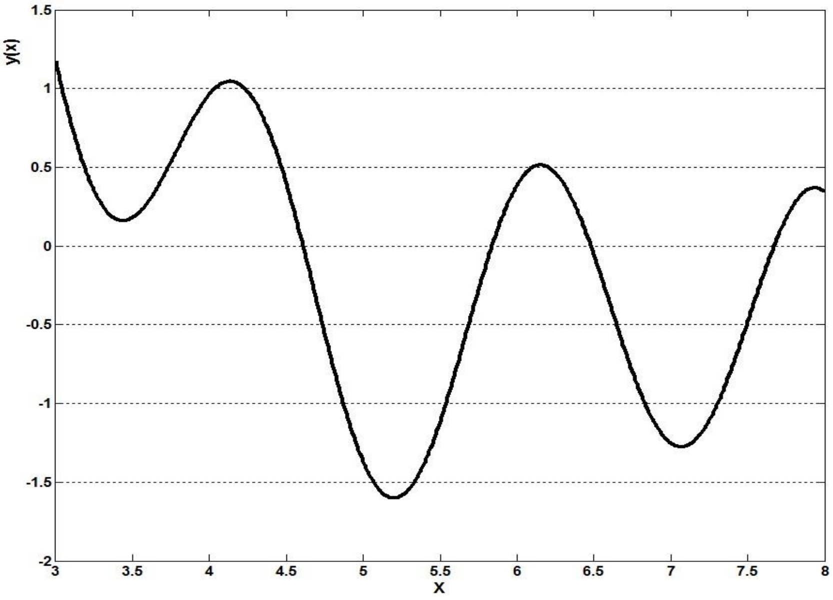

Fig.5.2. Graph of optimization function in experiment 1

This is the same experiment conducted in Brantley et al. (2013b). The simulation results of using different allocation rules are shown below in Figure 5.3. It is clear that OCBA-QR, POSD and D-optimality allocation rule perform much better than OCBA and EA. This shows that incorporating the quadratic equations as the response surface has greatly increased the probability of correct selection with a limited fixed simulation budget. In addition, the performance of D-optimality between OCBA-QR and POSD are significant. Therefore, it is important for us to derive efficient simulation budget allocation rules instead of simply using the D-optimality method. Lastly, our OCBA-QR performs slightly better than POSD in this experiment.

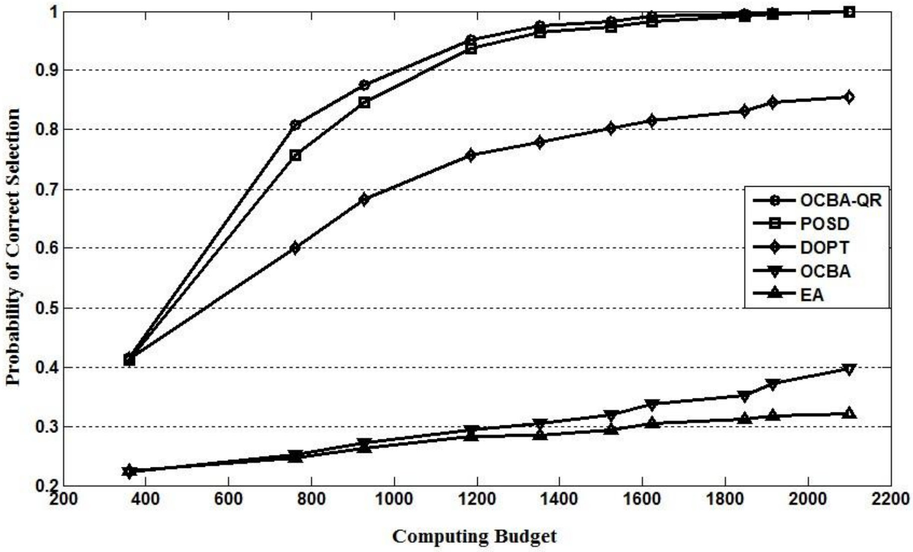

Fig.5.3. PCS comparison of OCBA-QR, POSD, DOPT, OCBA and EA

Experiment 5.2

Consider the function $f(x_i) = 1 - \sin^6(5\pi x_i)\exp\left(-2\ln 2\left(\frac{x_i - 0.1}{0.8}\right)^2\right)$. Under the assumption of discrete domain, we have divided the entire domain into 100 discrete points for $x \in [0.05, 1.05]$. It can be easily determined that the global minimum point is $x_{16} = 0.1$ with the optimal value of $y(x_{16}) = 0$. Four local minimum points can be found at $x_{26} = 0.3$, $x_{46} = 0.5$, $x_{66} = 0.7$, $x_{86} = 0.9$. The graphical representation of this function is shown in Figure 5.4.

The 100 discrete points are divided into 10 partitions, and each one contains 10 design locations. The experiment assumes that the noise for simulation is half of the standard normal random variables. Figure 5.5 shows the performance of each allocation procedure.

In this numerical experiment, our proposed method OCBA-QR also performs best among all five allocation rules. In particular, the advantage of using OCBA-QR rather than POSD becomes more significant for the problem.

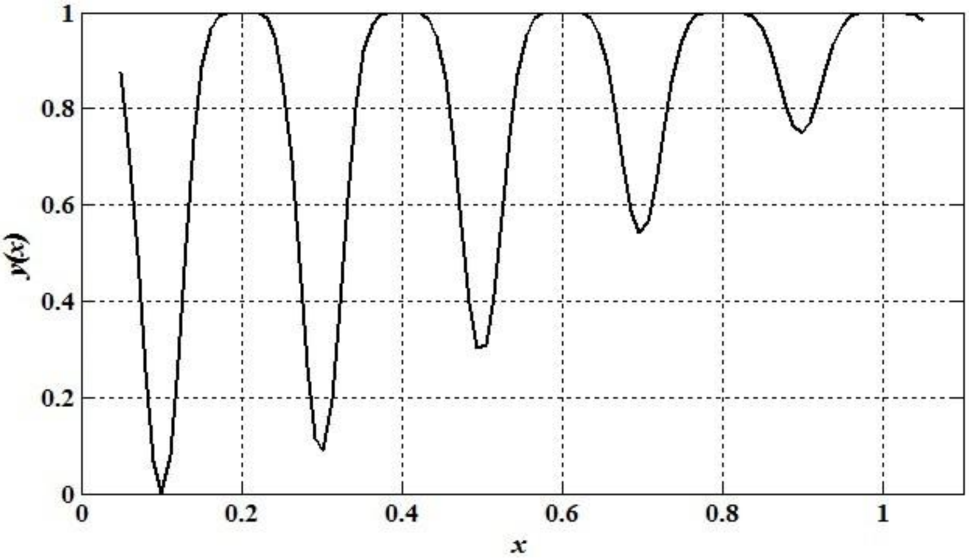

Fig.5.4. Graph of optimization function in experiment 2

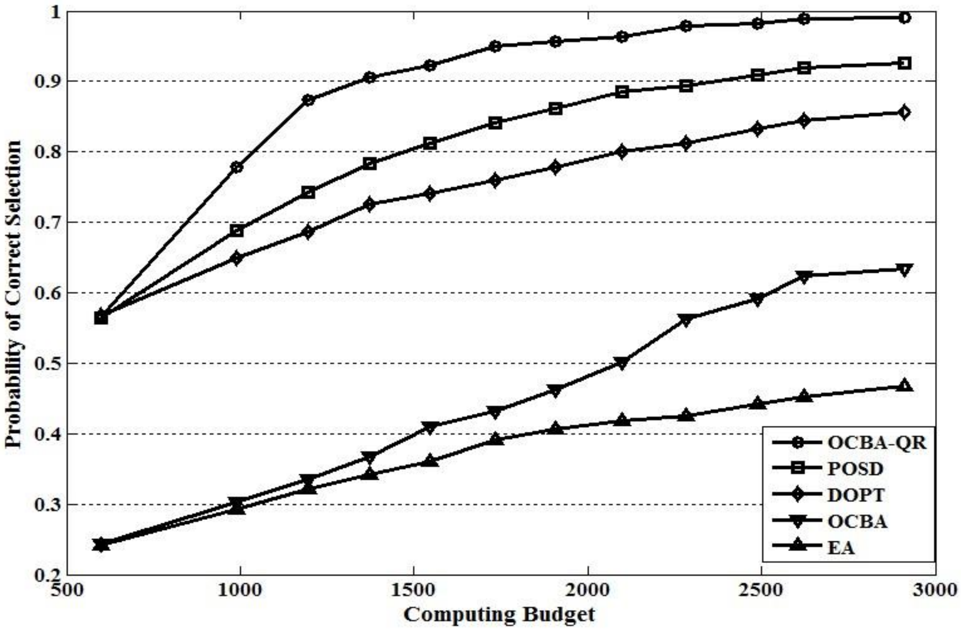

Fig.5.5. PCS comparison of OCBA-QR, POSD, DOPT,OCBA and EA.

Experiment 5.3

We consider a two dimensional problem in our last experiment. We use the 2-D Griewank Function which is one of the most common examples in global optimization literature (Fu et al., 2006). The 2-D form is given as

$$f(x_1, x_2) = \frac{1}{40}(x_1^2 + x_2^2) - \cos(x_1)\cos(\frac{x_2}{\sqrt{2}}) + 1$$

where x_1 and x_2 are continuous variables with $-10 \le x_1, x_2 \le 10$. As shown below in Figure 5.6, there many local minimum points, while the global minimum point is at $x_1 = x_2 = 0$ with optimal objective value of 0.

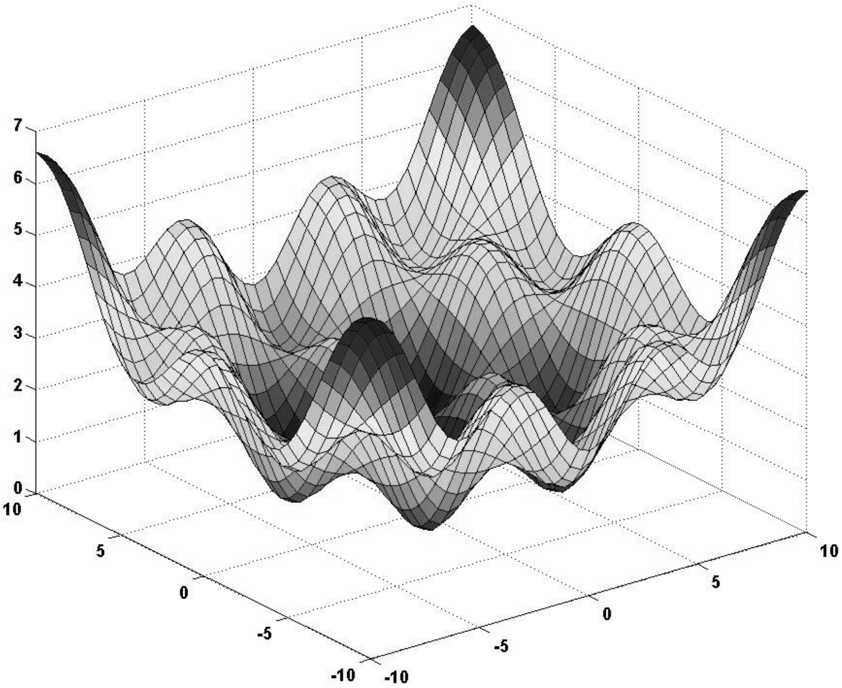

Fig.5.6. Graph of optimization function in experiment 3

In this experiment, we discretize the domain into 21×21 discrete points, i.e., $x_1 = [-10, -9, ..., 10]$ and $x_2 = [-10, -9, ..., 10]$. The discrete points are divided into 21 partitions with 21 points in each partition. Partition *i* consists of points $(i-11,-10),(i-11,-9),\cdots,(i-11,10)$. For each partition, we assume that the underlying performance can be modeled as a quadratic function, where the independent variable is x_2 and the dependent variable is $f(x_1,x_2)$. The noise of the simulation is assumed to be a standard normal random variable. We conduct the experiment for different allocation rules as shown previously. The performance of these methods is shown below in Figure 5.7. It is clear that our OCBA-QR method performs best among all different allocation rules.

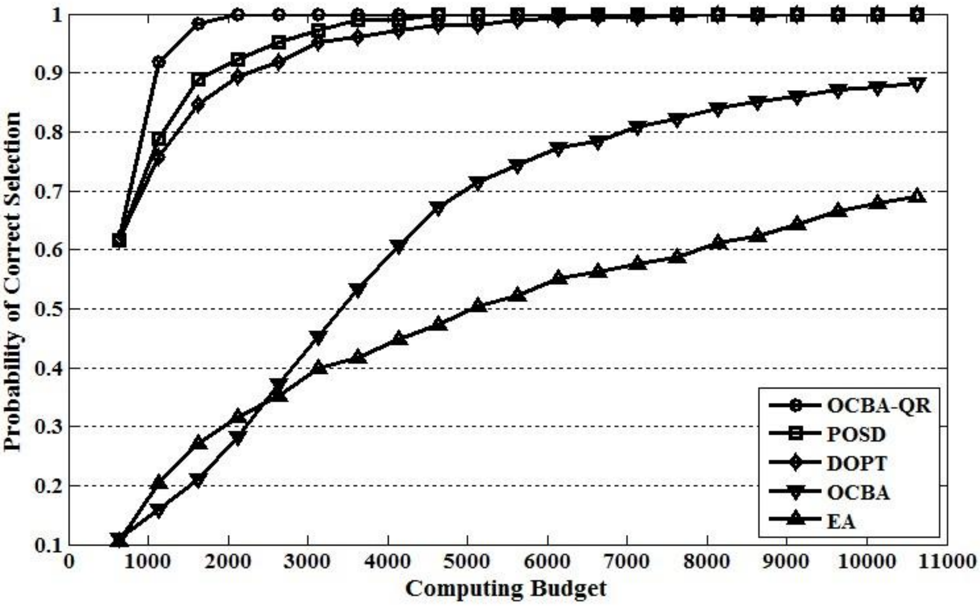

Fig.5.7. PCS comparison of OCBA-QR, POSD, DOPT, OCBA and EA.

From all the three experiments, we could conclude that our proposed allocation rule OCBA-QR is not only a better allocation rule in terms of asymptotical optimality but also performs better in practical simulation. In addition, we approximate the allocation rule when the number of partitions goes to infinity. The numerical experiments also show that the advantage of using our allocation rule becomes more significant when the number of partitions increases. This matches our theoretical derivation that our allocation rule approximates the optimal allocation when the number of partitions goes to infinity.

5.9. Conclusion

In this chapter, we have further enhanced the simulation efficiency of finding the best by incorporating the quadratic equation as the response surface. Based on large deviation theory, we have formulated the problem and derived the optimal allocation rule. We further analyze the limiting behaviors of the allocation rule when the number of partitions goes to infinity. The limiting allocation rule has been shown to be intuitive. The highly complex problem can be decomposed into small problems. The cross partition allocation rule is similar with the original OCBA problem, while the within partition allocation becomes the OSD for the best partition and feasibility determination problem for other partitions. We conducted numerical experiments to implement our proposed allocation rule. It has been shown to be the most efficient allocation rule compared with POSD, OCBA, DOPT and EA.

Chapter 6. Conclusion and Future Research

6.1. Conclusion

We propose three new optimal computing budget allocation (OCBA) procedures in this thesis for ranking and selection with a fixed limited simulation budget. In order to improve the simulation efficiency, we use large deviation theory to formulate these problems as optimization models and derive respective asymptotically optimal allocation rules and closed-form approximated allocation rules.

The first procedure aims to determine the most efficient way of allocating the simulation replications so as to maximize the probability of correctly ranking all alternatives completely. This procedure fills in the research gaps of OCBA in the area of statistical ranking as no previous research considered such a problem using the OCBA framework. Compared with existing indifference zone allocation rule, our procedure reduces the number of simulation budget significantly as shown in the numerical experiment results. Asymptotically optimal allocation rules can be used by decision makers who are concerned more about optimality, while the approximated allocation rules can be useful for practical implementation under finite budget.

Motivated by the idea of integrating the statistical ranking procedure into evolutionary algorithms, we extend the complete ranking problem to top *m* ranking, i.e., rank the top *m* designs out of *k* alternatives. The top *m* ranking problem can be reduced to complete ranking if *m* is equal to *k*, and to the original OCBA problem when *k* is equal to 1. Therefore, it can be regarded as a generalization of previous problems in the literature. We formulate the budget allocation problem using large deviation and derive the asymptotically optimal allocation rule and a closed-form approximated rule. The proposed approximated allocation rule is then integrated with genetic algorithms to solve simulation optimization problems. The numerical experiments have shown that significant simulation budget were saved by integrating our proposed budget allocation rule with GA.

The last problem we consider in this thesis is to determine the simulation budget allocation rule when the simulation output can be modeled by quadratic regression functions. The domain is divided into many partitions, and a quadratic equation is regressed in each partition. Using the large deviation theory, we have characterized the asymptotically optimal allocation rule while a previous approach only provides a heuristic approximated solution. We further analyze the limiting behaviors of the allocation rule assuming that the number of partitions goes to infinity. The limiting scenario analysis has provided us with more intuition and insight on the problem. The cross partition allocation rule has been shown to be similar with the original OCBA rule while the within partition allocation rule reduces to the feasibility determination problems for non-best partitions and the OSD for the best partition. Our limiting asymptotically optimal rule has been shown to be effective through numerical experiments.

6.2. Future Research

There are several limitations in our research which can lead to possible future research problems. Firstly, we have assumed independent sampling for the problem considered in the thesis. In practice, the design performances are usually sampled in the presence of correlation. Therefore, a potential research problem will be on how to determine the simulation budget allocation for complete ranking and top *m* ranking with correlated sampling.

Secondly, although we have integrated our budget allocation into GA, we only considered how the simulation budget should be allocated for each individual iteration. We did not consider how many simulation replications should be given to each iteration and how many iterations should the search algorithm run. Given that the total number of simulation replications is fixed, more iterations would mean less budget for each iteration and fewer iterations would result in more budget for individual iterations. How to make such a tradeoff between the number of iterations and the simulation budget for each iteration remains an open research topic.

Lastly, we used quadratic regression functions to model the simulation output in Chapter 5. However, other response surfaces can also be used based on the underlying structure of performance across the domain. Therefore, one possible extension of Chapter 5 will be to consider the simulation budget allocation rule when the simulation output can be modeled by other functions such exponential and log-linear. In addition, we only consider the one dimension problem in Chapter 5. It is also important to consider how we can use the quadratic equations to model the simulation output if each design location has more than one performance measurements, i.e., how can we determine the simulation budget allocation for multi-objective simulation optimization problems. Moreover, one assumption made in Chapter 5 is that the partition of the domain is given beforehand. It would however be more useful if we consider the scenario where the partition is not given. Therefore, how to partition the domain is another important research question that can be explored in the future.

Bibliography

Alirezaee, M.R. and M. Afsharian. 2007. A complete ranking of DMUs using restrictions in DEA models. *Applied Mathematics and Computation*, 189(2), 1550–1559.

Andradóttir, S., D. Goldsman, B. W. Schmeiser, L.W. Schruben, E. Yücesan. 2005. Analysis methodology: Are we done?. *Proceedings of the 2005 Winter Simulation Conference*, Association for Computing Machinery, New York, 790–796.

Atkinson, A.C., A. N. Donev. 1998. Optimum Experimental Designs. *Oxford Science Publications*, Oxford.

Barton, R. R. 2005. Issues in Development of Simultaneous Forward-Inverse Metamodels. *Proceedings of the 2005 Winter Simulation Conference*, IEEE, Piscataway, NJ 209-217. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, J. A. Joines, eds.

Beirlant, J., E.J. Dudewicz and E.C. van der Meulen.1982. Complete statistical ranking of populationswith tables and applications. *Journal of Computational and Applied Mathematics*, 1982. 8(3): 187-201.

Bechhofer, R. E. 1954. A Single-Sample Multiple Decision Procedure for Ranking Means of Normal Populations with known Variances. *Annals of Mathematical Statistics*, 25(1): 16-39.

Bechhofer, R. E., T. J. Santner, D.M.Goldman. 1995. Design and Analysis of Experiments for Statistical Selection, Screening, and Multiple Comparisons. John Wiley & Sons, New York.

Bishop, T. A. 1978. Designing simulation experiments to completely rank alternatives. *Proceeding of the 1978 Winter Simulation Conference*, pp. 203-205.

Bishop, T.A. and E.J. Dudewicz.1977. Complete ranking of reliability-related distributions. *IEEE Transactions on Reliability*, 26(5), 362.

Blickle,T. and L. Thiele. 1995. Comparison of Selection Schemes used in Genetic Algorithms. Computer Engineering and Communication Networks Lab TIK-Report, Nr. 11, Dec 1995,Version2

Borkar, V. S. 2008. Stochastic Approximation: A Dynamical Systems Viewpoint. Cambridge, UK: Cambridge University Press.

Branke, J., S. E. Chick and C. Schmidt. 2007. Selecting a selection procedure. *Management Science* 53(12): 1916-1932.

Brantley, M. W., L. H. Lee, C. H. Chen, and A. Chen. 2013a. Efficient Simulation Budget Allocation with Regression, *IIE Transactions*, 45(3), pp. 291-308.

Brantley, M. W., L. H. Lee, and C. H. Chen. 2013b. An Efficient Simulation Budget Allocation Method Incorporating Regression for Partitioned Domains, to appear in *Automatica*, 2013.

Chen, C. H. , D. He, M. Fu, and L. H. Lee. 2008. Efficient simulation budget allocation for selecting an optimal subset. *INFORMS Journal on Computing*, 20(4), pp. 579–595.

Chen, C. H., E. Yücesan, L. Dai, and H. C. Chen. 2010. Efficient Computation of Optimal Budget Allocation for Discrete Event Simulation Experiment. *IIE Transactions*, 42(1), pp. 60-70.

Chen, C. H. and L. H. Lee. 2010. Stochastic Simulation Optimization - An Optimal Computing Budget Allocation, System Engineering and Operations Research, vol. 1. Singapore: World Scientific.

Chen, C. H., J. Lin, E. Yücesan, and S. E. Chick. 2000. Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Discrete Event Dynamic Systems: Theory Applications*, 10, pp. 251-270.

Chen, E.J. 2009. Subset selection procedures. *Journal of Simulation*, 3, 202–210.

Chen, H. F. 2002. Stochastic Approximation and Its Applications, Kluwer Academic Publishers, Dordrecht, 2002.

Chen, W., S. Gao, C. H. Chen, L. Shi. 2013. An Optimal Sample Allocation Strategy for Partition-based Random Search. *IEEE Transactions on Automation Science and Engineering*, Digital Object Identifier: 10.1109/TASE.2013.2251881.

Cheng, R. C. H., J. P. C. Kleijnen. 1999. Improved design of queueing simulation experiments with highly heteroscedastic responses. *Operations Research*, 47(5) 762-777.

Chew, E. P., L. H. Lee, S.Y. Teng and C.H. Koh. 2009. Differentiated service inventory optimization using nested partitions and MOCBA. *Computers & Operations Research*, 36(5): 1703-1710.

Chick, S.E. and K. Inoue. 2001. New two-stage and sequential procedures for selecting the best simulated system. *Operations Research*, 49(5), p. 732-743.

DeGroot, M. H. 1970. Optimal Statistical Decisions. McGraw Hill, New York.

De la Garza, A. 1954. Spacing of Information in Polynomial Regression. *The Annals of Mathematical Statistics*, 25(1) 123-130.

Dembo, A. and O. Zeitouni. 1998. Large Deviations Techniques and Applications. New York: Springer-Verlag, 1998.

Dourmas, N. G., V. N. Nikitakos, Maria. L.A. 2008. A methodology for rating and ranking hazards in maritime formal safety assessment using fuzzy logic, R&RATA #2(Vol 1)2008, June.

Dudewicz, E. J., S. R. Dalal. 1975. Allocation of observations in ranking and selection with unequal variances. *Sankhya: The Indian Journal of Statistics*, 37 28–78.

Farley, S., A. Brodsky, and C. H. Chen. 2012. A Regression Dependent Iterative Algorithm for Optimizing Top-K Selection in Simulation Query Language. *International Journal of Decision Support System Technology*, 4(3), 12-24.

Fu, M.C. 2006. Gradient Estimation. Chapter 19 in Handbooks in Operations Research and Management Science: Simulation, S.G. Henderson and B.L. Nelson, eds., Elsevier,575–616, 2006.

Fu, M.C. 2008. What You Should Know About Simulation and Derivatives (Cover Story). *Naval Research Logistics*, 55(8), 723–736.

Fu, M. C., J. Hu, S. I. Marcus. 2006. Model-based randomized methods for global optimization. *Proceeding of 17th International Symposium on Mathematical Theory of Networks and Systems*, Kyoto, Japan, 355–363.

Fu, M. C., and J. Q. Hu. 1997. Conditional Monte Carlo: Gradient estimation and optimization applications. Kluwer Academic Publishers.

Fu, M. C., J.-Q. Hu, C.H. Chen and X. Xiong. 2007. Simulation Allocation for Determining the Best Design in the Presence of Correlated Sampling. *INFORMS Journal on Computing*, 19(1): 101-111.

Gen, M. and R. Cheng. 2000. Genetic Algorithms and Engineering Optimization, John Wiley & Sons.

Gendreau, M. and J.Y. Potvin. 2010. Handbook of Metaheuristics, International series in operations research and management science,vol 146

Gill, J. 2002. Bayesian Methods: A Social and Behavioural Sciences Approach. Chapman & Hall, Boca Raton, Florida.

Glasserman, P. 1991. Gradient estimation via perturbation analysis. Kluwer Academic Publishers, Boston, Massachusetts.

Glover, F.E. and G.A. Kochenberger. 2003. Handbook of Metaheuristics. Springer ,2003.

Glover, F. and M. Laguna. 1997. Tabu Search, Kluwer Academic Publishers, Norwell, MA.

Glynn, P.W. 1990. Likelihood Ratio Derivative Estimators for Stochastic Systems. *Communications of the ACM - Special issue on simulation* , 33,75-84.

Glynn, P. and S. Juneja. 2004. A large deviations perspective on ordinal optimization. *Proceedings of the 2004 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway, New Jersey. 577–585.

Gu, L. 2001. A Comparison of Polynomial Based Regression Models in Vehicle Safety Analysis. *2001 ASME Design Engineering Technical Conferences - Design Automation Conference,* Pittsburgh, PA, DAC-21063.

Gupta, S. S. 1956. On a decision rule for a problem in ranking means. Doctorial dissertation, Institute of Statistics, University of North Carolina, Chapel Hill, NC.

Gurkan, G. A.Y. Ozge and S.M. Robinson. 1994. Sample Path Optimization in Simulation. *Proceedings of the 1994 Winter Simulation Conference*, 247-254.

Haddock, J. and J. Mittenhall. 1992. Simulation Optimization using Simulated Annealing. *Computers & Industrial Engineering*, 22, 387-395.

Haupt, R. L. and S.E. Haupt. 2004. Practical Genetic Algorithms, 2nd Edition John Wiley & Sons.

He, D., S. E. Chick and C.H. Chen. 2007. Opportunity Cost and OCBA Selection Procedures in Ordinal Optimization for a Fixed Number of Alternative

Systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(5): 951-961.

He, D. H., L. H. Lee, C.H. Chen, M.C.Fu and S.Wasserkug. 2010. Simulation Optimization Using the Cross-Entropy Method with Optimal Computing Budget Allocation. *ACM Transactions on Modeling and Computer Simulation* 20(1), article 4.

Ho, Y.C., and X.R. Cao. 1991. Perturbation analysis and discrete event dynamic systems. Kluwer Academic.

Ho, Y.C., Q.C. Zhao, and Q.S. Jia. 2007. Ordinal Optimization: Soft Optimization for Hard Problems, New York, NY: Springer, 2007.

Hong, L. J. and B. L. Nelson. 2006. Discrete optimization via simulation using COMPASS. *Operations Research*, 54:115-129.

Hong, L. J., B. L. Nelson, and J. Xu. 2010. Speeding up COMPASS for high-dimensional discrete optimization via simulation. *Operations Research Letters*, 38:550-555.

Hsieh, B. W., C. H. Chen and S.C. Chang. 2001. Scheduling semiconductor wafer fabrication by using ordinal optimization-based simulation. *IEEE Transactions on Robotics and Automation* ,17(5): 599-608.

Hsieh, B. W., C. H. Chen and S.C. Chang. 2007. Efficient simulation-based composition of scheduling policies by integrating ordinal optimization with design of experiment. *IEEE Transactions on Automation Science and Engineering* .4(4): 553-568.

Hunter, S. R, C.H. Chen, R. Pasupathy, N.A. Pujowidianto, L.H. Lee, C.M. Yap. 2011. Optimal sampling laws for constrained simulation optimization on finite sets: the bivariate normal case. *Proceeding of 2011 Winter Simulation Conference* , 4294-4302.

Jia, Q.S. 2011. An adaptive sampling algorithm for simulation-based optimization with descriptive complexity preference. *IEEE Transactions on Automation Science and Engineering*,.8(4), pp. 720-731.

Jia, Q.S.. 2013a. Efficient computing budget allocation for simulation-based optimization with stochastic simulation time. *IEEE Transactions on Automatic Control*. 58 (2): 539-544.

Jia, Q.S., J. Shen, Z. Xu, and X Guan. 2012. Simulation-based policy improvement for energy management in commercial office buildings. *IEEE Transactions on Smart Grid*, 3(4), 2211-2223.

Jia, Q.S., E. Zhou and C.H. Chen. 2013b. Efficient computing budget allocation for finding simplest good designs. *IIE Transactions*,45(7), 736-750.

Kiefer, J. and J. Wolfowitz. 1952. Stochastic Estimation of the Maximum of a Regression Function. *Annals of Mathematical Statistics*, 23, 462-466.

Kiefer, J. 1959. Optimum Experimental Designs. *Journal of the Royal Statistical Society: Series B* (*Statistical Methodology*): 21(2) 272-319.

Kim, S. H. and B.L.Nelson. 2007. Recent advance in ranking and selection, *Proceeding of the 2007 winter simulation conference,* Institute of Electrical and Electronics Engineers, Piscataway, New Jersey. 162-172.

Kim, S. H.. and B. L. Nelson. 2006. Selecting the best system, *Handbooks in Operations Research and Management Science: Simulation*, S. G. Henderson and B. L. Nelson, Eds. Oxford: Elsevier Science, 2006, pp.501-534.

Kleijnen, J.P.C. 2008. Response surface methodology for constrained simulation optimization: An overview. *Simulation Modelling Practice and Theory*, 16, 50-64.

Koenig, L. W., A. M. Law. 1985. A procedure for selecting a subset ofsize m containing the l best of k independent normal populations. *Communications in Statistics - Simulation and Computation*, 14 719–734.

Kushner, H. J., and G. G. Yin. 2003. Stochastic Approximation and Recursive Algorithms and Applications. New York, NY.: Springer-Verlag.

Law, A. M., and Kelton, W. D. 1991. Simulation Modeling & Analysis. McGraw-Hill, Inc.

Lee, L. H., C. H Chen, E. P. Chew, J. Li, N. A. Pujowidianto, and S. Zhang. 2010. A Review of Optimal Computing Budget Allocation Algorithms for Simulation Optimization Problem. *International Journal of Operations Research*, 7(2), 19-31.

Lee, L. H., E. P. Chew, S. Teng, D. Goldsman. 2004. Optimal computing budget allocation for multi-objective simulation models. *Proceedings of the 2004 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway, New Jersey, 586-594.

Lee, L. H., E. P. Chew, S.Y. Teng and Y.K. Chen 2008. Multi-objective simulation-based evolutionary algorithm for an aircraft spare parts allocation problem. *European Journal of Operational Research* 189(2): 476-491.

Lee, L. H., E. P. Chew, S.Y.Teng and D. Goldsman. 2010. Finding the non-dominated Pareto set for multi-objective simulation models. *IIE Transactions* 42(9): 656-674.

Lee, L .H., C. Lee and Y.P. Tan. 2007. A Multi-Objective Genetic Algorithm for Robust Flight Scheduling Using Simulation. *European Journal of Operational Research*, 177 (3), 1948-1968.

Lee, L.H N.A. Pujowidianto, L.W. Li, C.H. Chen, C.M. Yap. 2012. Approximate Simulation Budget Allocation for Selecting the Best Design in the Presence of Stochastic Constraints. *IEEE Transactions on Automatic Control*, 57(11): 2940-2945 .

Levendovszky, J., L. T. Bernhofen, E .J. Dudewicz, and E.C. van der Meulen. 1996. Ranking of the best random number generators via entropy-uniformity theory. *American Journal of Mathematical and Management Sciences*,16(1):49-88.

Melas, V. B. 2006. Functional Approach to Optimal Experimental Design. Lecture Notes in Statistics 184. Springer, New York.

Ólafsson, S. and J. Kim. 2002. Simulation optimization. *Proceedings of the 2002 Winter Simulation Conference*,79-84.

Peng, Y., C. H. Chen, M. C. Fu, and J. Q. Hu. 2013. Efficient Simulation Resource Sharing and Allocation for Selecting the Best. *IEEE Transactions on Automatic Control*, 58 (4): 1017-1023.

Pujowidianto, N.A. L.H Lee, L.W.Li, C.H. Chen, C.M.Yap. 2009. Optimal Computing Budget Allocation for Constrained Optimization. *Proceedings of the 2009 Winter Simulation Conference*: 584-589.

Pujowidianto, N.A., S. R. Hunter, R. Pasupathy, L.H. Lee, C.H. Chen. 2012. Closed-form sampling laws for stochastically constrained simulation optimization on large finite sets. *Proceedings of the 2012 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway, New Jersey, Article No. 14.

Pujowidianto,N.A., S. R. Hunter, R.Pasupathy, L.H. Lee, C.H.Chen. 2013. On Approximating Optimal Sampling Laws for Stochastically Constrained Simulation Optimization on Large Finite Sets. Working Paper.

Robbins, H. and S. Monro. 1951. A Stochastic Approximation Method. *Annals of Mathematical Statistics*, 22,400-407.

Robinson, S.M. 1996. Analysis of Sample-Path Optimization, *Mathematics of Operations Research*, 21, 513-528.

Rubinstein, R.Y. and A. Shapiro. 1993. Discrete Event Systems: Sensitivity Analysis and Stochastic Approximation using the Score Function Method, John Wiley & Sons, New York.

Shi, L. and S. Ólafsson. 1997. An Integrated Framework for Deterministic and Stochastic Optimization. *Proceedings of the 1997 Winter Simulation Conference*, 358-365.

Spall, J.C., 1992. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control* 37, 332–341.

Schmidt, C., J. Branke and S.E.Chick 2007. Integrating techniques from statistical ranking into evolutionary algorithms, Applications of Evolutionary Computing. *Lecture Notes in Computer Science* , 3907: 752-763.

Shortle, J. C. H. Chen, B. Crain, A. Brodsky, D. Brod. 2012. Optimal Splitting for Rare-event Simulation. *IIE Transactions*, 44(5), 352-367.

Sivanandam, S. N. and S. N. Deepa. 2010. Introduction to Genetic Algorithm Springer London Limited.

Swisher, J. R., S. H. Jacobson, E. Yücesan. 2003. Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey. *ACM Transations on Modeling and Computer Simulation* 13 134–154.

Szechtman, R., E. Yücesan. 2008. A new perspective on feasibility determination. *Proceedings of the 2008 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway, New Jersey. 273–280.

Trailovic, L. and L. Y. Pao. 2004. Computing budget allocation for efficient ranking and selection of variances with application to target tracking algorithms. *IEEE Transactions on Automatic Control*, 49(1): 58-67.

Teng, S., L. H. Lee, E. P. Chew. 2010. Integration of indifference-zone with multi-objective computing budget allocation. *European Journal of Operational Research* 203(2): 419-429.

Törn, A., A Žilinskas. 1989. Global optimization. G. Goos, J. Hartmanis, eds. Lecture Notes in Computer Science, 350 Springer-Verlag, Berlin.

Waeber, R., P. I. Frazier, and S. G. Henderson. 2010. Performance measures for ranking and selection procedures. *Proceedings of the 2010 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Piscataway, New Jersey, 1235-1245.

Wong, W.P. L.H. Lee and W. Jaruphongsa. 2011. Budget Allocation for Effective Data Collection in Predicting an Accurate DEA Efficiency Score. *IEEE Transactions on Automatic Control*, 56 (6), 1235-1246.

Xiao, H., L.H. Lee and K.M. Ng. 2013. Optimal computing budget allocation for complete ranking. *IEEE Transactions on Automation Science and Engineering*, Digital Object Identifier: 10.1109/TASE.2013.2239289.

Yan, S., E. Zhou, and C. H. Chen.2012. Efficient Selection of a Set of Good Enough Designs with Complexity Preference. *IEEE Transactions on Automation Science and Engineering*, 9(3), 596-606.

Zhang, S., L. H. Lee, C. H. Chen, E. P. Chew, J. X. Li, N. A. Pujowidianto. 2013. Some Efficient Simulation Budget Allocation Rules for Simulation Optimization Problems. to appear in *International Journal of Services Operations and Informatics*.

Zhang, S., L.H. Lee, E.P.C hew, C.H. Chen and H.Y. Jen. 2012. An improved simulation budget allocation procedure to efficiently select the optimal subset of many alternatives. *IEEE International Conference on Automation Science and Engineering (CASE), 2012*, 230-236 .

Zhang, S., P. Chen, L. H. Lee, E.P. Chew and C.H. Chen. 2011. Simulation optimization using the particle swarm optimization with optimal computing budget allocation. *Proceedings of the 2011 Winter Simulation Conference*: 4303-4314.

Zhao, C .M., S.M. Lo, J.A. Lu and Z. Fang. 2004. A simulation approach for ranking of fire safety attributes of existing buildings, *Fire Safety Journal*, 29,557-579.

Appendix A: Proof of Lemma 5.1

From Theorem 5.1, we know that we only need three support points $\{x_{h1}, x_{hs}, x_{hk}\}, 1 < s < k$. We could derive ξ_{hi}^2 explicitly as follows. Firstly, we examine $\mathbf{X_h^t X_h}$.

$$\mathbf{X_h^t X_h} = N_{h\bullet} \begin{pmatrix} 1 & 1 & 1 \\ x_{h1} & x_{hs} & x_{hk} \\ x_{h1}^2 & x_{hs}^2 & x_{hk}^2 \end{pmatrix} \begin{pmatrix} \alpha_{h1} & 0 & 0 \\ 0 & \alpha_{hs} & 0 \\ 0 & 0 & \alpha_{hk} \end{pmatrix} \begin{pmatrix} 1 & x_{h1} & x_{h1}^2 \\ 1 & x_{hs} & x_{hs}^2 \\ 1 & x_{hk} & x_{hk}^2 \end{pmatrix}$$

$$(\mathbf{X_h^t X_h})^{-1} = \frac{1}{N_{h\bullet}} \begin{pmatrix} 1 & x_{h1} & x_{h1}^2 \\ 1 & x_{hs} & x_{hs}^2 \\ 1 & x_{hk} & x_{hk}^2 \end{pmatrix}^{-1} \begin{pmatrix} \alpha_{h1} & 0 & 0 \\ 0 & \alpha_{hs} & 0 \\ 0 & 0 & \alpha_{hk} \end{pmatrix}^{-1} \begin{pmatrix} 1 & 1 & 1 \\ x_{h1} & x_{hs} & x_{hk} \\ x_{h1}^2 & x_{hs}^2 & x_{hk}^2 \end{pmatrix}^{-1}$$

We also know that,

$$\begin{pmatrix} 1 & x_{h1} & x_{h1}^2 \\ 1 & x_{hs} & x_{hs}^2 \\ 1 & x_{hk} & x_{hk}^2 \end{pmatrix}^{-1}$$
$$= \begin{pmatrix} \dfrac{x_{hs}x_{hk}}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} & \dfrac{x_{h1}x_{hk}}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} & \dfrac{x_{h1}x_{hs}}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} \\ \dfrac{-(x_{hs}+x_{hk})}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} & \dfrac{-(x_{h1}+x_{hk})}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} & \dfrac{-(x_{h1}+x_{hs})}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} \\ \dfrac{1}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} & \dfrac{1}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} & \dfrac{1}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} \end{pmatrix}$$

and the inverse of a diagonal matrix is

$$\begin{pmatrix} \alpha_{h1} & 0 & 0 \\ 0 & \alpha_{hs} & 0 \\ 0 & 0 & \alpha_{hk} \end{pmatrix}^{-1} = \begin{pmatrix} 1/\alpha_{h1} & 0 & 0 \\ 0 & 1/\alpha_{hs} & 0 \\ 0 & 0 & 1/\alpha_{hk} \end{pmatrix}$$

Also,

$$\begin{pmatrix} 1 & 1 & 1 \\ x_{h1} & x_{hs} & x_{hk} \\ x_{h1}^2 & x_{hs}^2 & x_{hk}^2 \end{pmatrix}^{-1}$$

$$= \begin{pmatrix} \dfrac{x_{hs}x_{hk}}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} & \dfrac{-(x_{hs}+x_{hk})}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} & \dfrac{1}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} \\ \dfrac{x_{h1}x_{hk}}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} & \dfrac{-(x_{h1}+x_{hk})}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} & \dfrac{1}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} \\ \dfrac{x_{h1}x_{hs}}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} & \dfrac{-(x_{h1}+x_{hs})}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} & \dfrac{1}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} \end{pmatrix}$$

Therefore, the variance ξ_{hi}^2 can be shown to be

$$\xi_{hi}^2 = \sigma_h^2 \mathbf{X_{hi}^t (X_h^t X_h)^{-1} X_{hi}} = \sigma_h^2 \begin{pmatrix} 1 & x_{hi} & x_{hi}^2 \end{pmatrix} \mathbf{(X_h^t X_h)^{-1}} \begin{pmatrix} 1 \\ x_{hi} \\ x_{hi}^2 \end{pmatrix}$$

$$= \frac{\sigma_h^2}{N_{h\bullet}} \left[\frac{E_{hi,1}^2}{\alpha_{h1}} + \frac{E_{hi,s}^2}{\alpha_{hs}} + \frac{E_{hi,k}^2}{\alpha_{hk}} \right]$$

where $E_{hi,1} = \left\{ \dfrac{(x_{hs}-x_{hi})(x_{hk}-x_{hi})}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})} \right\}$, $E_{hi,s} = \left\{ \dfrac{(x_{h1}-x_{hi})(x_{hk}-x_{hi})}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})} \right\}$ and

$E_{hi,k} = \left\{ \dfrac{(x_{h1}-x_{hi})(x_{hs}-x_{hi})}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})} \right\}$.

Since

$$\begin{aligned} & P\left\{ \hat{y}(x_{Bb}) \geq \hat{y}(x_{hi}) \right\} \\ & = P\left\{ \hat{y}(x_{Bb}) \approx v \right\} P\left\{ \hat{y}(x_{hi}) \approx v \right\} \\ & = \int_v P\left\{ \hat{y}(x_{Bb}) \approx v \right\} P\left\{ \hat{y}(x_{hi}) \approx v \right\} dv \\ & = \int_v \exp\left(-N_{B\bullet} I_{Bb}(v)\right) \exp\left(-N_{h\bullet} I_{hi}(v)\right) dv \\ & = \int_v \exp\left(-\beta_B T I_{Bb}(v)\right) \exp\left(-\beta_h T I_{hi}(v)\right) dv \\ & = \inf_v \exp\left(-T\left(\beta_B I_{Bb}(v) + \beta_h I_{hi}(v)\right)\right) \end{aligned}$$

therefore,

$$\lim_{T\to\infty}\frac{1}{T}\ln P\left\{\hat{y}(x_{Bb})\geq\hat{y}(x_{hi})\right\}=\lim_{T\to\infty}\frac{1}{T}\ln\exp\left(-T\left(\beta_B I_{Bb}(v)+\beta_h I_{hi}(v)\right)\right)$$
$$=\inf_{v}\ \beta_B I_{Bb}(v)+\beta_h I_{hi}(v)$$

where $I_{hi}(v)=\dfrac{\left(v-y(x_{hi})\right)^2}{2\sigma_h^2\left[\dfrac{E_{hi,1}^2}{\alpha_{h1}}+\dfrac{E_{hi,s}^2}{\alpha_{hs}}+\dfrac{E_{hi,k}^2}{\alpha_{hk}}\right]}$.

The infimum can be achieved by differentiation,

$$v_{hi}^*=\left(\frac{\beta_B/\xi_{Bb}^2}{\beta_B/\xi_{Bb}^2+\beta_h/\xi_{hi}^2}\right)y(x_{Bb})+\left(\frac{\beta_h/\xi_{hi}^2}{\beta_B/\xi_{Bb}^2+\beta_h/\xi_{hi}^2}\right)y(x_{hi})$$

Therefore, for $1<h<m, h\neq B, 1<i<k$,

$$R_{Bb,hi}(\beta_B,\beta_h,\boldsymbol{\alpha}_\mathbf{B},\boldsymbol{\alpha}_\mathbf{h})=\frac{\left(y(x_{Bb})-y(x_{hi})\right)^2/2}{\dfrac{\sigma_B^2}{\beta_B}\left(\dfrac{E_{Bb,1}^2}{\alpha_{B1}}+\dfrac{E_{Bb,s}^2}{\alpha_{Bs}}+\dfrac{E_{Bb,k}^2}{\alpha_{Bk}}\right)+\dfrac{\sigma_h^2}{\beta_h}\left(\dfrac{E_{hi,1}^2}{\alpha_{h1}}+\dfrac{E_{hi,s}^2}{\alpha_{hs}}+\dfrac{E_{hi,k}^2}{\alpha_{hk}}\right)}$$

$$R_{Bb,Bi}(\beta_B,\boldsymbol{\alpha}_\mathbf{B})=\frac{\left(y(x_{Bb})-y(x_{Bi})\right)^2/2}{\dfrac{\sigma_B^2}{\beta_B}\left(\dfrac{E_{Bb,1}^2}{\alpha_{B1}}+\dfrac{E_{Bb,s}^2}{\alpha_{Bs}}+\dfrac{E_{Bb,k}^2}{\alpha_{Bk}}\right)+\dfrac{\sigma_B^2}{\beta_B}\left(\dfrac{E_{Bi,1}^2}{\alpha_{B1}}+\dfrac{E_{Bi,s}^2}{\alpha_{Bs}}+\dfrac{E_{Bi,k}^2}{\alpha_{Bk}}\right)}$$

Appendix B: Proof of Lemma 5.2

Under assumption 5.1, $(x_{hs}-x_{hi})=(s-i)*d$ where d is the distance between consecutive design locations. As a result,

$$E_{hi,1}^{2}=\left(\frac{(x_{hs}-x_{hi})(x_{hk}-x_{hi})}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})}\right)^{2}=\left(\frac{(s-i)(k-i)}{(s-1)(k-1)}\right)^{2},s,i\in\{2,3,...,k-1\}.$$

Similarly, we can conclude that

$$E_{hi,s}^{2}=\left(\frac{(i-1)(i-k)}{(s-1)(s-k)}\right)^{2},E_{hi,k}^{w}=\left(\frac{(i-1)(i-s)}{(k-1)(k-s)}\right)^{2}.$$

From Theorem 5.1, we must have three support points for a quadratic regression. Therefore, $\alpha_{h1},\alpha_{hs},\alpha_{hk}$ must be strictly positively and never go to zero, i.e., there always exists $\kappa>0$ such that $\min\{\alpha_{h1},\alpha_{hs},\alpha_{hk}\}>\kappa$. Therefore,

$$\left(\frac{E_{hi,1}^{2}}{\alpha_{h1}}+\frac{E_{hi,s}^{2}}{\alpha_{hs}}+\frac{E_{hi,k}^{2}}{\alpha_{hk}}\right)=\frac{\left(\frac{(s-i)(k-i)}{(s-1)(k-1)}\right)^{2}}{\alpha_{h1}}+\frac{\left(\frac{(i-1)(i-k)}{(s-1)(s-k)}\right)^{2}}{\alpha_{hs}}+\frac{\left(\frac{(i-1)(i-s)}{(k-1)(k-s)}\right)^{2}}{\alpha_{hk}}$$

is always bounded below by

$$\left(\left(\frac{(s-i)(k-i)}{(s-1)(k-1)}\right)^{2}+\left(\frac{(i-1)(i-k)}{(s-1)(s-k)}\right)^{2}+\left(\frac{(i-1)(i-s)}{(k-1)(k-s)}\right)^{2}\right)$$

and bounded above by

$$\left(\left(\frac{(s-i)(k-i)}{(s-1)(k-1)}\right)^{2}+\left(\frac{(i-1)(i-k)}{(s-1)(s-k)}\right)^{2}+\left(\frac{(i-1)(i-s)}{(k-1)(k-s)}\right)^{2}\right)/\kappa.$$

Thus, we conclude that $\left[\frac{E_{hi,1}^2}{\alpha_{h1}}+\frac{E_{hi,s}^2}{\alpha_{hs}}+\frac{E_{hi,k}^2}{\alpha_{hk}}\right]\le C\left[\frac{E_{qj,1}^2}{\alpha_{q1}}+\frac{E_{qj,s}^2}{\alpha_{qs}}+\frac{E_{qj,k}^2}{\alpha_{qk}}\right]$.

Appendix C: Proof of Theorem 5.2

(1)We can rewrite problem (5.19) as follows:

$$\begin{aligned}
&\max \quad z-\gamma\sum_{h=1}^{m}\beta_h = 1\\
s.t.\quad & R_{Bb,Bi}(\beta_B,\boldsymbol{\alpha}_\mathbf{B}) \ge z, \forall i\in\{1,2,...,k\}, i\ne b\\
& R_{Bb,hi}(\beta_B,\beta_h,\boldsymbol{\alpha}_\mathbf{B},\boldsymbol{\alpha}_\mathbf{h}) \ge z, \forall h\in\{1,2,...,m\}, h\ne B; \forall i\in\{1,2,...,k\}\\
& \alpha_{h1}+\alpha_{hs}+\alpha_{hk} = 1, \forall h\in\{1,2,...,m\}\\
& \beta_h,\alpha_{h1},\alpha_{hs},\alpha_{hk} \ge 0, \forall h\in\{1,2,...,m\}
\end{aligned} \tag{C1}$$

Since (C1) is a concave optimization problem, the KKT conditions are necessary and sufficient for optimality. Therefore, there exist $\lambda_{hi}\ge 0$ such that

$$\sum_{i\ne b}\lambda_{Bi}+\sum_{h\ne B}\sum_{i=1}^{k}\lambda_{hi} = 1 \tag{C2}$$

$$\sum_{i=1}^{k}\lambda_{hi}\frac{\partial R_{Bb,hi}(\beta_B^*,\beta_h^*,\boldsymbol{\alpha}_\mathbf{B}^*,\boldsymbol{\alpha}_\mathbf{h}^*)}{\partial\beta_h} = \gamma, \forall h\ne B \tag{C3}$$

$$\sum_{i\ne b}\lambda_{Bi}\frac{\partial R_{Bb,Bi}(\beta_B^*,\boldsymbol{\alpha}_\mathbf{B}^*)}{\partial\beta_B}+\sum_{h\ne B}\sum_{i=1}^{k}\lambda_{hi}\frac{\partial R_{Bb,hi}(\beta_B^*,\beta_h^*,\boldsymbol{\alpha}_\mathbf{B}^*,\boldsymbol{\alpha}_\mathbf{h}^*)}{\partial\beta_B} = \gamma \tag{C4}$$

$$\lambda_{hi}\left(z-R_{Bb,hi}(\beta_B^*,\beta_h^*,\boldsymbol{\alpha}_\mathbf{B}^*,\boldsymbol{\alpha}_\mathbf{h}^*)\right) = 0, \forall h\in\{1,2,...,m\}, h\ne B; \forall i\in\{1,2,...,k\} \tag{C5}$$

$$\lambda_{Bi}\left(z-R_{Bb,Bi}(\beta_B^*,\boldsymbol{\alpha}_\mathbf{B}^*)\right) = 0, \forall i\in\{1,2,...,k\}, i\ne b \tag{C6}$$

where (C3) and (C4) are the stationary conditions and (C5) and (C6) are the complementary slackness conditions.

Suppose $\gamma = 0$. Since $\partial R_{Bb,hi}(\bullet)/\partial\beta_h > 0, \forall h$,therefore $\lambda_{hi} = 0, \forall h, \forall i$. However, this contradicts with equation (C2) which indicates that at least $\lambda_{hi} > 0$ for some h and i. This concludes that γ must be strictly positive.

For any $\forall h \in \{1, 2, ..., m\}, h \neq B$, if all $\lambda_{hi} = 0, \forall i$, it will lead to $\gamma = 0$, which is not true as shown just now. Therefore, we can conclude that there exists at least a $i_h \in \{1, 2, ..., k\}$ such that $\lambda_{hi_h} > 0$. From $\lambda_{hi_h} > 0$ and the complementary slackness condition (C5), we can conclude with the following equation:

$$z = R_{Bb,hi_h}(\beta_B^*, \beta_h^*, \boldsymbol{\alpha}_{\mathbf{B}}^*, \boldsymbol{\alpha}_{\mathbf{h}}^*) = R_{Bb,qi_q}(\beta_B^*, \beta_h^*, \boldsymbol{\alpha}_{\mathbf{B}}^*, \boldsymbol{\alpha}_{\mathbf{q}}^*), \forall h, q \in \{1, 2, ..., m\}, h \neq B \quad \text{(C7)}$$

which is equivalent to

$$\begin{aligned} &\frac{\left(y(x_{hi_h}) - y(x_{Bb})\right)^2 / 2}{\frac{\sigma_B^2}{\beta_B^*}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*} + \frac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right) + \frac{\sigma_h^2}{\beta_h^*}\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*} + \frac{E_{hi_h,s}^2}{\alpha_{hs}^*} + \frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)} \\ &= \frac{\left(y(x_{qi_q}) - y(x_{Bb})\right)^2 / 2}{\frac{\sigma_B^2}{\beta_B^*}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*} + \frac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right) + \frac{\sigma_q^2}{\beta_q^*}\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*} + \frac{E_{qi_q,s}^2}{\alpha_{qs}^*} + \frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)} \end{aligned} \quad \text{(C8)}$$

Equation (C8) can be rearranged as

$$\begin{aligned}&\frac{\left(y(x_{hi_h})-y(x_{Bb})\right)^2\sigma_B^2}{\beta_B^*}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)\\&+\frac{\left(y(x_{hi_h})-y(x_{Bb})\right)^2\sigma_q^2}{\beta_q^*}\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)\\&=\frac{\left(y(x_{qi_q})-y(x_{Bb})\right)^2\sigma_B^2}{\beta_B^*}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)\\&+\frac{\left(y(x_{qi_q})-y(x_{Bb})\right)^2\sigma_h^2}{\beta_h^*}\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)\end{aligned}\tag{C9}$$

Case (a): If $y(x_{hi_h})>y(x_{qi_q})$, i.e., $\left(y(x_{hi_h})-y(x_{Bb})\right)^2>\left(y(x_{qi_q})-y(x_{Bb})\right)^2$, equation (C9) yields

$$\begin{aligned}&\frac{\left(y(x_{hi_h})-y(x_{Bb})\right)^2\sigma_q^2}{\beta_q^*}\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)\\&\le\frac{\left(y(x_{qi_q})-y(x_{Bb})\right)^2\sigma_h^2}{\beta_h^*}\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)\end{aligned}$$

and it can be rearranged as

$$\begin{aligned}\frac{\beta_h^*}{\beta_q^*}&\le\frac{\left(y(x_{qi_q})-y(x_{Bb})\right)^2\sigma_h^2}{\left(y(x_{hi_h})-y(x_{Bb})\right)^2\sigma_q^2}\frac{\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)}{\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)}\\&\le\frac{\left(V_U-V_L\right)^2\sigma_h^2}{\Delta^2\sigma_q^2}\frac{\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)}{\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)}\\&\le\frac{\left(V_U-V_L\right)^2\sigma_h^2}{\Delta^2\sigma_q^2}\left(\left(\frac{(s-i_h)(k-i_h)}{(s-1)(k-1)}\right)^2+\left(\frac{(i_h-1)(i_h-k)}{(s-1)(s-k)}\right)^2+\left(\frac{(i_h-1)(i_h-s)}{(k-1)(k-s)}\right)^2\right)/\kappa\end{aligned}\tag{C10}$$

where $s, i_h \in \{2,3,...,k-1\}$ and the second last inequality follows from assumption 5.2 and the last inequality follows from Lemma 5.2.

Case (b): If $y(x_{hi_h}) \le y(x_{qi_q})$, i.e., $y(x_{hi_h}) < y(x_{qi_q})$, rearrange (C9) to be the following equation:

$$\begin{aligned}
&\frac{\left(y(x_{hi_h}) - y(x_{Bb})\right)^2 \sigma_q^2}{\beta_q^*}\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*} + \frac{E_{qi_q,s}^2}{\alpha_{qs}^*} + \frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right) \\
&= \frac{\left(\left(y(x_{qi_q}) - y(x_{Bb})\right)^2 - \left(y(x_{hi_h}) - y(x_{Bb})\right)^2\right)\sigma_B^2}{\beta_B^*}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*} + \frac{E_{Bb,s}^2}{\alpha_{Bs}^*} + \frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right) \quad \text{(C11)} \\
&\quad + \frac{\left(y(x_{qi_q}) - y(x_{Bb})\right)^2 \sigma_h^2}{\beta_h^*}\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*} + \frac{E_{hi_h,s}^2}{\alpha_{hs}^*} + \frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)
\end{aligned}$$

Substitute equation (C3) into equation (C4),

$$\frac{\sum_{i \ne b} \lambda_{Bi} \frac{\partial R_{Bb,Bi}(\beta_B^*, \boldsymbol{\alpha}_\mathbf{B}^*)}{\partial \beta_B}}{\gamma} + \sum_{h \ne B} \sum_{i=1}^{k} \frac{\lambda_{hi} \frac{\partial R_{Bb,hi}(\beta_B^*, \beta_h^*, \boldsymbol{\alpha}_\mathbf{B}^*, \boldsymbol{\alpha}_\mathbf{h}^*)}{\partial \beta_B}}{\sum_{i=1}^{k} \lambda_{hi} \frac{\partial R_{Bb,hi}(\beta_B^*, \beta_h^*, \boldsymbol{\alpha}_\mathbf{B}^*, \boldsymbol{\alpha}_\mathbf{h}^*)}{\partial \beta_h}} = 1. \qquad \text{(C12)}$$

Therefore,

$$\sum_{h \ne B} \frac{\sum_{i=1}^{k} \lambda_{hi} \frac{\partial R_{Bb,hi}(\beta_B^*, \beta_h^*, \boldsymbol{\alpha}_\mathbf{B}^*, \boldsymbol{\alpha}_\mathbf{h}^*)}{\partial \beta_B}}{\sum_{i=1}^{k} \lambda_{hi} \frac{\partial R_{Bb,hi}(\beta_B^*, \beta_h^*, \boldsymbol{\alpha}_\mathbf{B}^*, \boldsymbol{\alpha}_\mathbf{h}^*)}{\partial \beta_h}} \le 1. \qquad \text{(C13)}$$

It can be shown that

$$\frac{\partial R_{Bb,hi}(\beta_B^*,\beta_h^*,\boldsymbol{\alpha}_{\mathbf{B}}^*,\boldsymbol{\alpha}_{\mathbf{h}}^*)/\partial\beta_B}{\partial R_{Bb,hi}(\beta_B^*,\beta_h^*,\boldsymbol{\alpha}_{\mathbf{B}}^*,\boldsymbol{\alpha}_{\mathbf{h}}^*)/\partial\beta_h}=\frac{\sigma_B^2\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)/\beta_B^{*2}}{\sigma_h^2\left(\frac{E_{hi,1}^2}{\alpha_{h1}^*}+\frac{E_{hi,s}^2}{\alpha_{hs}^*}+\frac{E_{hi,k}^2}{\alpha_{hk}^*}\right)/\beta_h^{*2}}.$$

Therefore, the inequality (C13) can be reduced to

$$\sum_{h\neq B}\frac{\sigma_B^2/\beta_B^{*2}}{\sigma_h^2/\beta_h^{*2}}\frac{\sum_{i=1}^{k}\lambda_{hi}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)}{\sum_{i=1}^{k}\lambda_{hi}\left(\frac{E_{hi,1}^2}{\alpha_{h1}^*}+\frac{E_{hi,s}^2}{\alpha_{hs}^*}+\frac{E_{hi,k}^2}{\alpha_{hk}^*}\right)}\leq 1. \qquad \text{(C14)}$$

From Lemma 5.1, there always exist $C_{hi}>0$ such that

$$\left(\frac{E_{Bb,1}^2}{\alpha_{B1}}+\frac{E_{Bb,s}^2}{\alpha_{Bs}}+\frac{E_{Bb,k}^2}{\alpha_{Bk}}\right)\Big/\left(\frac{E_{hi,1}^2}{\alpha_{h1}}+\frac{E_{hi,s}^2}{\alpha_{hs}}+\frac{E_{hi,k}^2}{\alpha_{hk}}\right)\leq C_{hi}.$$

Take $C=\min\left(C_{hi}\right)$. Hence,

$$\sum_{h\neq B}\frac{\sigma_B^2/\beta_B^{*2}}{\sigma_h^2/\beta_h^{*2}}\leq 1/C. \qquad \text{(C15)}$$

Therefore, $\frac{1}{\beta_B}\leq\frac{1}{\beta_h}\sqrt{\frac{\sigma_h^2}{C\sigma_B^2}}$. Equation (C11) can be reduced to

$$\begin{aligned}
&\frac{\left(y(x_{hi_h})-y(x_{Bb})\right)^2\sigma_q^2}{\beta_q^*}\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)\\
&\leq\frac{\left(\left(y(x_{qi_q})-y(x_{Bb})\right)^2-\left(y(x_{hi_h})-y(x_{Bb})\right)^2\right)\sigma_B^2}{\beta_h^*}\sqrt{\frac{\sigma_h^4}{C^2\sigma_B^4}}\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)\\
&\quad+\frac{\left(y(x_{qi_q})-y(x_{Bb})\right)^2\sigma_h^2}{\beta_h^*}\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)
\end{aligned}$$

Therefore, we conclude that

$$\frac{\beta_h^*}{\beta_q^*} \le \frac{\sigma_B^2\left(\frac{E_{Bb,1}^2}{\alpha_{B1}^*}+\frac{E_{Bb,s}^2}{\alpha_{Bs}^*}+\frac{E_{Bb,k}^2}{\alpha_{Bk}^*}\right)\left(\delta_{qi_q}^2-\delta_{hi_h}^2\right)\sqrt{\left(\frac{\sigma_h^2}{C\sigma_B^2}\right)^2}}{\delta_{hi_h}^2\sigma_q^2\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)}$$

$$+\frac{\delta_{qi_q}^2\sigma_h^2\left(\frac{E_{hi_h,1}^2}{\alpha_{h1}^*}+\frac{E_{hi_h,s}^2}{\alpha_{hs}^*}+\frac{E_{hi_h,k}^2}{\alpha_{hk}^*}\right)}{\delta_{hi_h}^2\sigma_q^2\left(\frac{E_{qi_q,1}^2}{\alpha_{q1}^*}+\frac{E_{qi_q,s}^2}{\alpha_{qs}^*}+\frac{E_{qi_q,k}^2}{\alpha_{qk}^*}\right)}$$

$$\le \frac{\left(\sigma_B^2 K_{Bb}\left(\delta_{qi_q}^2-\delta_{hi_h}^2\right)\sqrt[3]{\frac{\sigma_h^4}{C^2\sigma_B^4}}+\delta_{qi_q}^2\sigma_h^2 K_{hi_h}\right)}{\delta_{hi_h}^2\sigma_q^2 K_{hi_q}}$$

where $\delta_{hi_h}^2=\left(y(x_{hi_h})-y(x_{Bb})\right)^2, \delta_{qi_q}^2=\left(y(x_{qi_q})-y(x_{Bb})\right)^2$

$$K_{Bb}=\left(\left(\frac{(s-b)(k-b)}{(s-1)(k-1)}\right)^2+\left(\frac{(b-1)(b-k)}{(s-1)(s-k)}\right)^2+\left(\frac{(b-1)(b-s)}{(k-1)(k-s)}\right)^2\right)/\kappa$$

$$K_{hi_h}=\left(\left(\frac{(s-i_h)(k-i_h)}{(s-1)(k-1)}\right)^2+\left(\frac{(i_h-1)(i_h-k)}{(s-1)(s-k)}\right)^2+\left(\frac{(i_h-1)(i_h-s)}{(k-1)(k-s)}\right)^2\right)/\kappa$$

$$K_{hi_q}=\left(\left(\frac{(s-i_q)(k-i_q)}{(s-1)(k-1)}\right)^2+\left(\frac{(i_q-1)(i_q-k)}{(s-1)(s-k)}\right)^2+\left(\frac{(i_q-1)(i_q-s)}{(k-1)(k-s)}\right)^2\right)/\kappa\,.$$

(2) Let $\omega=\arg\min_{h\neq q}\beta_q^*$. According to (1), there must exist $c>0$ such that $\beta_h^*\le c\beta_w^*,\ \forall h\neq B$. Thus, (2) is true.

(3) From (1), we see that $\beta_h^* / c \le \beta_q^*, \forall h, q \ne B$. Then,

$\beta_1^* + (m-1)\beta_h^* / c \le \sum_{q=1}^{m} \beta_q^* = 1$ and hence, $\beta_h^* = O(\frac{1}{m-1}), \forall h \ne B$ as $m \to \infty$.

(4) From (C15) we know that $\sum_{h \ne B} \left(\frac{\sigma_B^2 / \beta_B^2}{\sigma_h^2 / \beta_h^2} \right) \le \frac{1}{C}$. Define $\eta = \arg\min_{h \ne B} \frac{\sigma_B^2 / \beta_B^2}{\sigma_h^2 / \beta_h^2}$,

and therefore, $\frac{\sigma_B^2 / \beta_B^2}{\sigma_\eta^2 / \beta_\eta^2} \le \frac{1}{C(m-1)}$. Thus, $\frac{\beta_\eta}{\beta_B} \le \sqrt{\frac{\sigma_\eta^2}{C\sigma_B^2}} \sqrt{\frac{1}{(m-1)}}$.

Therefore, $\frac{\beta_\eta}{\beta_B} \to 0$ as $m \to \infty$. From Theorem 5.2 (2), we see that $\beta_h^* \le c\beta_{\min}^*, \forall h \ne B,\ \beta_{\min}^* = \min_q \left\{ \beta_q^* \right\}$ and we can conclude that $\frac{\beta_h}{\beta_B} \to 0$ for any $h \ne B$ □ .