BAYESIAN DESIGNS FOR SEQUENTIAL LEARNING PROBLEMS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Jing Xie

May 2014

BAYESIAN DESIGNS FOR SEQUENTIAL LEARNING PROBLEMS

Jing Xie, Ph.D.

Cornell University 2014

We consider the Bayesian formulation of a number of learning problems, where we focus on sequential sampling procedures for allocating simulation effort efficiently.

We derive Bayes-optimal policies for the problem of multiple comparisons with a known standard, showing that they can be computed efficiently when sampling is limited by probabilistic termination or sampling costs. We provide a tractable method for computing upper bounds on the Bayes-optimal value of a ranking and selection problem, which enables evaluation of optimality gaps for existing ranking and selection procedures. Applying techniques from optimal stopping, multi-armed bandits and Lagrangian relaxation, we are able to efficiently solve the corresponding dynamic programs.

We develop a new value-of-information-based procedure for the problem of Bayesian optimization via simulation, which incorporates both correlated prior beliefs and correlated sampling distributions. We also introduce a sequential Bayesian algorithm for optimization of expensive functions under low-dimensional input uncertainties. These implementations take advantage of machine learning tools that enable exploring combinatorially large solution spaces, or estimating expectations of simulation output variables with random inputs.

We present theoretical results characterizing the proposed procedures, compare them numerically against previously developed or standard benchmarking procedures, and apply them to applications in emergency services, manufacturing, and health care.

BIOGRAPHICAL SKETCH

Jing Xie was born in 1988 in the city of Anqing, Anhui province, China. With a born talent in music and maths, Jing's childhood was fulfilled with beautiful melodies from her violin, as well as tricky but interesting math problems.

At the age of 18, Jing went to Fudan University in Shanghai, China for undergraduate studies majored in Mathematics. Thanks to the advice and support from her beloved father, who was a math teacher, she decided to pursue further studies abroad after receiving her Bachelor's degree. Yet because of this goal, her years at Fudan were packed with hard-working days and nights.

Luckily, all her efforts were paid back in March 2009, when she was admitted to her dream Ph.D. program at the School of Operations Research and Information Engineering at Cornell University. She then joined this program, received a Master's degree in Operations Research in 2012, and completed her doctoral dissertation on simulation-based sequential Bayesian algorithms, under the guidance of her advisor Prof. Peter Frazier. She had a wonderful time living in Ithaca, a beautiful town with lots of waterfalls and snows. Her travels to Caribbean, Alaska, Yellowstone, etc., were also full of joyful and memorable moments.

Upon graduation from Cornell University with her Ph.D., Jing is excited to begin her career at a financial services firm in New York City, where she has the opportunity to leverage her capabilities, and develop her potential and skills.

To my parents, Yue Xie and Hanmei Zhang.

ACKNOWLEDGEMENTS

I would never have been able to complete this dissertation without the help and support from many people around me.

I would like to express my deepest gratitude to my advisor, Prof. Peter Frazier, for his exceptional guidance, caring and patience at all times, and for the incredibly relaxed and delightful atmosphere he provided me for doing research. His encouragement and trust always motivated me to go forward. His knowledge and insights shielded me away from detours, and led me through all the obstacles.

I am also very grateful to my other committee members, Prof. Shane Henderson and Prof. Bruce Turnbull, for their recognition and advice on my work. Special thanks also go to Prof. Stephen Chick, for his sparkling ideas and untiring support, and to Sethuraman Sankaran, Abhay Ramachandra, and Prof. Alison Marsden, for their help on our fruitful collaboration.

I would like to thank the entire faculty at the School of Operations Research and Information Engineering at Cornell University. Their vivid classes, inspiring talks, and lovely smiles are lifetime memories of mine. I would also like to thank all the colleagues in my school, and everyone I met at Cornell, for their friendship and company that warmed my heart when I was weak and tired.

Finally, I owe a very important debt to my parents, Yue Xie and Hanmei Zhang, whose love and support have always been selfless and unconditional. As their only child, I deeply appreciate their parenting wisdom and energy, their high expectations, and their belief in the value of learning.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

We consider a discrete and finite set of alternative systems whose performance can only be evaluated via stochastic simulation. Our goal is to efficiently allocate a limited simulation budget among the alternatives, to find one whose expected performance (sometimes under environmental uncertainties) is as large as possible, or to find a subset of them with expected performance exceeding a threshold of known value. These problems are called "discrete optimization via simulation" (DOvS) and "multiple comparisons with a known standard" (MCS) respectively.

We focus on how simulation effort should be allocated among the alternative systems to best support selection or comparison of alternatives when sampling stops. The most straightforward approach for allocating sampling effort, and the approach most commonly employed by practitioners, is to simulate each system an equal number of times. This is inefficient, because some alternatives can be immediately established as being sub-optimal (in DOvS), or as being substantially far from the threshold (in MCS), after only a few samples, and hence we should focus our later effort on the other alternatives that need many more samples before an accurate determination can be made.

To design a strategy that samples more efficiently, we employ a Bayesian statistical approach. The Bayesian framework allows us to formulate the problem of allocating effort as a problem in sequential decision-making under uncertainty, and to design fully sequential and adaptive sampling procedures. While Bayesian experimental designs need careful construction of priors and may require more computation to decide where to sample, they offer significant advantage in analyzing a limited amount or a dynamic sequence of data, and in limiting simulation effort. For the MCS problem, we provide Bayes-optimal sampling procedures by efficiently solving a dynamic program. For

DOvS, we propose one-step lookahead sampling procedures.

We introduce several key concepts and review the related literature in Section 1.1 - 1.6. In Section 1.7 we describe the organization of this thesis.

1.1 Ranking and Selection

In a ranking and selection (R&S) problem, we wish to select the best among several competing alternatives. We measure the quality of an alternative by its expected performance, and use simulation samples to estimate this expected performance. We typically make no structural assumptions on the relationships between the values of the alternatives, and the number of alternatives considered is typically small (less than 1000).

Our goal in R&S is to allocate our simulation sampling effort efficiently among the alternatives, so as to accurately determine which alternative has the largest expected performance, while at the same time limiting simulation effort.

This problem has been considered by many authors, under four distinct mathematical formulations. We specifically consider the Bayesian formulation in this thesis, for which early work dates to Raiffa and Schlaifer (1968), with recent surveys Chick (2006) and Frazier (2012). The other mathematical formulations of the problem are the indifference-zone formulation (see the handbook Bechhofer et al. (1968), the monograph Bechhofer et al. (1995), and the survey Kim and Nelson (2006)); the optimal computing budget allocation, or OCBA (Chen and Lee 2010); and the large-deviations approach (Glynn and Juneja 2004).

In the Bayesian formulation of the R&S problem, we place a prior distribution on the unknown true expected performance of each alternative, and our goal is to design

an algorithm for allocating simulation effort with good average-case performance under the prior. While some work in this area, such as Raiffa and Schlaifer (1968), Chick and Inoue (2001b), and Chick and Inoue (2001a), considers two-stage algorithms, much of the recent work, such as Gupta and Miescke (1996), Chick et al. (2010), Frazier et al. (2008), Chick and Gans (2009), and Chick and Frazier (2012), has focused on sequential procedures, whose allocations of sampling effort are potentially more responsive to previous samples, and thus promise greater efficiency.

Bayes-optimal sequential R&S procedures are characterized by the dynamic programming equations, and given sufficient computational power, can be computed by solving these equations. These equations have been used to compute Bayes-optimal procedures for problems with one alternative of unknown value and one of known value (Chick and Gans 2009, Chick and Frazier 2012), and for problems with two alternatives of unknown value (Frazier et al. 2008). However, for problems with more than a few alternatives, solving these dynamic programming equations becomes computationally infeasible, due to the curse of dimensionality (Powell 2007).

1.2 Feasibility Determination

Procedures for variants of the MCS problem have been developed under several frequentist formulations of R&S, where the objective is to select in the presence of a stochastic constraint, the set of feasible systems, or the best system. They also refer to the problem as "feasibility determination".

In the indifference-zone R&S literature, Paulson (1962) and Kim (2005) provide fully sequential procedures for determining if any of the systems is better than the standard, and if so, selecting the one with the best mean. Bechhofer and Turnbull (1978),

Nelson and Goldsman (2001) provide two-stage procedures for the same task. Andradóttir et al. (2005) and Andradóttir and Kim (2010) consider the problem of finding the system with the best expected primary performance measure, subject to a constraint on the expected value of a secondary performance measure. They present a procedure that determines systems' feasibility in the presence of a stochastic constraint, and combines this procedure with a finding-the-best procedure to identify the best feasible system. Batur and Kim (2010) provides fully sequential procedures for identifying a set of feasible or near-feasible systems given multiple stochastic constraints. Healey et al. (2012) extends these procedures to selection of the best feasible system under multiple constraints while allowing correlated sampling.

Within a large deviations framework, which was introduced for R&S by Glynn and Juneja (2004), Szechtman and Yücesan (2008) considers MCS, or feasibility determination. That chapter characterizes the fractional allocation that maximizes the asymptotic rate of decay of the expected number of incorrect determinations, and presents a stochastic approximation algorithm that yields a budget allocation that provably converges to the optimal one. While this is appealing, Glynn and Juneja (2011) argues that such methods, which calculate optimal allocations assuming large deviations rate functions are known, and then plug in estimates of the rate functions, may have rates of convergence that are significantly worse than desired because of the difficulty of estimating rate functions. This difficulty can be avoided when sampling distributions are bounded, but the difficulty remains when sampling distributions are unbounded. Hunter et al. (2011), on the other hand, considers selecting an optimal system from among a finite set of competing systems, based on a stochastic objective function and subject to multiple stochastic constraints. Hunter and Pasupathy (2012) then assumes a bivariate normal distribution for the objective and (single) constraint performance measures, and gives explicit and asymptotically exact characterizations.

1.3 Discrete Optimization via Simulation

In DOvS, we have a discrete set of alternative systems whose performance can each be evaluated via stochastic simulation, and we wish to allocate a limited simulation budget among them to find one whose expected performance is as large as possible.

Because of its importance, previous authors have proposed algorithms of several types to address the DOvS problem, including randomized search (Andradóttir 1998, 2006, Zhou et al. 2008), metaheuristics (Shi and Ólafsson 2000), metamodel-based algorithms (Barton 2009, van Beers and Kleijnen 2008), Bayesian value-of-information algorithms (Chick 2006, Frazier et al. 2010), local search algorithms (Wang et al. 2013, Hong and Nelson 2006, Xu et al. 2010), model-based search (Hu et al. 2012, Wang et al. 2010, Luo and Lim 2013), and R&S algorithms (see reviews in Section 1.1). Henderson and Nelson (2006) and Fu (2002) provide handbook/survey of the field.

Several previous authors have considered Bayesian formulations of optimization via simulation. The setting most frequently studied is that of R&S, with relatively few alternatives, an independent prior distribution, and independent sampling (Gupta and Miescke 1996, Chick and Inoue 2001b, Frazier et al. 2008, Chick and Frazier 2012). Bayesian optimization via simulation with correlated prior distributions for problems with many alternatives was considered in a discrete setting (Frazier et al. 2009) and in a continuous setting (Villemonteix et al. 2009, Huang et al. 2006, Scott et al. 2011). This work in a continuous setting parallels work on noise-free Bayesian global optimization (see Section 1.5).

1.4 VOI Approach and KG Policies

Due to the computational challenge of Bayes-optimal sequential procedures, work in Bayesian DOvS and R&S has focused in large part on developing sub-optimal procedures. Chapter 3 and 4 of this thesis are based upon the value of information (VOI) approach, exemplified by Gupta and Miescke (1996), Frazier et al. (2008), Chick and Gans (2009) and Chick and Frazier (2009), which attempts to manage the trade-off between the consequences of an immediate decision and the cost of additional sampling. In this approach, sampling decisions are made to achieve the highest potential for improving the final selection decision.

Sampling algorithms of this type are called knowledge-gradient (KG) algorithms (Frazier 2009), as they assess the expected value of information in a given increment of sampling. Specifically, we begin with a prior distribution on the values of the sampling means, updating this prior distribution based on sampling information, and use VOI calculations to decide how to best allocate sampling effort – which alternative, or collection of alternatives, would be most valuable to sample next. The advantage of doing so is that making decisions based on the VOI automatically addresses the exploration (evaluating areas of high uncertainty) versus exploitation (evaluating areas of high estimated values) tradeoff, and tends to reduce the number of function evaluations required on average to reach a given solution quality, potentially (but not necessarily) at the cost of requiring more computation to decide where to sample.

One-step lookahead sampling procedures like KG are also commonly used in Bayesian global optimization (see Section 1.5), where finding the optimal policy is usually considered intractable.

1.5 Optimization of Expensive Functions

In many applications, our goal is to find the global optimum of a computationally expensive continuous function. Bayesian optimization is one of the most efficient methods to achieve this goal, for which early work dates back to Kushner (1964) (see Santner et al. (2003), Brochu et al. (2009) for summaries). By taking into account both exploration and exploitation, it reduces the number of function evaluations required to find a good solution (Mockus 1994, Jones et al. 1998, Jones 2001, Sasena 2002).

Bayesian (global) optimization is applicable when the objective is a black-box function with no closed-form expressions, of which one can only obtain (possibly noisy) observations at sampled values. Bayesian optimization techniques combine a prior belief about the objective function with observed evidence to derive a posterior, which serves as a response surface, or a surrogate function (Forrester et al. 2008), that can be searched quickly and extensively to determine where to sample next. These techniques are particularly useful when samples are expensive or time-consuming, when derivatives are not accessible, and when the underlying function is non-concave.

While we focus on discrete problems in this thesis, assumptions and methodologies adopted in coutinuous Bayesian optimization can also be applied, if our discrete alternatives are embedded in a continuous space.

Within the recent literature on Bayesian optimization, our work is closely related to Efficient Global Optimization, or EGO (Sasena 2002, Bartz-Beielstein et al. 2005, Huang et al. 2006, Lizotte et al. 2007, Hutter 2009), which applies Gaussian processes (see Section 1.6) and the sequential expected improvement approach (Jones et al. 1998, Ginsbourger et al. 2008) to derivative-free optimization and experimental design. It differs from KG in that it attempts to maximize the expected improvement in the value

of the best *observed* point, while KG looks at the expected improvement in the best estimated *overall* value, which might be somewhere we have not measured.

1.6 Gaussian Process Priors

Our Bayesian methods, in their use of multivariate normal prior distributions, makes a link to Gaussian process (GP) priors (Mockus 1994, Rasmussen and Williams 2006) and stochastic kriging (Ankenman et al. 2010, Chen et al. 2012, 2013).

GP priors are frequently used in Bayesian global optimization (Kushner 1964, Mockus 1989, Jones et al. 1998), where people use such priors to model their belief about an implicit continuous function over $\mathbb{R}^d$ that closer arguments are more likely to correspond to similar values. Previous work has demonstrated that the correlations in a GP prior are extremely important for reducing the number of samples needed to evaluate an expensive function, because they allow us to learn about areas that have not been measured from those that have.

When alternatives correspond to points on a grid, as they do in many integer-ordered problems like resource allocation (e.g., each alternative specifies the number of each of several employee types to have present), our use of a multivariate normal prior distribution can be implemented by placing a GP prior over the continuum, and then only considering points on the grid.

1.7 Thesis Organization

We now briefly summarize the contents of each chapter.

Chapter 2

Chapter 2 considers the problem of efficiently allocating simulation effort to determine which of several simulated systems have mean performance exceeding a threshold of known value. Within a Bayesian formulation of this problem, the optimal fully sequential policy for allocating simulation effort is the solution to a dynamic program. When sampling is limited by probabilistic termination or sampling costs, we show that this dynamic program can be solved efficiently, providing a tractable way to compute the Bayes-optimal policy. The solution uses techniques from optimal stopping and multi-armed bandits. We then present further theoretical results characterizing this Bayes-optimal policy, compare it numerically to several approximate policies, and apply it to applications in emergency services and manufacturing.

A version of Chapter 2 appeared as Xie and Frazier (2013a), which won the 2013 INFORMS Computing Society Student Paper Award, and was a finalist in the 2011 INFORMS Junior Faculty Interest Group Paper Competition.

Chapter 3

Chapter 3 addresses discrete optimization via simulation. We show that allowing for both a correlated prior distribution on the means (e.g., with discrete kriging models) and sampling correlation (e.g., with common random numbers, or CRN) can significantly improve the ability to identify the best alternative. Correlation in the prior belief allow us to learn about an alternative's value from samples of similar alternatives. Correlation in sampling, achieved through common random numbers, allow us to reduce the variance in comparing one alternative to another. These two correlations are brought together for the first time in a highly-sequential knowledge-gradient sampling algorithm,

which chooses points to sample using a Bayesian value of information (VOI) criterion. We provide almost sure convergence guarantees as the number of samples grows without bound when parameters are known, provide approximations that allow practical implementation, and demonstrate that CRN leads to improved optimization performance for VOI-based algorithms in sequential sampling environments with a combinatorial number of alternatives and costly samples.

A version of Chapter 3 appeared as Frazier et al. (2011). A journal version of Chapter 3 is in review (Xie et al. 2013).

Chapter 4

In many applications of simulation-based optimization, the random output variable whose expectation is being optimized is a deterministic function of a low-dimensional random vector. This deterministic function is often expensive to compute, making simulation-based optimization difficult. Motivated by an application in the design of bypass grafts for cardiovascular surgery with uncertainty about input parameters, Chapter 4 uses Bayesian methods to design an algorithm that exploits this random vector's low-dimensionality to improve performance.

A version of Chapter 4 appeared as Xie et al. (2012).

Chapter 5

Chapter 5 considers the Bayesian formulation of the ranking and selection problem, with an independent normal prior, independent samples, and a cost per sample. While a

number of procedures have been developed for this problem in the literature, the gap between the best existing procedure and the Bayes-optimal one remains unknown, because computation of the Bayes-optimal procedure using existing methods requires solving a stochastic dynamic program whose dimension increases with the number of alternatives. In this chapter, we give a tractable method for computing an upper bound on the value of the Bayes-optimal procedure, which uses a decomposition technique to break a high-dimensional dynamic program into a number of low-dimensional ones, avoiding the curse of dimensionality. This allows calculation of the optimality gap for any given problem setting, giving information about how much additional benefit we may obtain through further algorithmic development. We apply this technique to several problem settings, finding some in which the gap is small, and others in which it is large.

A version of Chapter 5 appeared as Xie and Frazier (2013b).

Chapter 6

Chapter 6 concludes the contributions of this thesis.

CHAPTER 2

SEQUENTIAL BAYES-OPTIMAL POLICIES FOR MULTIPLE COMPARISONS WITH A KNOWN STANDARD

We consider multiple comparisons with a (known) standard (MCS), in which simulation is used to determine which alternative systems under consideration have performance surpassing that of a standard with known value. MCS problems arise in many applications. We consider the following two examples in detail:

- Administrators of a city's emergency medical services would like to know which of several methods under consideration for positioning ambulances satisfy mandated minimums for percentage of emergency calls answered on time.
- A manufacturing firm is unsure about market and operations conditions in the coming month. Executives would like to know under which conditions their production line can maintain a positive net expected revenue. They plan to use a simulator of their operations to answer this question.

We focus on how simulation effort should be allocated among the alternative systems to best support comparison of alternatives when sampling stops. We first formulate the problem in a Bayesian framework. Using methods from multi-armed bandits and optimal stopping (see, e.g., Gittins and Jones (1974) and DeGroot (1970) respectively), we explicitly characterize and then efficiently compute Bayes-optimal sequential sampling policies for MCS problems. Such policies provide optimal average case performance, where the average is taken under a prior distribution that we choose.

The MCS problem is frequently studied as a special case of a larger class of problems called multiple comparisons with a control (MCC). In MCC problems, the standard (also called the "control") against which we compare is itself the mean performance

of a stochastic system. When this mean is modeled as known, we recover the MCS problem. This can be appropriate when the standard system has been in use for a long period of time, providing sufficient data to estimate its mean performance with high accuracy (Nelson and Goldsman 2001). MCS problems also arise when the standard is some known performance requirement, rather than the mean of a stochastic system.

Several books and survey papers review the previous literature on MCC: Hochberg and Tamhane (1987) and Hsu (1996) are general references on multiple comparisons; Goldsman and Nelson (1994) focuses on multiple comparisons in simulation; and Fu (1994) reviews multiple comparisons as they relate to simulation optimization. Within this MCC literature, work on designing sampling procedures focuses on creating simultaneous confidence intervals for the differences of the mean of each system with that of an unknown control: Paulson (1952), Dunnett (1955) create one-stage procedures assuming independent normal sampling with common known variance; and Dudewicz and Ramberg (1972), Dudewicz and Dalal (1983), Bofinger and Lewis (1992), Damerdji and Nakayama (1996) create two-stage procedures allowing more general sampling distributions. This previous work focuses on difficulties introduced by an unknown control, while we consider a known standard. Also, these procedures have only one or two stages, while we focus on fully sequential procedures, whose ability to adapt the sampling scheme to previous samples offers better sampling efficiency.

Procedures for variants of the MCS problem have also been developed in the indifference-zone ranking and selection (R&S) literature (see Section 1.2). They consider problems in a frequentist context, where the goal is to create a procedure with a worst-case statistical guarantee on solution quality. In contrast, we work within a Bayesian context, where the goal is to provide good performance in the average case. MCS (also called feasibility determination) is also studied in a large deviations frame-

work (see Section 1.2). In other related work, Picheny et al. (2010) proposes adaptive experimental designs with kriging metamodels for approximating a continuous function accurately around a particular level-set. The current work differs from all previous work by finding optimal fully sequential procedures in a Bayesian formulation of the MCS problem that explicitly models a limited ability to sample.

The ability shown in this chapter to explicitly and efficiently compute the optimal sequential policy contrasts the MCS problem with Bayesian R&S (see Section 1.1), and other problems in Bayesian experimental design and Bayesian optimal learning, including global optimization (Mockus 1989), dynamic pricing (Araman and Caldenty 2009), inventory control (Ding et al. 2002), and sensor networks (Krause et al. 2008), where finding the optimal policy is usually considered intractable. In such problems, a common suboptimal approach is to compute a myopic one-step lookahead policy (Gupta and Miescke 1996, Chick et al. 2010, Jones et al. 1998, Lizotte et al. 2007). Policies of this type are also called knowledge-gradient (KG) policies (Frazier 2009). In our numerical experiments, we derive the KG policy and compare it against the optimal policy. We find that in some cases the KG policy performs extremely well (see also Frazier et al. (2008, 2009)), while in other cases it performs poorly. This variability in performance is similar to results in Frazier and Powell (2010), Ryzhov et al. (2012).

Our framework and the resulting ability to compute the Bayes-optimal policies is general. First, it allows two different methods for modeling the limited ability to sample: an explicit cost for each sample (appropriate, e.g., when using on-demand cloud-computing services, in which fees are proportional to the number of CPU hours consumed), and/or a random ceiling on the number of samples allowed. Second, it allows a broad class of terminal payoff functions for modeling the consequences of correct and incorrect comparisons. Third, it provides the ability to model sampling distribu-

tions within any exponential family, which includes common families of sampling distributions like normal with known sampling variance, normal with unknown sampling variance, Bernoulli, and Poisson. We present in detail two such cases, normal with known variance and Bernoulli, and derive additional theoretical results for these special settings.

While our results allow computing the Bayes-optimal policies in a broad class of problem settings, these results have two important limitations. First, they cannot incorporate information about a *known* sampling budget. In this situation, we recommend a heuristic policy based on the optimal policy for a stochastic budget. Second, they require that the value of the standard against which systems are compared is known.

We formulate the problem in Section 2.1. In Section 2.2 we present Bayes-optimal policies for general sampling distributions, considering separately the case of an almost surely finite horizon with or without sampling costs (Section 2.2.1), and the case of an infinite horizon with sampling costs (Section 2.2.2). Then, in Sections 2.3 and 2.4, we specialize to two types of sampling: Bernoulli samples, and normal samples with known variance. We give theoretical results particular to these more specialized cases, and provide techniques for computing the optimal policies efficiently. In Section 2.5 we demonstrate the resulting Bayes-optimal algorithms on illustrative problems, and on two examples in emergency services and manufacturing.

2.1 Problem Formulation

In this section we formulate the general Bayesian MCS problem, which allows both a random ceiling on the number of samples, and sampling costs. We have k alternative systems that we can simulate, and samples from each alternative are independent and

from distributions that do not change over time. For each $x = 1, 2 \ldots, k$, let $f(\cdot|\eta_x)$ be the probability density function (pdf) or probability mass function (pmf) for samples from alternative x, where η_x is an unknown parameter or vector of parameters residing in a parameter space Ξ. We further assume that the space of possible sampling distributions $\{f(\cdot|\eta) : \eta \in \Xi\}$ form an exponential family. See DeGroot (1970) Chapter 9 for an in-depth treatment of exponential families. This assumption of an exponential family allows most common parametric sampling distributions, including the normal (with known variance) and Bernoulli distributions considered in detail in Sections 2.3 and 2.4, as well as normal with unknown variance, Poisson, multinomial, and many others.

We wish to find the set of alternatives whose underlying performance is above a corresponding threshold or control. The underlying performance of each alternative x is characterized by the mean of its sampling distribution, θ_x, which is a known function of η_x. The corresponding threshold is d_x. Hence we want to determine the set $\mathbb{B} = \{x : \theta_x \geq d_x\}$.

We take a Bayesian approach, placing a prior probability distribution on each unknown η_x. This prior distribution represents our subjective beliefs about this sampling distribution. To facilitate computation, we adopt independent conjugate priors. Specifically, we suppose the independent prior distributions on $\eta_1, \ldots, \eta_k$ come from a common conjugate exponential family $\mathscr{D}$ with parameter space Λ. For example, in Section 2.3 where samples are Bernoulli-distributed, the prior is beta-distributed and Λ is the space of parameters of the beta distribution. Let the corresponding parameters of these prior distributions be $S_{0,1}, \ldots, S_{0,k}$, each of which resides in Λ. Denote by $\mathbf{S}_0$ the vector composed of $S_{0,x}$ with x ranging from 1 to k.

Time is indexed by $n = 1, 2, \ldots$. At each time n we choose an alternative $x_n \in \{1, \ldots, k\}$ from which to sample, and observe a corresponding sample y_n which has pdf

or pmf $f(\cdot \mid \eta_{x_n})$. We refer to the decision x_n as our "sampling decision", and our focus in this chapter is on how to best make these sampling decisions, and a related stopping decision discussed below.

As our prior is conjugate to our sampling distribution, our samples result in a sequence of posterior distributions on $\eta_1, \ldots, \eta_k$, each of which resides in the same conjugate family parameterized by Λ. We denote the parameters of these posteriors at time $n \geq 1$ by $S_{n,1}, \ldots, S_{n,k}$, and the vector composed of them by $\mathbf{S}_n$. Then for $n \geq 1$, $\mathbf{S}_n = G(\mathbf{S}_{n-1}, x_n, y_n)$, where $G(\cdot,\cdot,\cdot)$ is some known and fixed function determined by the exponential and conjugate families. Moreover, for all x, the posterior remains independent across x under this update. Define $\mathbb{S} = \Lambda^k$, which is the state space of the stochastic process $(\mathbf{S}_n)_{n\geq 0}$. We will sometimes refer to a generic element of $\mathbb{S}$ as $\mathbf{s} = (s_1, \ldots, s_k)$, and a generic element of Λ as s. In this chapter we use boldfaced parameters to refer to multiple alternatives and regular font to refer to a single alternative.

We allow decisions to depend only upon the data available from previous samples. To make this requirement more formal, we define a filtration $(\mathscr{F}_n)_{n\geq 0}$, where $\mathscr{F}_n$ is the sigma-algebra generated by $x_1, y_1, \ldots, x_n, y_n$. We require that $x_{n+1} \in \mathscr{F}_n$ for $n \geq 0$. In addition to the sampling decisions $(x_n)_{n\geq 1}$, we also choose the total number of samples we take, denoted by τ. We require τ to be a stopping time of the filtration, i.e., we require the event $\{\tau = n\}$ to be $\mathscr{F}_n$–measurable for all $n \geq 0$.

We refer to a collection of rules for making all of the required decisions in a decision-making problem as a policy. Thus, in this problem a policy π is composed of a sampling rule for choosing the sequence of sampling decisions $(x_n)_{n\geq 1}$, and a stopping rule for choosing τ.

For each $n \geq 0$, let $\mathbb{E}_n$ denote the conditional expectation with respect to the infor-

mation available after n samples, so $\mathbb{E}_n[\cdot] = \mathbb{E}[\cdot \mid \mathscr{F}_n]$. When the expectation depends on the policy π, we write $\mathbb{E}^\pi$ for the unconditional expectation, and $\mathbb{E}_n^\pi$ for the conditional expectation with respect to $\mathscr{F}_n$.

In the general formulation of the MCS problem that we consider here, we model the need to sample efficiently in two complementary ways. First, we suppose that each sample incurs a nonnegative cost. For $x = 1, \ldots, k$, denote by $c_x \geq 0$ the sampling cost for alternative x. Second, we suppose that there is some random time horizon T beyond which we will be unable to sample, so that we stop sampling at time $\tau \wedge T$, where $\wedge$ is the minimum operator.

Most frequently this horizon T is imposed because the results of the simulation are needed by the simulation analyst. For analytical convenience, we assume that T is geometrically distributed, and independent of the sampling process. Let $1-\alpha$ be the parameter of this geometric distribution, with $0 < \alpha < 1$, so that $\mathbb{E}[T] = 1/(1-\alpha)$. We also allow T to be a random variable that is infinite with probability 1, in which case we take $\alpha = 1$. In either case, we can equivalently model this random time horizon by supposing that external circumstances may require us to stop after each sample independently with probability $1-\alpha$. While our model does not allow a deterministic horizon T, one can apply the Bayes-optimal procedures we develop in such situations as heuristics by choosing α so that $T = 1/(1-\alpha)$. We use this method in Section 2.5.1 and 2.5.2.

We assume that sampling is penalized, either through a finite horizon ($\alpha < 1$), or a cost per sample ($c_x > 0$ for all x), or both. That is, we disallow the combination of $\alpha = 1$ and $c_x = 0$, which prevents the unrealistic situation of sampling from x forever at no cost.

Define a terminal payoff function r with the following decomposition:

$$r(B;\theta,d) = \sum_{x\notin B} r_0(x;\theta_x) + \sum_{x\in B} r_1(x;\theta_x),$$

where r_0 and r_1 are known real-valued functions. This decomposition is necessary for our analysis, and without it many of our results would not hold.

Adapted to the information filtration $(\mathscr{F}_n)_{n\geq 0}$, we choose a sequence of sets $(B_n)_{n\geq 0}$ to approximate the objective set $\mathbb{B}$. We require that each $B_n \subseteq \{1,2,\ldots,k\}$ is chosen to maximize the expected terminal payoff given the available data after n samples. Formally, for all $n \geq 0$,

$$\begin{aligned} B_n &= \operatorname*{argmax}_{B\subseteq\{1,2,\ldots,k\}, B\in\mathscr{F}_n} \mathbb{E}_n\left[r(B;\theta,d)\right] \\ &= \operatorname*{argmax}_{B\subseteq\{1,2,\ldots,k\}, B\in\mathscr{F}_n} \left\{ \sum_{x\notin B} \mathbb{E}_n\left[r_0(x;\theta_x)\right] + \sum_{x\in B} \mathbb{E}_n\left[r_1(x;\theta_x)\right] \right\}. \end{aligned}$$

Denote by $h_{ix}(s)$ $(i=0,1)$ the expectation of $r_i(x;\theta_x)$ when η_x follows a distribution in family $\mathscr{D}$ with parameter s, i.e., $\eta_x \sim \mathscr{D}(s)$, and define $h_x(s) = \max\{h_{0x}(s), h_{1x}(s)\}$ for $s \in \Lambda$, $x = 1,\ldots,k$. Simple algebra then yields

$$B_n = \{x : h_{0x}(S_{n,x}) \leq h_{1x}(S_{n,x})\} \quad \text{and} \quad \mathbb{E}_n\left[r(B_n;\theta,d)\right] = \sum_{x=1}^{k} h_x(S_{n,x}). \tag{2.1}$$

Our estimate of the set $\mathbb{B}$ is $B_{\tau\wedge T}$ when sampling stops. Our goal is to find a policy that maximizes the expected total reward, i.e., to solve the problem

$$\sup_{\pi} \mathbb{E}^{\pi}\left[r(B_{\tau\wedge T};\theta,d) - \sum_{n=1}^{\tau\wedge T} c_{x_n} \right]. \tag{2.2}$$

2.1.1 Terminal Payoff Functions: Conditions and Examples

While our results apply to general terminal payoff functions, some of our theoretical results require additional conditions defined below. Payoff Condition 1 states that, for

each x, $\{h_x(S_{n,x})\}_{n\geq 0}$ is a sub-martingale, i.e., that the expected terminal payoff (not including the cost of sampling) improves as we collect more sample information. Payoff Condition 2 states that this improvement has an upper bound that may depend upon the starting posterior, but not on the number of samples taken. These conditions together provide necessary bounds for the reward functions, the Gittins indices, and the value functions introduced later in Section 2.2. Payoff Condition 3 states, roughly speaking, that the improvement in expected terminal payoff from an arbitrarily large amount of additional sampling vanishes as we sample more and more. Payoff Condition 3 provides additional results that further facilitate computation of the optimal policy. When a theoretical result requires one or more of these conditions, we state this explicitly in the result.

Condition 1. *For any $x = 1,\ldots,k$ and $s \in \Lambda$, $h_x(s) \leq \mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = s, x_1 = x\right]$.*

Condition 2. *There exist deterministic non-negative functions $H_1,\ldots,H_k$ on Λ such that, for any x, $n \geq 0$ and $s \in \Lambda$, $\mathbb{E}\left[h_x(S_{n,x}) \mid S_{0,x} = s, x_1 = \cdots = x_n = x\right] - h_x(s) \leq H_x(s)$.*

Condition 3. *There exist deterministic non-negative functions $\widetilde{H}_1,\ldots,\widetilde{H}_k$ on Λ such that, for any x and $s \in \Lambda$,*

$$\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = s, x_1 = x\right] - h_x(s) \leq \widetilde{H}_x(s), \tag{2.3}$$

$$\lim_{n\to\infty}\left[\sup_{s\in PS(x;n)} \widetilde{H}_x(s)\right] = 0, \tag{2.4}$$

where $PS(x;n) := \left\{s \in \Lambda\colon \exists s' \in \Lambda \; s.t. \; \mathbb{P}\left[S_{n,x} = s \mid S_{0,x} = s', x_1 = \cdots = x_n = x\right] > 0\right\}$.

We will consider the following two terminal payoff functions in detail throughout the chapter. Let m_{ix} for $i = 0, 1$ and $x = 1,\ldots,k$ be non-negative constants.

Example 1: (**0-1 Terminal Payoff**) $r_0(x;\theta_x) = m_{0x}\cdot \mathbf{1}_{\{x\notin\mathbb{B}\}}$, $r_1(x;\theta_x) = m_{1x}\cdot \mathbf{1}_{\{x\in\mathbb{B}\}}$. When $m_{0x} = m_{1x} = 1$ for all x, $r(B;\theta,d)$ equals the total number of alternatives correctly classified by B.

Example 2: (**Linear Terminal Payoff**) $r_0(x;\theta_x) = m_{0x}(d_x - \theta_x)$, $r_1(x;\theta_x) = m_{1x}(\theta_x - d_x)$.

We further characterize these example terminal payoff functions in Table 2.1, giving explicit expressions for $h_{ix}(s)$, B_n, and $H_x(s)$ under these payoff functions, and showing that both payoff functions satisfy Payoff Conditions 1 and 2. The table uses the following additional notation:

$$
\begin{aligned}
&p_x(s) := \mathbb{P}\{\theta_x \geq d_x \mid \eta_x \sim \mathscr{D}(s)\}, \quad \mu(s) := \mathbb{E}[\theta_x \mid \eta_x \sim \mathscr{D}(s)], \\
&\mu_{nx} := \mathbb{E}_n[\theta_x] = \mu(S_{n,x}), \quad B(x) := \left\{ s\colon\ p_x(s) \geq \frac{m_{0x}}{m_{0x}+m_{1x}} \right\}, \\
&A_0(s) := \mathbb{E}\left[m_{0x}(\theta_x - d_x)^- + m_{1x}(\theta_x - d_x)^+ \,\middle|\, \eta_x \sim \mathscr{D}(s) \right],
\end{aligned}
$$

where $(z)^+ = \max(z,0)$ and $(z)^- = \max(-z,0)$ denote the positive part and the negative part respectively. The proof of the statements in this table can be found in Appendix A.

Table 2.1: Example Terminal Payoff Functions and Their Properties

	0-1 Terminal Payoff	Linear Terminal Payoff
$h_{0x}(s)$	$m_{0x}[1 - p_x(s)]$	$m_{0x}[d_x - \mu(s)]$
$h_{1x}(s)$	$m_{1x} \cdot p_x(s)$	$m_{1x}[\mu(s) - d_x]$
B_n	$\left\{x : p_x(S_{n,x}) \geq \frac{m_{0x}}{m_{0x}+m_{1x}}\right\}$	$\{x : \mu_{nx} \geq d_x\}$
Payoff Condition 1	Yes	Yes
Payoff Condition 2	Yes	Yes
$H_x(s)$	$m_{0x} \cdot \mathbf{1}_{\{s \notin B(x)\}} + m_{1x} \cdot \mathbf{1}_{\{s \in B(x)\}} - h_x(s)$	$A_0(s) - h_x(s)$

2.2 The Optimal Solution

In this section we present the optimal solution to the Bayesian MCS problem (2.2), which allows both a geometrically distributed sampling horizon, and sampling costs. We first present some preliminary results, and then give solutions for a geometrically distributed horizon in Section 2.2.1, and for an infinite horizon with sampling costs in Section 2.2.2. The results in this section apply to the general sampling framework given in Section 2.1, and in later sections we specialize to sampling with Bernoulli observations (Section 2.3) and normal observations (Section 2.4).

We solve the problem (2.2) using dynamic programming (DP) (Bellman (1954), and see references Dynkin and Yushkevich (1979), Bertsekas (2005, 2007), Powell (2007)). In the DP approach, we define a value function $V : \mathbb{S} \mapsto \mathbb{R}$. For each state $\mathbf{s} \in \mathbb{S}$, $V(\mathbf{s})$ is the optimal expected total reward attainable when the initial state is $\mathbf{s}$. That is,

$$V(\mathbf{s}) = \sup_{\pi} \mathbb{E}^{\pi} \left[r(B_{\tau \wedge T}; \theta, d) - \sum_{n=1}^{\tau \wedge T} c_{x_n} \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right]. \tag{2.5}$$

An optimal policy is any policy π attaining this supremum.

Before describing these optimal policies in Sections 2.2.1 and 2.2.2, we transform the value function to a form that supports later theoretical development. Let $R_0(\mathbf{s}) := \sum_{x=1}^{k} h_x(s_x)$. Since R_0 is a deterministic function of the initial state, we can subtract it from the value function, and redefine V as the optimal expected incremental reward over R_0. We then have the following proposition.

Proposition 1.

$$V(\mathbf{s}) = \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{n=1}^{\tau} \alpha^{n-1} \mathscr{R}_{x_n}(S_{n-1,x_n}) \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right], \tag{2.6}$$

where the reward functions $\mathscr{R}_x : \Lambda \mapsto \mathbb{R}$ for $x = 1, \ldots, k$ are defined by

$$\mathscr{R}_x(s) = \mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = s, x_1 = x \right] - h_x(s) - c_x. \tag{2.7}$$

It is clear that $\mathscr{R}_x$ is bounded below by $-c_x$ under Payoff Payoff Condition 1. It is bounded above by $-c_x + H_x$ under Payoff Condition 2, and by $-c_x + \widetilde{H}_x$ under Payoff Condition 3.

In the following subsections we divide our assumption of an almost surely finite horizon ($\alpha < 1$) or a cost per sample ($c_x > 0$ for all x) into two distinct cases, and solve the MCS problem in each case using a distinct technique. The first (Section 2.2.1) assumes $\alpha < 1$ (geometric horizon), and the second (Section 2.2.2) assumes $\alpha = 1$ and $c_x > 0$ for all x (infinite horizon with sampling costs).

2.2.1 Geometric Horizon

We first consider the MCS problem with T almost surely finite, i.e., with $0 < \alpha < 1$. We make no restrictions on the sampling costs c_x, allowing them to be 0 or strictly positive. In this case, (2.6) is a multi-armed bandit (MAB) problem (see, e.g., Mahajan and Teneketzis (2008)).

To solve a Bayesian MAB problem, Gittins and Jones (1974) showed that it is sufficient to compute Gittins indices $\nu_x(s)$ for each possible state s, which can be written here as

$$\nu_x(s) = \max_{\tau > 0} \mathbb{E}\left[\frac{\sum_{n=1}^{\tau} \alpha^{n-1} \mathscr{R}_x(S_{n-1,x})}{\sum_{n=1}^{\tau} \alpha^{n-1}} \,\middle|\, S_{0,x} = s, x_1 = \cdots = x_\tau = x \right]. \tag{2.8}$$

The optimal sampling rule, whose decisions we denote $(x_n^*)_{n \geq 1}$, is then to select at each time the alternative with a corresponding state that has the largest Gittins index. The optimal stopping time, which we write as τ^*, is the first time when all the k indices are non-positive. Formally,

$$x_{n+1}^* = \operatorname*{argmax}_x \{\nu_x(S_{n,x})\}, \ \forall n \geq 0; \qquad \tau^* = \inf\{n : \nu_x(S_{n,x}) \leq 0, \forall x\}.$$

When $c_x = 0$ for all x and Payoff Condition 1 holds, the Gittins indices are always nonnegative by Proposition 2. We may then choose τ^* to be $+\infty$.

Computation of (2.8) is much easier than solving the full DP because the dimension of $s \in \Lambda$ is smaller than that of $\mathbf{s} \in \mathbb{S} = \Lambda^k$, and the computational complexity of solving a DP scales poorly with the dimension of the state space, due to the so-called curse of dimensionality (Powell 2007).

We introduce the following bounds on the Gittins indices to serve approximate computation of the optimal policy when the state space is infinite (see Section 2.4.1).

Proposition 2. *Under Payoff Condition 1, $\nu_x \geq -c_x$. Under Payoff Conditions 1 and 2, $-c_x \leq \nu_x \leq -c_x + H_x$.*

2.2.2 Infinite Horizon With Sampling Costs

We now consider the MCS problem with $T = \infty$ almost surely and positive sampling costs, i.e., $\alpha = 1$ and $c_x > 0$ for all x. With these values for α and c_x, (2.6) becomes

$$V(\mathbf{s}) = \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{n=1}^{\tau} \mathscr{R}_{x_n}(S_{n-1,x_n}) \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right].$$

Now fix some x and consider a sub-problem in which only alternative x can be sampled. The optimal expected reward for this single-alternative problem with initial state s is then

$$V_x(s) = \sup_{\tau_x} \mathbb{E} \left[\sum_{n=1}^{\tau_x} \mathscr{R}_x(S_{n-1,x}) \,\middle|\, S_{0,x} = s, x_1 = \cdots = x_{\tau_x} = x \right], \tag{2.9}$$

where τ_x is the stopping time. We immediately have the following bounds on V_x.

Proposition 3. *$V_x \geq 0$. Under Payoff Condition 2, $V_x \leq H_x$.*

Standard results from the DP literature (see, e.g., Dynkin and Yushkevich (1979)) show that V_x satisfies Bellman's recursion,

$$\begin{aligned} V_x(s) &= \max\left[0, L_x(s, V_x)\right], \quad \text{where} \\ L_x(s, V_x) &= \mathscr{R}_x(s) + \mathbb{E}[V_x(S_{1,x}) \mid S_{0,x} = s, x_1 = x]. \end{aligned} \tag{2.10}$$

Here, the value function V_x is not necessarily Borel-measurable, but is universally measurable, and so the expectation of V_x is taken in this more general sense.

This problem is a standard optimal stopping problem (for details see Bertsekas (2007) Section 3.4) that can be solved by specifying the set of states $\mathbb{C}_x$ on which we should continue sampling (also called the continuation set), which implicitly specifies the set on which we should stop (the stopping set) as $\Lambda \setminus \mathbb{C}_x$. They are optimally specified as $\mathbb{C}_x = \{s \in \Lambda : V_x(s) > 0\}$ and $\Lambda \setminus \mathbb{C}_x = \{s \in \Lambda : V_x(s) = 0\}$. Then, an optimal solution to (2.9) is the stopping time τ_x^* given by $\tau_x^* = \inf\{n \geq 0 : S_{n,x} \notin \mathbb{C}_x\}$.

We allow τ_x^* to be ∞, in which case the state of alternative x never leaves $\mathbb{C}_x$. Even if τ_x^* is almost surely finite, there may be no deterministic upper bound. For example, in the Bayesian formulation of the sequential hypothesis testing problem (Wald and Wolfowitz 1948), there is no fixed almost sure upper bound on the number of samples taken by the Bayes-optimal policy even though sampling has a fixed cost, and considerable research effort has gone to creating good sub-optimal policies that stop within a fixed amount of time (Siegmund 1985). In our problem, however, the set of possible states for x after n samples, i.e., $PS(x;n)$, shrinks under Payoff Condition 3 so that it is contained by $\Lambda \setminus \mathbb{C}_x$ when n exceeds a deterministic value. This gives a deterministic upper bound on τ_x^*, as demonstrated by the following proposition.

Proposition 4. *Under Payoff Condition 3, τ_x^* has a deterministic upper bound N_x, where*

$$N_x := \min\left\{n\colon \left[\sup_{s \in PS(x;n')} \widetilde{H}_x(s)\right] \leq c_x,\ \forall n' \geq n\right\}. \tag{2.11}$$

This bound is also computationally useful because it allows us to restrict the state space when solving Bellman's recursion for the optimal policy (see Sections 2.3.2 and 2.4.2).

Given the independence among the alternatives, our original problem can be decomposed into k sub-problems. This decomposition is used in the proof of the following theorem, Theorem 1, which relates the value functions of these sub-problems to the original problem and gives the optimal policy for the original problem.

Theorem 1. *The value function is given by,* $V(\mathbf{s}) = \sum_{x=1}^{k} V_x(s_x)$. *Furthermore, any policy with sampling decisions* $(x_n^*)_{n\geq 1}$ *and stopping time* τ^* *satisfying the following conditions is optimal:*

$$x_{n+1}^* \in \{x : S_{n,x} \in \mathbb{C}_x\},\ \forall n \geq 0; \qquad \tau^* = \inf\{n \geq 0 : S_{n,x} \notin \mathbb{C}_x, \forall x\}.$$

The following proposition shows that if each τ_x^* is bounded above, then the optimal stopping time for the whole problem is also bounded above.

Proposition 5. *Suppose that each* τ_x^* *has a deterministic upper bound* N_x. *Then the optimal stopping rule* τ^*, *as characterized in Theorem 1, has a deterministic upper bound* $\sum_{x=1}^{k} N_x$.

2.3 Specialization to Bernoulli Sampling

In this section we specialize the results of Section 2.2 to the specific case of Bernoulli samples. We give explicit expressions for quantities described generally in Section 2.2, and then present additional theoretical results and computational methods. Later, in Section 2.4, we pursue the same agenda for another commonly considered type of sampling: normal samples with known sampling variance.

We first give explicit expressions for the statistical model, the reward function, and Bellman's recursion. Here for each x, the underlying performance parameter $\theta_x \in (0,1)$ is the only component of η_x, and the corresponding threshold is $d_x \in (0,1)$. At each time $n \geq 1$, $y_n \mid \theta, x_n \sim \text{Bernoulli}(\theta_{x_n})$.

We adopt a conjugate $\text{Beta}(a_{0x}, b_{0x})$ prior for each θ_x with $a_{0x}, b_{0x} \geq 1$, under which θ_x is independent of $\theta_{x'}$ for $x \neq x'$. Our Bernoulli samples then result in a sequence of posterior distributions on θ_x which are again independently beta-distributed with parameters $S_{n,x} = (a_{nx}, b_{nx})$ in parameter space $\Lambda = [1,+\infty) \times [1,+\infty)$. If we sample x at time n, then

$$(a_{nx}, b_{nx}) = \begin{cases} (a_{n-1,x}+1, b_{n-1,x}), & \text{if } y_n = 1 \\ (a_{n-1,x}, b_{n-1,x}+1), & \text{if } y_n = 0 \end{cases}.$$

We take $\mathbf{S}_n = (\mathbf{a}_n, \mathbf{b}_n)$ as the state of the DP, where the state space is $\mathbb{S} = [1,+\infty)^k \times [1,+\infty)^k$. The state update function G is given by, for $n \geq 1$,

$$(\mathbf{a}_n, \mathbf{b}_n) = \mathbf{1}_{\{y_n=1\}} \cdot (\mathbf{a}_{n-1} + \mathbf{e}_{x_n}, \mathbf{b}_{n-1}) + \mathbf{1}_{\{y_n=0\}} \cdot (\mathbf{a}_{n-1}, \mathbf{b}_{n-1} + \mathbf{e}_{x_n}),$$

where $\mathbf{e}_{x_n}$ denotes a length-k vector with 1 at element x_n and 0 elsewhere.

Now for any $(a,b) \in \Lambda$, $\mathscr{D}(a,b) = \text{Beta}(a,b)$, $\mu(a,b) = a/(a+b)$ and $p_x(a,b) = 1 - I_{d_x}(a,b)$, where the regularized incomplete beta function $I_\cdot(\cdot,\cdot)$ is defined for $a,b > 0$ and $0 \leq d \leq 1$ by

$$I_d(a,b) = \frac{B(d;a,b)}{B(a,b)} = \frac{\int_0^d t^{a-1}(1-t)^{b-1}dt}{\int_0^1 t^{a-1}(1-t)^{b-1}dt}.$$

By Remark 1 in Appendix A and definition (2.7), we also have

$$\mathscr{R}_x(a,b) = -c_x - h_x(a,b) + \frac{a}{a+b} \cdot h_x(a+1,b) + \frac{b}{a+b} \cdot h_x(a,b+1). \tag{2.12}$$

When $\alpha = 1$ and $c_x > 0$ for all x, in each sub-problem with alternative x,

$$\mathbb{E}[V_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x] = \frac{a}{a+b} V(a+1,b) + \frac{b}{a+b} V(a,b+1),$$

by Remark 1 in Appendix A. Thus (2.10) becomes $V_x(a,b) = \max[0, L_x(a,b,V_x)]$, where

$$\begin{aligned} L_x(a,b,V_x) = -c_x - h_x(a,b) + \frac{a}{a+b}\left[h_x(a+1,b) + V_x(a+1,b)\right] \\ + \frac{b}{a+b}\left[h_x(a,b+1) + V_x(a,b+1)\right]. \end{aligned} \tag{2.13}$$

2.3.1 Geometric Horizon

To compute the optimal policy for a geometric horizon in Section 2.2.1 with respect to Bernoulli sampling, we use a technique from Varaiya et al. (1985) for off-line computation of Gittins indices, which assumes a finite state space.

Since the state space is infinite in our problem, we apply this technique in an approximate sense. For each alternative x with initial state (a_{0x}, b_{0x}), we first truncate the horizon for its sub-problem to N_0 (we use $N_0 = 50$ in our experiments). We then apply Varaiya et al. (1985) to pre-compute the Gittins indices for a finite set of states: $\{(a_{0x}+n_a, b_{0x}+n_b) : n_a, n_b \in \mathbb{N}, n_a+n_b \leq N_0, x = 1\ldots,k\}$. When an alternative is sampled more than N_0 times, we take the current state as the new (a_{0x}, b_{0x}), and recompute the indices for a new set of states.

2.3.2 Infinite Horizon with Sampling Costs

For the infinite horizon case with sampling costs, we explicitly compute the optimal policy for Bernoulli sampling, which is characterized for general sampling distributions in Section 2.2.2.

By Theorem 1, it suffices to evaluate for each alternative x all possible states in the sampling process. If Payoff Condition 3 holds, then $V_x(a_{0x}+n_a, b_{0x}+n_b) = 0$ for $n_a+n_b \geq N_x$ by Propositions 3 and 4; and for $n_a+n_b = n < N_x$, $V_x(a_{0x}+n_a, b_{0x}+n_b)$

can be computed recursively from $n = N_x - 1$ to $n = 0$ using (2.13). If Payoff Condition 3 does not hold, or some N_x are large, a pre-specified N_0 can be used instead of N_x to reduce computation. That is, we approximate $V_x(a_{0x}+n_a, b_{0x}+n_b)$ by 0 for $n_a+n_b \geq N_0$ (we use $N_0 = 1000$ in our experiments).

2.3.3 Example Terminal Payoff Functions

We state in Table 2.2 that 0-1 and linear terminal payoff functions satisfy Payoff Condition 3 with Bernoulli sampling, and we give explicit expressions for N_x and $\widetilde{H}_x(a,b)$. The proof of the statements in the table can be found in Appendix A.

Table 2.2: Example Terminal Payoff Functions with Bernoulli Sampling

	0-1 Terminal Payoff	Linear Terminal Payoff
Payoff Condition 3	Yes	Yes
$\widetilde{H}_x(a,b)$	$\frac{m_{0x}+m_{1x}}{2\sqrt{2\pi(a+b)}}$	$\frac{\max\{m_{0x},m_{1x}\}+m_{0x}}{4(a+b+1)}$
N_x	$\left(\lceil \frac{(m_{0x}+m_{1x})^2}{8\pi c_x^2} \rceil - 2\right)^+$	$\left(\lceil \frac{\max\{m_{0x},m_{1x}\}+m_{0x}}{4c_x} \rceil - 3\right)^+$

2.4 Specialization to Normal Sampling

We now consider normally distributed samples with known variance. As done in Section 2.3 for Bernoulli samples, we give explicit expressions for the quantities described generally in Section 2.2, and then present additional theoretical results to compute the optimal policy.

Here the sampling precision is known for each alternative x and denoted by β_x^{ε}.

Hence η_x only consists of θ_x, and $d_x \in (-\infty, +\infty)$ for all x. We have $y_n \mid \theta, x_n \sim \mathcal{N}(\theta_{x_n}, 1/\beta_{x_n}^{\varepsilon})$ for all $n \geq 1$. We adopt an independent conjugate $\mathcal{N}(\mu_{0x}, 1/\beta_{0x})$ prior for each θ_x, and our normal samples result in a sequence of normal posterior distributions on θ_x with parameters $S_{n,x} = (\mu_{nx}, \beta_{nx})$ in parameter space $\Lambda = (-\infty, +\infty) \times [0, +\infty)$. We take $\mathbf{S}_n = (\boldsymbol{\mu}_n, \boldsymbol{\beta}_n)$ as the state of the DP, where the state space is $\mathbb{S} = (-\infty, +\infty)^k \times [0, +\infty)^k$.

Using Bayes rule, we write the state update function G as follows. For all $n \geq 0$,

$$\mu_{n+1,x} = \begin{cases} [\beta_{nx}\mu_{nx} + \beta_x^{\varepsilon} y_{n+1}]/\beta_{n+1,x} & \text{if } x = x_{n+1} \\ \mu_{nx} & \text{otherwise} \end{cases},$$

$$\beta_{n+1,x} = \begin{cases} \beta_{nx} + \beta_x^{\varepsilon} & \text{if } x = x_{n+1} \\ \beta_{nx} & \text{otherwise} \end{cases}.$$

Frazier et al. (2008) gives a probabilistically equivalent form of this update in terms of an $\mathcal{F}_n$-adapted sequence of standard normal random variables $Z_1, Z_2, \ldots$. More specifically, for all $n \geq 1$,

$$(\boldsymbol{\mu}_n, \boldsymbol{\beta}_n) = \left(\boldsymbol{\mu}_{n-1} + \tilde{\sigma}_{x_n}(\beta_{n-1,x_n}) Z_n \mathbf{e}_{x_n}, \boldsymbol{\beta}_{n-1} + \beta_{x_n}^{\varepsilon} \mathbf{e}_{x_n}\right), \tag{2.14}$$

where $\tilde{\sigma}_x \colon (0, \infty] \mapsto [0, \infty)$ for each x is defined by

$$\tilde{\sigma}_x(\gamma) = \sqrt{(\gamma)^{-1} - (\gamma + \beta_x^{\varepsilon})^{-1}} = \sqrt{\beta_x^{\varepsilon}/[\gamma(\gamma + \beta_x^{\varepsilon})]}.$$

It follows that for any $(\mu, \beta) \in \Lambda$, $\mathcal{D}(\mu, \beta) = \mathcal{N}(\mu, 1/\beta)$ and

$$\begin{aligned} p_x(\mu, \beta) &= 1 - \Phi\left(\sqrt{\beta}(d_x - \mu)\right), \\ \mathcal{R}_x(\mu, \beta) &= -c_x + \mathbb{E}\left[h_x(\mu + \tilde{\sigma}_x(\beta) Z, \beta + \beta_x^{\varepsilon})\right] - h_x(\mu, \beta), \end{aligned} \tag{2.15}$$

where Φ is the standard normal cdf and Z is a standard normal random variable.

When $\alpha = 1$ and $c_x > 0$ for all x, (2.10) becomes

$$\begin{aligned} V_x(\mu,\beta) &= \max\left[0, L_x(\mu,\beta,V_x)\right], \text{ where} \\ L_x(\mu,\beta,V_x) &= \mathscr{R}_x(\mu,\beta) + \mathbb{E}[V_x(\mu+\tilde{\sigma}_x(\beta)Z, \beta+\beta_x^{\varepsilon})]. \end{aligned} \tag{2.16}$$

2.4.1 Geometric Horizon

Similar to the Bernoulli sampling case in Section 2.3.1, we first truncate the horizon for each sub-problem to N_0. With normal sampling, this is not yet enough to provide a finite set of states: although β is discrete, μ takes continuous values. We thus need the following condition on H_x.

Special Condition 1. *For any fixed x and β, $H_x(\mu,\beta) \to 0$ as $\mu \to +\infty$ or $\mu \to -\infty$.*

Under Payoff Conditions 1, 2 and Special Condition 1, we know from (2.7), Propositions 2 and 3 that for any fixed β, $\mathscr{R}_x(\mu,\beta) \to -c_x$, $\nu_x(\mu,\beta) \to -c_x$, and $V_x(\mu,\beta) \to 0$ as $\mu \to +\infty$ or $\mu \to -\infty$. We can then truncate and discretize the range of μ_x as follows.

Let $\varepsilon, \delta > 0$ be small (we take $\varepsilon = \delta = 0.01$ in our experiments). For each fixed $\beta \in \{\beta_{0x} + n\beta_x^{\varepsilon} : 0 \le n \le N_0\}$, we compute an interval $\left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right]$ (guaranteed to exist under Special Condition 1) such that for all $\mu \notin \left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right]$, we have $0 \le H_x(\mu,\beta) \le \varepsilon$, and hence $-c_x \le \nu_x(\mu,\beta) \le -c_x + \varepsilon$ by Proposition 2. We then discretize $\left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right]$ into points with interval δ, denoted by $\left\{\mu_x^i(\beta)\right\}_i$.

We now use the technique from Varaiya et al. (1985) to pre-compute the Gittins indices for a finite set of states, $\left\{\left(\mu_x^i(\beta),\beta\right) : \beta \in \{\beta_{0x} + n\beta_x^{\varepsilon} : 0 \le n \le N_0\}, x = 1,\ldots,k\right\}$, with the transition probability matrix approximated by the density ratios using (2.14):

$$\mathbb{P}\left[\left(\mu_x^i(\beta),\beta\right) \to \left(\mu_x^j(\beta+\beta_x^{\varepsilon}), \beta+\beta_x^{\varepsilon}\right)\right] = \frac{\varphi\left(\frac{\mu_x^j(\beta+\beta_x^{\varepsilon}) - \mu_x^i(\beta)}{\tilde{\sigma}_x(\beta)}\right)}{\sum_k \varphi\left(\frac{\mu_x^k(\beta+\beta_x^{\varepsilon}) - \mu_x^i(\beta)}{\tilde{\sigma}_x(\beta)}\right)},$$

where φ is the standard normal pdf. For an arbitrary state (μ, β) of alternative x, we set

$$\nu_x(\mu,\beta) = \begin{cases} -c_x & \text{if } \mu \notin \left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right], \\ \nu_x\left(\mu_x^i(\beta), \beta\right) & \text{otherwise, where } i = \operatorname{argmin}_j \left\{ |\mu - \mu_x^j(\beta)| \right\}. \end{cases}$$

As we did in the Bernoulli sampling case, we also track the number of samples taken from each alternative and recompute the indices when an alternative is sampled more than N_0 times.

2.4.2 Infinite Horizon with Sampling Costs

Computation of the optimal policy in Section 2.2.2 is not trivial for normal sampling. To implement Bellman's recursion (2.16) directly, we need to evaluate each single-alternative value function over the whole continuous domain of μ, which is not possible. Instead, we truncate and discretize the range of μ to evaluate it approximately. The following condition is required for applying truncation.

Special Condition 2. *For any fixed x and β, we can compute an interval $\left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right]$ such that $\mu \notin \left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right] \Rightarrow V_x(\mu,\beta) = 0$.*

Under Payoff Condition 3 and Special Condition 2, we can evaluate each V_x as follows. By Propositions 3 and 4, $V_x(\mu, \beta_{0x} + n\beta_x^{\varepsilon}) = 0$ for all $n \geq N_x$ and $\mu \in \mathbb{R}$. For each $\beta \in \{\beta_{0x} + n\beta_x^{\varepsilon} : 0 \leq n < N_x\}$, we compute $\left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right]$ (given in Special Condition 2) as a boundary of the value of μ within $\mathbb{C}_x$, and discretize it into points $\left\{\mu_x^i(\beta)\right\}_i$ with an interval of δ between them (we set $\delta = 0.01$ in our experiments). Using Remark 2 in Appendix A and (2.16), we know that each $V_x\left(\mu_x^i(\beta), \beta\right)$ can be computed

recursively for $\beta \in \{\beta_{0x} + n\beta_x^{\varepsilon} : 0 \le n < N_x\}$. For any arbitrary state (μ, β), we set

$$V_x(\mu,\beta) = \begin{cases} 0 & \text{if } \mu \notin \left[\overline{\mu}_x(\beta), \underline{\mu}_x(\beta)\right], \\ V_x\left(\mu_x^i(\beta), \beta\right) & \text{otherwise, where } i = \operatorname{argmin}_j \left\{|\mu - \mu_x^j(\beta)|\right\}. \end{cases}$$

2.4.3 Example Terminal Payoff Functions

Remark 3 in Appendix A describes the explicit computation of $\mathscr{R}_x(\mu,\beta)$ for $(\mu,\beta) \in \Lambda$ under 0-1 and linear terminal payoff. Table 2.3 shows that with normal sampling, these payoff functions satisfy Payoff Condition 3, Special Conditions 1 and 2. It uses the following additional notation:

$$A_0 = \max\{m_{0x}, m_{1x}\}\left[\left(\sqrt{1+\beta_x^{\varepsilon}/\beta} - 1\right)A_3 + \pi^{-1}\sqrt{\beta_x^{\varepsilon}/\beta}\right],$$

$$A_1 = A_4^2 + 2A_4A_5, \qquad A_2 = 2(A_4 + A_5)/\pi, \qquad A_3 = \pi^{-2} - A_4^2,$$

$$\text{where} \qquad A_4 = 1 + 1/\sqrt{2\pi e} \quad \text{and} \quad A_5 = c_x/\max\{m_{0x}, m_{1x}\}.$$

The proof of the statements in this table can be found in Appendix A.

Table 2.3: Example Terminal Payoff Functions with Normal Sampling

	0-1 Terminal Payoff	Linear Terminal Payoff
Payoff Condition 3	Yes	Yes
$\widetilde{H}_x(\mu,\beta)$	A_0	$\frac{m_{0x}+m_{1x}}{\sqrt{2\pi\beta}}$
N_x	$\lceil \frac{A_2+\sqrt{A_2^2-4A_1A_3}}{2A_1} \rceil$	$\lceil \frac{(m_{0x}+m_{1x})^2}{2\pi c_x^2 \beta_x^{\varepsilon}} \rceil$
Special Condition 1	Yes	Yes
Special Condition 2	Yes	Yes

2.5 Numerical Results

In this section, we test the performance of the Bayes-optimal policies with a collection of numerical experiments. We first present illustrative example problems in Section 2.5.1, and then present applications to ambulance positioning in Section 2.5.2, and the revenue of a production line in Section 2.5.3.

We introduce the following sampling policies for comparison. In the implementation of all of these policies, ties in the argmax are broken uniformly at random.

1. **Pure Exploration (PE):** In this policy, we choose the next alternative to sample uniformly and independently at random, i.e., $x_n \sim \text{Uniform}(1,\ldots,k)$ for all $n \geq 1$.
2. **Large Deviations (LD):** In this policy, we sample according to the asymptotically optimal fractional allocations $p_1^*,\ldots,p_k^*$ with the highest exponential rate of decay for the expected number of incorrect determinations. Szechtman and Yücesan (2008) provides explicit characterizations for Bernoulli and normal sampling. Our implementation of LD is idealized, as it uses the exact values of p^*, which require knowing θ. These exact values are unavailable in practice, and so practical algorithms, such as those proposed in Szechtman and Yücesan (2008), use estimates instead.
3. **Andradóttir and Kim (AK):** AK is a feasibility check procedure for normally distributed systems with a single constraint, described in Algorithm $\mathscr{F}$ in Andradóttir and Kim (2010), and Procedure $\mathscr{F}_B^I$ in Batur and Kim (2010). It provides a statistical guarantee on performance with pre-specified tolerance level ε^{AK} and lower bound $1-\alpha^{AK}$ on the probability of correct decision (PCD). It uses an initial stage with $n_0^{AK} \geq 2$ samples from each alternative. Our implementation uses known variance instead of an estimate. Batur and Kim (2010) states that AK is

robust to non-normality, so we apply it to both Bernoulli and normal sampling.

4. **Knowledge Gradient (KG):** In this policy, the sampling decision is the one that would be optimal if only one measurement were to remain, i.e., $x_{n+1} \in \text{argmax}_x\{\mathscr{R}_x(S_{n,x})\}$ for all $n \geq 0$. Such policies have also been called myopic or one-step-lookahead policies, and are discussed in detail in Frazier (2009). When sampling costs are strictly positive, KG stops when the one-step expected reward becomes non-positive. This stopping rule is $\tau = \inf\{n \geq 0 : \mathscr{R}_x(S_{n,x}) \leq 0, \forall x\}$. An analogous stopping rule for ranking and selection was introduced in Frazier and Powell (2008).

2.5.1 Illustrative Example Problems

We first explore the relative performance of these policies and the Bayes-optimal one on six different problem settings: geometric horizon with Bernoulli sampling; geometric horizon with normal sampling; deterministic horizon with Bernoulli sampling; deterministic horizon with normal sampling; infinite horizon with Bernoulli sampling; and infinite horizon with normal sampling. We adopt the 0-1 terminal payoff function with $m_{0x} = m_{1x} = 1$ for all x.

For Bernoulli sampling, we test $k = 100$ alternatives with each d_x picked from Uniform $[0,1]$ and each θ_x randomly generated from prior distribution Beta$(1,1)$. For normal sampling, we test $k = 50$ alternatives with d_x picked from $\mathscr{N}(0,100)$, θ_x generated from prior distribution $\mathscr{N}(0,100)$, and β_x^ε picked from Uniform $[0.5,2]$. For the geometric horizon, we set $c_x = 0$ for all x and vary α so that $\mathbb{E}[T] = \frac{1}{1-\alpha}$ varies within $[100,1000]$ in the Bernoulli sampling case, and within $[50,500]$ in the normal sampling case. The actual values of T are randomly generated according to the corresponding ge-

ometric distribution. For the deterministic horizon, we apply the Bayes-optimal policy as a heuristic, by choosing α so that $\frac{1}{1-\alpha}$ is equal to the horizon. For the infinite horizon, we set $c_x = c$ for all x and vary c within $[0.001, 0.01]$.

Since LD does not have a stopping rule, we only implement it for the geometric horizon and the deterministic horizon. For AK, we set $n_0^{AK} = 2$. If AK has not yet classified an alternative as in or out of the level set by the end of the horizon, we classify it according to the optimal level set estimate B_n. For the finite (geometric or deterministic) horizon, we optimize AK's performance over the values of ε^{AK} and α^{AK} using a simple grid search, and report performance with the best values of these parameters. For the infinite horizon with sampling costs, we found that AK's expected total reward was significantly lower than that of the other policies, for both normal and Bernoulli sampling and across all values of ε^{AK}, α^{AK} and c. This is due to AK's tendency to occasionally stop too late in the presence of sampling costs. To improve its performance in the infinite horizon setting, we introduce an additional parameter $\overline{T}$ to the procedure and prevent AK from starting new sampling stages after a deterministic time $\overline{T}$. We then optimize over the values of $\varepsilon^{AK}, \alpha^{AK}$ and $\overline{T}$ and use the best values found. For PE, we also use a deterministic stopping rule $\tau = \overline{T}$ for the infinite horizon, and optimize over $\overline{T}$ to report its performance. We find that the best deterministic value of $\overline{T}$ for PE and AK is usually near the Bayes-optimal policy's expected stopping time.

Figure 2.1 shows that in all six problem settings, the Bayes-optimal policies significantly outperform PE. Because naive strategies like PE are the ones most commonly used in practice, we see a substantial potential of performance improvement by using optimal policies instead.

LD's poor performance seems surprising, but can be understood as follows. In most sample paths, LD allocates all its samples to one or two alternatives that have signifi-

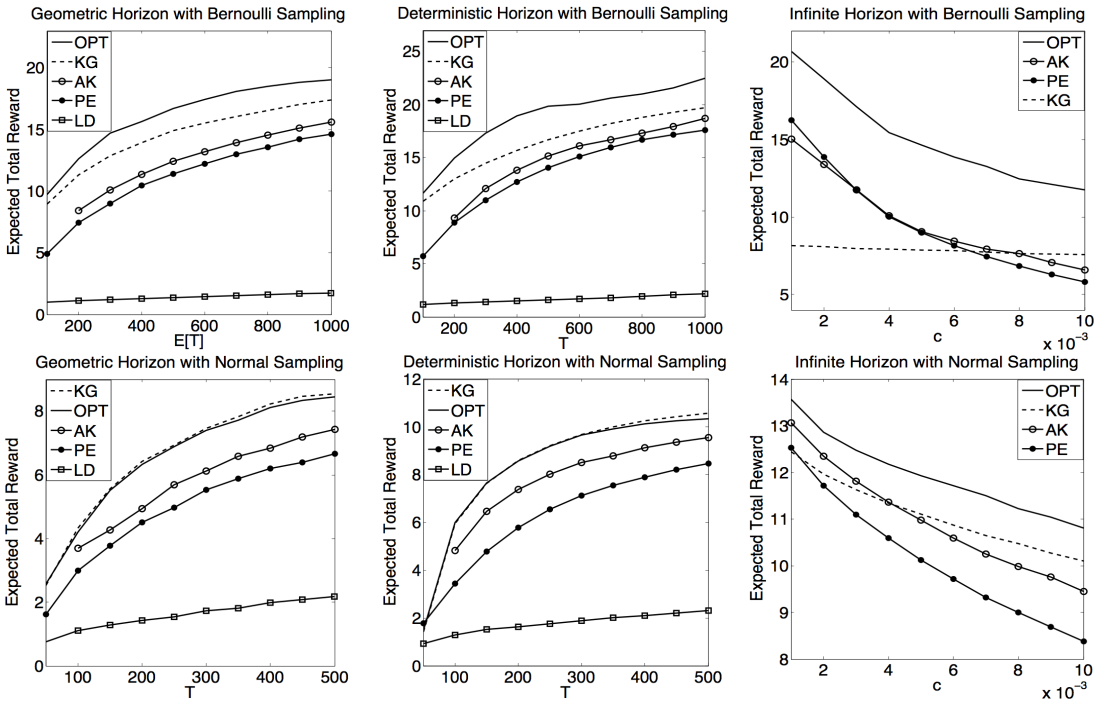

Figure 2.1: Performance of the following policies: Pure Exploration (PE), Large Deviations (LD), Andradóttir and Kim (AK), Knowledge Gradient (KG), and the approximate implementation of Bayes-Optimal (OPT) using necessary truncation and discretization. The maximum length of the 95% confidence intervals for the values in the plots is 0.24.

cantly smaller $|\theta_x - d_x|$ and hence significantly larger p_x^* (see Szechtman and Yücesan (2008)). As a result, the total reward earned by sampling, which is the increment in the number of correct determinations, is always below 3. Indeed, given a fixed or expected sample size N, the expected number of samples allocated to any alternative x with $p_x^* < \frac{1}{N}$ is less than 1. This can be the case for most of the k alternatives, especially when N/k is small. While LD has optimal asymptotic performance, this does not necessarily lead to good finite time performance.

We now look at the performance of AK. With normal sampling, it performs significantly better than PE, but is outperformed uniformly by the Bayes-optimal policy, and by KG when c is relatively large. The main reason is that AK looks for a feasible set

with the target tolerance level and PCD, but does not consider that the payoff of a high-quality determination may not be worth the sampling effort required to achieve it given a limited budget or positive sampling costs. With Bernoulli sampling, AK performs closer to PE. While Batur and Kim (2010) demonstrates that batching can be helpful in achieving approximate normality and satisfying statistical guarantees for difficult Bernoulli cases, where θ_x is close to 0 or 1, we implement AK directly without batching because our focus is on expected total reward, rather than satisfying statistical guarantees.

The performance of KG depends greatly on the problem. It is almost optimal in the geometric-horizon normal-sampling and the deterministic-horizon normal-sampling settings, while in the other four settings it is significantly suboptimal. Its worst performance comes in the infinite-horizon Bernoulli-sampling setting, where it is even worse than PE and MV with small values of c.

To understand the behavior of KG, first consider the two problem settings with normal sampling. KG makes its decisions using the one-step approximation $\mathscr{R}_x(S_{n,x})$ to the true value of sampling alternative x. This approximation is the sum of a one-step value of information (VOI) and the cost of sampling. As observed in Frazier and Powell (2010), Chick and Frazier (2011), the one-step VOI for normal sampling can significantly underestimate the true VOI when more samples will be taken later. This causes KG stopping rules to stop too soon (Chick and Frazier 2011), hurting their performance in problems with strictly positive sampling costs. This is likely to be the largest contributor to KG's suboptimality in the infinite-horizon normal-sampling problem.

Unlike the stopping decision, this underestimation of the true VOI has little effect on allocation decisions, because the level of underestimation is relatively constant across alternatives, and the alternative with the largest one-step VOI tends to also have near-maximal true VOI (Chick and Frazier 2011, Frazier et al. 2008, 2009). This is the rea-

son that KG does so well in the geometric-horizon normal-sampling and deterministic-horizon normal-sampling problem settings, where there are no sampling costs and the only decisions concern allocation. In this case, KG's performance is comparable with that of the Bayes-optimal policy. Although KG actually outperforms our implementation of the Bayes-optimal policy by a slight margin for large $\mathbb{E}[T]$ in the geometric horizon and large T in the deterministic horizon, this small gap is an artifact due to numerical inaccuracies introduced by discretizing the state-space when solving the DP that defines the Bayes-optimal policy, and would vanish with finer discretization. For discussions of discretization error in dynamic programming, see Powell (2007) and Bertsekas and Tsitsiklis (1996).

In problems with Bernoulli sampling, the discrete nature of the samples often causes the one-step VOI to be 0. This occurs when a single sample x_n, y_n is not enough to alter our decision on whether to place an alternative x in B_n, even when significant uncertainty about θ_x remains and more than one sample could alter our decision. In these situations, KG stops sampling immediately if there are positive sampling costs, or otherwise allocates its sample randomly (an inefficient strategy) among the alternatives. For this reason, KG performs poorly in both settings with Bernoulli sampling.

2.5.2 Ambulance Quality of Service Application

To demonstrate the Bayes-optimal policies in a more realistic application setting, we use it to analyze methods for positioning ambulances in a large city. We use the ambulance simulation introduced by Maxwell et al. (2010), which simulates ambulances responding to emergency calls in a city. The simulation model is very loosely based on the city of Edmonton, but is sufficiently modified in call arrival rates, etc., that the

results have no bearing on actual ambulance performance in that city. The city considers an emergency call to be answered on time if an ambulance arrives within 8 minutes, and otherwise it considers the call to be missed. We suppose that the city is considering several different static allocations of their fleet of 16 ambulances across the 11 bases in the city, and would like to know for each candidate allocation and each of several different call arrival rates whether it meets the minimum requirement of 70% of calls answered on time.

This is an MCS problem. Each alternative x corresponds to an hourly call arrival rate λ_x and an ambulance positioning plan. Based on these, each sample from x gives the number of calls answered on time during a two-week simulation. Since the emergency calls are generated according to a Poisson process, the expected total number of calls during two weeks for alternative x is known analytically as $M_x = 24 \times 14 \times \lambda_x$. Instead of directly measuring the fraction of calls answered on time in each simulation and estimating its expectation, we take θ_x/M_x as the long-term percentage of calls answered on time (see Henderson (2000)). The set of alternatives meeting or exceeding 70% of calls answered on time is therefore $\mathbb{B} = \{x : \theta_x/M_x \geq 0.7\} = \{x : \theta_x \geq d_x\}$, where $d_x = 0.7 \times M_x$. We chose 25 ambulance positioning plans and 25 values for the hourly call arrival rate from $[3, 6.6]$ for our experiment. This provides a collection of $25 \times 25 = 625$ alternatives.

The number of calls answered on time in a two-week simulation is approximately normally distributed. This was confirmed by visual examination of the empirical distribution for several randomly picked alternatives. We also assume a common sampling precision for all the alternatives. We confirmed that this assumption is reasonable by calculating and comparing the sampling precisions of several different alternatives chosen at random. To estimate the common sampling precision, we randomly chose 5

alternatives, sampled 20 times from each of them to estimate their individual sampling precisions, and used the average of the 5 sampling precisions as the estimate of the common sampling precision. This estimate was 1.4×10^{-3}. In problems with a high degree of variation in the sampling variances, one might instead estimate the sampling precisions separately for each alternative, or assume normal samples with unknown mean and unknown variance, with an inverse-gamma prior on the unknown sampling variance (DeGroot 1970). In this second case, the optimal policy could be computed by applying the theoretical results in this chapter to the exponential family of normal distributions with unknown mean and unknown variance, but computing solutions to these dynamic programs would require further work.

We use independent normal priors for each θ_x. We take a single sample from each alternative, set the prior mean μ_{0x} to this sampled value, and the prior precision β_{0x} to the common sampling precision. This is equivalent to using a non-informative prior and starting sampling by taking a single sample from each alternative. We then follow one of several different sampling policies for a deterministic horizon, where we investigate performance at a few fixed times. For the heuristic Bayes-optimal policy, we set $\alpha = 0.999$ (corresponding to a horizon of $1/(1-\alpha) = 1000$) and $c_x = 0$ for all x. We assume a 0-1 terminal payoff function with $m_{0x} = m_{1x} = 1$ for x, hence our estimate of set $\mathbb{B}$ at each time n is $B_n = \{x : \mu_{nx} \geq d_x\}$ by Table 1 and (2.15). For AK, we report its best performance over multiple sample paths with $n_0^{AK} = 2$, ε^{AK} chosen from $\{5, 10, 25, 50, 100, 110, 120, 130, 140, 150\}$, and α^{AK} chosen from $\{0.1, 0.3, 0.5, 0.7, 0.9, 0.99, 0.999, 0.9999, 0.99999, 0.999999\}$. As $1 - \alpha^{AK}$ gives a lower bound on AK's probability of correct decision, one would typically choose α^{AK} closer to 0, e.g., to 0.1 or 0.3. The set over which we optimize includes both these smaller values, as well as values closer to 1, because we found that increasing α^{AK} improved AK's performance.

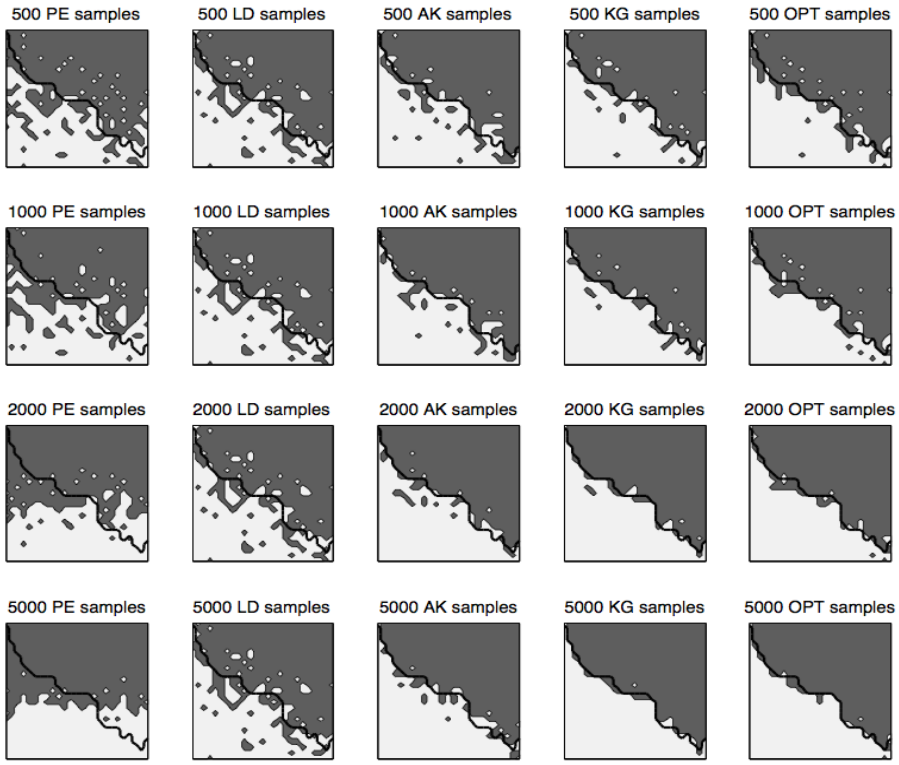

Figure 2.2: Performance of the sampling policies in the ambulance quality of service application: Pure Exploration (PE), Large Deviations (LD), Andradóttir and Kim (AK), Knowledge Gradient (KG), and Bayes-Optimal (OPT). In each plot, the black curve is the boundary of $\mathbb{B}$ (the set of alternatives answering at least 70% of the emergency calls on time); the light region is the estimate of the set $\mathbb{B}$ under the corresponding policy given the marked number of samples; and the dark region is the complement of the light region. Hourly call arrival rates $3, 3.15, 3.3, \ldots, 6.6$ are distributed along the vertical axis. Ambulance positioning plans are distributed along the horizontal axis, and are sorted to make the black line decreasing.

Figure 2.2 compares the Bayes-optimal policy against PE, LD, AK, and KG. The performance of each policy is measured by the similarity between $\mathbb{B}$ and $(B_n)_{n\geq 0}$, where we independently estimated $\mathbb{B}$ through exhaustive simulation of each θ_x. We also sorted the ambulance positioning plans used to construct the alternatives, in order of decreasing θ_x (at a fixed value of the call arrival rate), to make the set $\mathbb{B}$ easier to visualize. Each panel shows a black line, which is the independently obtained high-accuracy estimate

of the boundary between $\mathbb{B}$ and its complement. For each policy and after each of 500, 1000, 2000, and 5000 samples we plot the current estimate B_n as the light region, and the complement of B_n as the dark region.

Figure 2.2 shows that PE behaves poorly in distinguishing among the alternatives. Under this policy, after 5000 samples total, approximately $5000/625 = 8$ samples have been taken from each alternative. For those alternatives x with θ_x close to d_x, this number of samples is much too small to accurately estimate whether x is in $\mathbb{B}$ or not. Moreover, the estimates given by LD barely change as the number of samples increases. The reason is that it only samples from 2 alternatives out of 625 in the 5000 samples. In contrast, AK, KG and the Bayes-optimal policy are much more efficient, while the latter two significantly outperform AK. AK's best performance occurs at a very large tolerance level, $\varepsilon^{AK} = 130$, and a very small lower bound on the PCD, $1 - \alpha^{AK} = 10^{-4}$. We believe this is because the budget we are considering is much smaller than AK would typically require to provide a meaningful bound on PCD with a large number of alternatives ($k = 625$). The excellent performance of KG relative to optimal should not be surprising: samples are approximately normally distributed and there are no sampling costs, which is the setting from Section 2.5.1 in which KG was nearly optimal. Had the problem used Bernoulli sampling or strictly positive sampling costs, then KG likely would not have performed as well.

As shown in Figure 2.2, after 5000 samples under KG or the Bayes-optimal policy, we have estimated $\mathbb{B}$ with a high degree of accuracy. Indeed, with only 500 samples from either of these policies, we estimate $\mathbb{B}$ with greater accuracy than is possible with 5000 samples under PE or LD. This factor of 10 in sampling effort represents a dramatic savings of simulation time, and demonstrates the value of using an optimal or near-optimal sampling policy when performing MCS.

2.5.3 Revenue of a Production Line Application

We consider a manufacturing firm that can choose, on each day, whether or not to operate a production line. If it operates the line, it earns a stochastic net revenue that can be positive or negative. If it chooses not to operate the line, its net revenue is 0.

At the beginning of each day, the firm observes market and operational conditions X. Each X takes one of $k < \infty$ values. The firm has a simulator that, given X, can simulate a net daily revenue $Y(X)$. We suppose that this simulator is cumbersome, taking a long time to run, and hence the plant manager does not want to run it at the beginning of each day. Instead, at time 0, the manager would like to use simulation to estimate $\theta_x = \mathbb{E}[Y(X) \mid X = x]$ for a wide variety of conditions x, and then estimate the set $\mathbb{B} = \{x : \theta_x \geq 0\}$ in which it is profitable to operate the production line. The manager can then make decisions when each X becomes known with a simple look up table based on this estimate of $\mathbb{B}$.

Suppose that the plant manager recreates the look up table every month. From historical data and forecasts, she has estimated the long-run percentage of time each condition x occurs as q_x. Thus the expected monthly net revenue she earns from an estimate B of $\mathbb{B}$ is a one-sided linear terminal payoff with $d_x = 0$ and $m_x = 30q_x$ for all x, i.e., $r(B;\theta,d) = \sum_{x\in B} m_x\theta_x$.

In performing her simulations, the plant manager has no time limit, but she is using a third-party computing service such as Amazon EC2 to perform her computation. This service charges her a monetary cost of c_x for each simulation of condition x, which is determined by the length of time that this simulation requires to run and the unit cost per CPU hour determined from the computing service's pricing structure (Amazon.com 2012).

The simulation of the production line we consider is based on Buchholz and Thummler (2005) and "Optimization of a Production Line" in SimOpt.org (2011). It has 4 service queues arranged serially, each of which has a fixed finite capacity 5. Parts (customer orders) arrive to queue 1 according to a Poisson process with rate λ. Each queue sends the parts into a corresponding single server with first-come-first-serve discipline, and the service time is exponentially distributed with rate γ. Parts leaving queue i after service are immediately transferred to the next queue $i+1$ (if possible). Whenever the service queue $i+1$ is full, the server i is said to be blocked and the part in it cannot leave even if it is completed, since there is no room in the next queue.

We have $k = 500$ alternatives in our experiment. Each alternative condition x corresponds to a customer arrival rate λ_x in $\{5.1, 5.2, \ldots, 7.5\}$ and a queue service rate γ_x in $\{5.1, 5.15, \ldots, 6.05\}$. Denote by Γ_x the expected number of parts leaving the last queue (expected number of completed orders) in an 8-hour period under condition x. We assume that the net revenue per order filled is \$50, and operation of the production line over an 8-hour period has a fixed basic cost of \$2800. The expected revenue of the production line under condition x is then $\theta_x = 50\Gamma_x - 2800$. The level set we wish to estimate is thus $\mathbb{B} = \{x : \Gamma_x \geq 56\}$.

The number of parts leaving the last queue in an 8-hour simulation is approximately normally distributed, which was confirmed by visual examination of the empirical distribution for several randomly picked alternatives. As in the ambulance quality of service application, we assume a common sampling precision for all the alternatives and estimate it by calculating and averaging over the sampling precisions of several different alternatives chosen at random. We also use independent normal priors for each θ_x, where we set μ_{0x} to be the value of an initial sample, and set β_{0x} to be the common sampling precision. We then follow one of several different sampling policies, assuming an

Table 2.4: Performance of the Sampling Policies in the Revenue of a Production Line Application

Sampling Policy	PE	AK	KG	OPT
expected terminal payoff: $\mathbb{E}\left[\sum_{x\in B_\tau} m_x\theta_x\right]$	2709	2748	2667	2859
expected sampling cost: $\mathbb{E}\left[\sum_{n=1}^{\tau} c_{x_n}\right]$	198	195	84	198
expected total reward: $\mathbb{E}\left[\sum_{x\in B_\tau} m_x\theta_x - \sum_{n=1}^{\tau} c_{x_n}\right]$	2511	2553	2583	2661

infinite horizon with $q_x = 1/500$, representing homogeneous long-run percentages for each condition, $m_x = 30q_x$ as described above, $c_x = 0.06$, and $\beta_x^{\varepsilon} = 1.2\times 10^{-3}$ for all x. This sampling cost c_x of \$0.06 per simulation corresponds to a computing service that charges \$0.12 per CPU hour, and a simulation that takes 30 minutes for each replication. Our estimate of the set $\mathbb{B}$ when we stop at time τ is then $B_\tau = \{x : \mu_{\tau x} \geq 0\}$ by Table 1. To examine the expected total reward under each policy, we independently estimate each θ_x through exhaustive simulation.

Table 2.4 compares the Bayes-optimal policy against three other policies: PE, AK, and KG, showing their expected terminal payoff, expected sampling cost, and expected total reward by averaging over 2000 independent sample paths. The maximum length of the 95% confidence intervals for the values in the table is 3. Similar to Section 2.5.1, we report the performance of PE with a best deterministic stopping rule $\tau = \overline{T}$, and report the performance of AK with best pre-specified values of $\varepsilon^{AK}, \alpha^{AK}$ and $\overline{T}$. Our results show that the best deterministic value of $\overline{T}$ for PE and AK is very close to the expected stopping time under the Bayes-optimal policy. While PE, AK and KG are all sub-optimal, KG is the best among the three sub-optimal policies. Though it stops too soon, its advanced efficiency in allocating samples still results in a relatively high expected total reward.

We also performed an additional experiment to assess the degradation of the Bayes-optimal policy caused by approximating the sampling precisions as known. We find that an optimal policy with a high-fidelity estimate of each alternative's individual sampling precision obtained through exhaustive simulation has an expected total reward of 2664, which offers a 0.1% improvement over our previous implementation. We see that in this particular example, the degradation is very small.

2.6 Conclusions

By applying methods from multi-armed bandits and optimal stopping, we are able to efficiently solve the dynamic program for the Bayesian MCS problem and find Bayes-optimal fully sequential sampling and stopping policies. While researchers have searched for Bayes-optimal policies for other related problems in sequential experimental design and effort allocation for simulation, tractable computation of the optimal policies has remained elusive in many problems, and so the results in this chapter place the MCS problem together with a select group of problems in sequential experimental design for which the sequential Bayes-optimal policies can be computed efficiently.

The Bayes-optimal policies presented are flexible, allowing limitations on the ability to sample to be modeled with either a random horizon or sampling costs or both, allowing sampling distributions from any exponential family, and allowing a broad class of terminal payoff functions. While they do not allow for problems with fixed horizon, the optimal policy for random horizons can be used heuristically in such situations. We provide explicit computations for Bernoulli sampling and normal sampling with known variance. We also provide expressions for the KG policy and show that it works extremely well for normal sampling with a geometrically distributed horizon and no sam-

pling costs. Although the KG policy is not Bayes optimal, and results in some performance loss, its ease of use may make it attractive to practitioners facing MCS problems of this type.

In conclusion, the results in this chapter provide new tools for simulation analysts facing MCS problems. These new tools dramatically improve efficiency over naive sampling methods, and make it possible to efficiently and accurately solve previously intractable MCS problems.

CHAPTER 3

BAYESIAN OPTIMIZATION VIA SIMULATION WITH PAIRWISE SAMPLING AND CORRELATED PRIOR BELIEFS

We consider discrete optimization via simulation, in which we have a discrete set of alternative systems whose performance can each be evaluated via stochastic simulation, and we wish to allocate a limited simulation budget among them to find one whose expected performance is as large as possible. See Section 1.3 for reviews of literature.

We study this problem in a Bayesian context, where we place a prior probability distribution on the values of the alternatives, and use value of information (VOI) calculations within a knowledge-gradient (KG) sampling algorithm to decide which alternative, or collection of alternatives, would be most useful to sample next. The advantage of doing so is that making decisions based on the VOI automatically addresses the exploration versus exploitation tradeoff, and tends to reduce the number of function evaluations required on average to reach a given solution quality, potentially (but not necessarily) at the cost of requiring more computation to decide where to sample.

The prior probability distribution that we consider is a multivariate normal distribution, and allows for correlation in our prior belief between two alternatives. This models a belief that two alternatives with similar characteristics often have similar expected performance, and allows the algorithm that we construct to do well even in problems where the number of alternatives is much larger than the number of samples that we can take.

We allow common random numbers (CRN), in which multiple alternatives are simulated using the same stream of random numbers. This induces correlation in the noise, which can be advantageous for optimization when the correlation is positive, because it allows more accurate estimation of the differences between alternatives' values.

Our analysis differs from this previous literature by allowing the use of CRN. This has been perceived to be difficult, because sampling with CRN makes it difficult to compute the VOI, and to maintain a closed-form posterior distribution. We overcome these difficulties by calculating the VOI for observing the *difference* in value between two alternatives, which can be done analytically, and by calculating the posterior with adaptively updated point estimates of the noise covariance. We show that, in the context of VOI-based algorithms, using CRN can greatly improve performance.

Sampling with correlated means and CRN in the Bayesian setting using VOI methods has been considered by Chick and Inoue (2001a), but assumed two-stage sampling rather than fully sequential sampling, and restricted attention to conjugate prior distributions for the unknown means. Others have considered sampling with CRN in the optimal computing budget allocation framework (Fu et al. 2004), in the indifference-zone setting (Clark and Yang 1986, Nelson and Matejcik 1995), and in the multiple comparisons problem (Yang and Nelson 1991, Nakayama 2000, Kim 2005). The current work differs from this previous work in its focus on problems with many alternatives, enabled by a multivariate normal prior distribution with arbitrary covariance.

We present three techniques that reduce the computation required to find a point, or pair of points, with a large VOI. The first is to use the gradient of the VOI in performing this search, calculating it over an embedding of our discrete alternatives into a continuous space. This use of the gradient of the VOI differs from the more common use of gradients of the response surface in optimization. The second is to consider a VOI with a restricted set of implementation decisions. The third is to use data structures that avoid enumerating alternatives, instead tracking only those alternatives that have been sampled, and reconstructing required portions of the posterior distribution as needed. This is standard in GP regression, but contrasts with previous work on optimization via

simulation with CRN (Clark and Yang 1986, Nelson and Matejcik 1995, Chick and Inoue 2001a, Fu et al. 2004). These three techniques were applied in Scott et al. (2011) to a continuous setting without CRN.

We also provide an almost sure guarantee of convergence to the global optimum, as the number of samples taken grows without bound, when parameters are known. In addition to allowing correlated sampling, this theoretical result contrasts with Scott et al. (2011) in having conditions that are easier to verify. It also contrasts with other work that focuses on convergence to local optima (Hong and Nelson 2006, Xu et al. 2010, Wang et al. 2013).

The current chapter extends a report of our preliminary work (Frazier et al. 2011) in a number of ways. It provides an enhanced version of the algorithm that scales to much larger problems, a theoretical analysis showing convergence to a global optimum, a derivation of a maximum likelihood estimation method for estimating covariance parameters from samples observed with CRN, and additional numerical comparisons with other algorithms on larger problems.

We begin in Section 3.1 by formally defining our problem and the statistical model in which we perform inference. Section 3.2 describes a generic sampling algorithm that forms the basis for specific sampling algorithms defined later in the chapter. Section 3.3 defines the VOI and the corresponding KG factor, and shows how it can be computed in the context of optimization via simulation with correlated sampling. Section 3.4 takes these VOI and KG computations, and uses them to create allocation rules for the KG sampling algorithm. Section 3.5 states theoretical results on consistency of KG algorithms, showing that these algorithms can produce consistent estimates of the global optimum in the limit as the sampling budget grows large, when parameters are known. Section 3.6 discusses practical implementation issues, regarding prior distributions and

computation of the KG algorithm's decisions. Numerical results in Section 3.7 show a distinct advantage to the ability to sequentially sample with CRN in discrete optimization via simulation problems. Appendices prove theoretical results and derive gradient and statistical estimation results used in the algorithm.

3.1 Sampling Model and Mechanism for Posterior Inference

Consider a collection of k alternatives with stochastic performance. If we sample from all k alternatives together using CRN, then we observe a normal random vector. Let the mean vector of this normal distribution be $\theta = [\theta(1), \ldots, \theta(k)]^T$, and let its covariance matrix be Λ, where ^{T} denotes matrix transposition. We wish to find the alternative x with the largest sampling mean $\theta(x)$.

We use a Bayesian formulation, in which we begin with a multivariate normal prior on θ,

$$\theta \sim \mathcal{N}(\mu_0, \Sigma_0). \tag{3.1}$$

The choice of Σ_0 allows for conjugate prior distributions for θ (Chick and Inoue 2001a) or for GP priors (Rasmussen and Williams 2006), which are related to kriging models (Cressie 1993). A parametric family can be used to specify μ_0 and Σ_0 in terms of a function taking the alternatives and few additional parameters as arguments. In practice, the parameters specifying μ_0 and Σ_0, as well as the sampling covariance Λ, are unknown, but we will initially assume they are fully known for simplicity. Then, we will relax this assumption in Section 3.6.

In this chapter, the ith entry of a length-k vector v (e.g., θ and μ_0) is written $v(i)$, and the (i, j)th entry of a k-by-k matrix M (e.g., Σ_0 and Λ) is written $M(i, j)$. Moreover, for an ordered collection of m alternatives $\vec{x} = (x^{(1)}, x^{(2)}, \ldots, x^{(m)})$ with elements $x^{(i)} \in$

$\{1,2,\ldots,k\}$ for each i, we use $v(\vec{x})$ to denote the length-m sub-vector of v with the ith entry equal to $v\left(x^{(i)}\right)$. Let $\vec{x}'$ with elements in $\{1,2,\ldots,k\}$ be another vector of alternatives with m' entries. We denote by $M(\vec{x},\vec{x}')$ the m-by-m' sub-matrix of M with the (i,j)th entry equal to $M\left(x^{(i)},x'^{(j)}\right)$.

3.1.1 Sampling Model and Distribution of Outputs

At each time $n = 1,2,\ldots$ we choose a set of the alternatives to sample, specified as a row vector $\vec{x}_n$ with elements in $\{1,2,\ldots,k\}$, and sample each of the chosen alternatives once using CRN. Each alternative may appear at most once in $\vec{x}_n$. We then observe a column vector $\vec{y}_n$, with one entry for each alternative sampled. The conditional distribution of $\vec{y}_n$ given $\vec{x}_n$, θ is assumed to be Gaussian and independent of previous observations,

$$\vec{y}_n \mid \theta,\vec{x}_n,(\vec{x}_m,\vec{y}_m : m < n) \sim \mathcal{N}\left(\theta\left(\vec{x}_n\right),\Lambda\left(\vec{x}_n,\vec{x}_n\right)\right). \tag{3.2}$$

Although (3.2) is general, in our algorithm below, the sampling decision $\vec{x}_n$ is either a singleton x_n, with corresponding observation y_n, or a pair of alternatives $\left(x_n^{(1)},x_n^{(2)}\right)$, with corresponding observations $(y_n^{(1)},y_n^{(2)})$. The notation $\vec{x}_n$ and $\vec{y}_n$ indicates the general case, in which one or more alternatives is sampled, while x_n and y_n always indicates a single alternative. The sampling distribution of (3.2) for these two cases (singletons and pairs) are

$$\begin{aligned} y_n \mid \theta,x_n &\sim \mathcal{N}\left(\theta(x_n),\Lambda(x_n,x_n)\right), \quad \text{and} \\ (y_n^{(1)},y_n^{(2)}) \mid \theta,\left(x_n^{(1)},x_n^{(2)}\right) &\sim \mathcal{N}\left(\begin{bmatrix}\theta\left(x_n^{(1)}\right)\\ \theta\left(x_n^{(2)}\right)\end{bmatrix},\begin{bmatrix}\Lambda\left(x_n^{(1)},x_n^{(1)}\right) & \Lambda\left(x_n^{(1)},x_n^{(2)}\right)\\ \Lambda\left(x_n^{(2)},x_n^{(1)}\right) & \Lambda\left(x_n^{(2)},x_n^{(2)}\right)\end{bmatrix}\right). \end{aligned}$$

These sampling distributions are sufficient for calculating posterior distributions

from observations in the sampling algorithms that we propose, but when computing the VOI in Section 3.3 below, we will also consider three additional sampling distributions. First, we will consider the sampling distribution of observing only the *difference* between a pair $\left(x_n^{(1)}, x_n^{(2)}\right)$ of alternatives,

$$y_n^{(1)} - y_n^{(2)} \mid \theta, \left(x_n^{(1)}, x_n^{(2)}\right) \sim \mathcal{N}\left(\theta(x_n^{(1)}) - \theta(x_n^{(2)}), \Lambda(x_n^{(1)}, x_n^{(1)}) + \Lambda(x_n^{(2)}, x_n^{(2)}) - 2\Lambda(x_n^{(1)}, x_n^{(2)})\right).$$

Second, we will consider the sampling distribution of observing not necessarily one but $\beta_n \geq 1$ vectors of samples from the distribution given by (3.2), each generated using an independent CRN stream. We do this to compute an average VOI per sample. The value of β_n can be fixed beforehand, or can be chosen adaptively. We generalize $\vec{y}_n$ to refer to the average of these β_n observations, so

$$\vec{y}_n \mid \theta, \vec{x}_n, \beta_n \sim \mathcal{N}\left(\theta\left(\vec{x}_n\right), \Lambda\left(\vec{x}_n, \vec{x}_n\right)/\beta_n\right). \tag{3.3}$$

Third, we will consider the sampling distribution of observing $\beta_n \geq 1$ independent differences between a pair $\left(x_n^{(1)}, x_n^{(2)}\right)$, continuing to let $\vec{y}_n = (y_n^{(1)}, y_n^{(2)})$ denote the average of these observations,

$$y_n^{(1)} - y_n^{(2)} \mid \theta, \left(x_n^{(1)}, x_n^{(2)}\right), \beta_n \sim \mathcal{N}\left(\theta(x_n^{(1)}) - \theta(x_n^{(2)}), \beta_n^{-1}\left[\Lambda(x_n^{(1)}, x_n^{(1)}) + \Lambda(x_n^{(2)}, x_n^{(2)}) - 2\Lambda(x_n^{(1)}, x_n^{(2)})\right]\right). \tag{3.4}$$

These last three sampling distributions are used only to compute the VOI. In the sampling algorithms that we propose, we always observe from both alternatives when sampling from a pair, and take only one sample at a time from a singleton or pair even when we calculate a VOI with $\beta_n > 1$.

3.1.2 Posterior Distribution for Unknown Means and its Computation

With the sampling scheme in (3.2), and the assumption that the sampling covariance matrix Λ is known, we can compute a closed-form expression for the posterior distribution on θ. We let $\mathbb{E}_n$ and Var_n indicate the conditional expectation and variance respectively with respect to the data $\vec{x}_1, \vec{y}_1, \vec{x}_2, \vec{y}_2, \ldots, \vec{x}_n, \vec{y}_n$, where each $\vec{y}_n$ is sampled according to (3.2). Define $\mu_n = \mathbb{E}_n \theta$ and $\Sigma_n = \mathrm{Var}_n \theta$. The posterior distribution on θ is normal (see, e.g., Gelman et al. 2004, Sec. 14.6),

$$\theta \mid \vec{x}_1, \vec{y}_1, \vec{x}_2, \vec{y}_2, \ldots, \vec{x}_n, \vec{y}_n \;\sim\; \mathcal{N}(\mu_n, \Sigma_n),$$

where the posterior mean μ_n and variance Σ_n can be computed analytically, either directly from the prior and the full data, or recursively, updating as each new datapoint $\vec{x}_n, \vec{y}_n$ is added.

When the number of alternatives k is large, it is computationally infeasible to store all of μ_n and Σ_n, because Σ_n is a k-by-k matrix. Therefore, we use a method commonly used in GP regression, which calculates the posterior distribution on the sampled alternatives and any desired additional alternatives, without requiring a k-by-k matrix. We briefly describe this method here, giving some notation to be used later, and focusing on singletons and pairs.

Let $\mathscr{X}_n$ denote the cumulative row vector of alternatives sampled from time 1 to time n, i.e., the concatenation of $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n$ into a row. Alternatives appear more than once if they are sampled more than once. For example, if $\vec{x}_1 = x_1$ and $\vec{x}_2 = \left(x_2^{(1)}, x_2^{(2)}\right)$, then $\mathscr{X}_1 = (x_1)$ and $\mathscr{X}_2 = \left(x_1, x_2^{(1)}, x_2^{(2)}\right)$. In addition, if $x_1 = x_2^{(1)} = x$ then $\mathscr{X}_2 = \left(x, x, x_2^{(2)}\right)$.

In Section 3.3 we will compute the VOI for an arbitrary (singleton or pair) sampling

decision $\vec{x}$ at time $n+1$. Let the vector $\mathscr{X}_{n,\vec{x}}$ denote the row concatenation of $\mathscr{X}_n$ and $\vec{x}$. To compute the VOI, we require the posterior distribution on $\theta\left(\mathscr{X}_{n,\vec{x}}\right)$, which is multivariate normal with mean $\mu_n\left(\mathscr{X}_{n,\vec{x}}\right)$ and covariance $\Sigma_n\left(\mathscr{X}_{n,\vec{x}},\mathscr{X}_{n,\vec{x}}\right)$. We introduce the following expressions for computing these quantities. Let $\mathscr{Y}_n$ be the cumulative column vector of sampling observations up to time n, i.e., the columnar concatenation of $\vec{y}_1,\vec{y}_2,\ldots,\vec{y}_n$, so each entry of $\mathscr{Y}_n$ is the observation from the corresponding entry in $\mathscr{X}_n$. Let Γ_n be the block diagonal matrix with n blocks: $\Lambda(\vec{x}_1,\vec{x}_1),\Lambda(\vec{x}_2,\vec{x}_2),\ldots,\Lambda(\vec{x}_n,\vec{x}_n)$. We then define three quantities, the measurement residual $\widetilde{\mathscr{Y}_n}$, the residual covariance S_n, and the optimal Kalman gain $K_n(\vec{x})$, by

$$\widetilde{\mathscr{Y}_n} = \mathscr{Y}_n - \mu_0\left(\mathscr{X}_n\right), \quad S_n = \Sigma_0\left(\mathscr{X}_n,\mathscr{X}_n\right)+\Gamma_n, \quad K_n(\vec{x}) = \Sigma_0\left(\mathscr{X}_{n,\vec{x}},\mathscr{X}_{n,\vec{x}}\right) L\left[S_n\right]^{-1}. \tag{3.5}$$

Here, the matrix L is defined by concatenating an $|\mathscr{X}_n|$-by-$|\mathscr{X}_n|$ identify matrix with an $|\mathscr{X}_n|$-by-$|\vec{x}_n|$ matrix of zeros, so $L=[I_{|\mathscr{X}_n|},\vec{0}]^T$ if $\vec{x}=x$, and $L=[I_{|\mathscr{X}_n|},\vec{0},\vec{0}]^T$ if $\vec{x}=\left(x^{(1)},x^{(2)}\right)$. Here and elsewhere, $|\cdot|$ denotes the length of a vector. We will assume in Section 3.5 that Σ_0 and Λ are positive definite. That assumption implies that $\Sigma_0\left(\mathscr{X}_n,\mathscr{X}_n\right)$ is positive semidefinite and that Γ_n is positive definite, so that S_n is positive definite and that its inverse $[S_n]^{-1}$ exists.

The posterior mean and covariance matrix of $\theta\left(\mathscr{X}_{n,\vec{x}}\right)$ at time n are then given respectively by

$$\mu_n\left(\mathscr{X}_{n,\vec{x}}\right) = \mu_0\left(\mathscr{X}_{n,\vec{x}}\right) + K_n(\vec{x})\widetilde{\mathscr{Y}_n}, \tag{3.6}$$

$$\Sigma_n\left(\mathscr{X}_{n,\vec{x}},\mathscr{X}_{n,\vec{x}}\right) = \left(I_{|\mathscr{X}_{n,\vec{x}}|} - K_n(\vec{x})L^T\right)\Sigma_0\left(\mathscr{X}_{n,\vec{x}},\mathscr{X}_{n,\vec{x}}\right). \tag{3.7}$$

In implementing (3.5), one should not invert S_n directly, as doing so when n is large is numerically unstable. Instead, one can perform a Cholesky decomposition, and then

solve a numerical system, as is described in Sec. 2.2 of Rasmussen and Williams (2006). This is more stable, and faster. For further discussion of implementation issues in GP regression, see Rasmussen and Williams (2006).

3.2 Generic Sampling Algorithm

We now formalize our proposed DOvS algorithm. The notation in Section 3.1 allows us to formalize it in a way that is amenable to handling a very large number of alternatives: statistics are tracked only for alternatives that have been sampled or are being considered for sampling in the next stage.

The algorithm samples in a sequential manner. This requires the specification of an allocation rule, which maps $\mathscr{X}_n, \mathscr{Y}_n$ to a set of alternatives to sample next, and a stopping rule, which decides whether or not to stop sampling. The allocation rules we use are based on VOI principles described in Section 3.3 and are presented in Section 3.4. The default stopping rule we use in this chapter is to stop after a pre-specified number of samples is observed.

The generic algorithm below is written to be able to handle either a known or an unknown sampling covariance matrix Λ. When it is unknown, as is typical in applications, the sampling covariance parameters are estimated. In this case, we also maintain estimates of the parameters μ_0 and Σ_0 defining the prior distribution in an empirical Bayes fashion, as described below.

1. **Initialize:** Select an allocation rule and a stopping rule. If the sampling covariance Λ and the mean vector μ_0 and the covariance matrix Σ_0 for the unknown sampling means θ are known, then specify these parameters, initialize $n = 0$ to

be the number of stages of sampling done so far, and initialize $\mathscr{X}_0$ and $\mathscr{Y}_0$ to be empty vectors. If Λ, μ_0 and Σ_0 are not all known, then describe the functional forms of Λ, μ_0 and Σ_0 in terms of a collection of parameters (see Section 3.6.1), and take an initial stage of samples to estimate those parameters, setting n, $\mathscr{X}_n$ and $\mathscr{Y}_n$ accordingly (see Section 3.6.2).

2. **Update parameters (Empirical Bayes):** If the parameters determining Λ are unknown and their estimates are to be updated, then use the maximum likelihood estimator described in Section 3.6.2 to estimate them using all data (collected in $\mathscr{X}_n$ and $\mathscr{Y}_n$).

3. **Check allocation and stopping rule:** If the stopping rule says to stop sampling, go to Step 5. Otherwise, use the allocation rule to choose a set of alternatives, $\vec{x}_{n+1}$, to sample next.

4. **Sample:** Sample $\vec{y}_{n+1}$ using CRN according to (3.2) with the chosen $\vec{x}_{n+1}$. Concatenate $\vec{y}_{n+1}$ with $\mathscr{Y}_n$ to get $\mathscr{Y}_{n+1}$, and $\vec{x}_{n+1}$ with $\mathscr{X}_n$ to get $\mathscr{X}_{n+1}$. Increment n and go back to Step 2.

5. **Selection rule:** Select as the best the alternative in $\mathscr{X}_n$ with the largest posterior mean. The can be found by computing $\mu_n(\mathscr{X}_n)$ according to (3.6) with $\mathscr{X}_{n,\vec{x}} = \mathscr{X}_n$, and then taking the largest component of this vector.

3.3 Value of Information

In this section we derive analytic expressions for computing the VOI, resulting from sampling singletons, or sampling the difference between pairs of alternatives. These VOI calculations are then used to derive our allocation rules in Section 3.4 for use in the algorithm of Section 3.2.

VOI is a concept which encompasses the expected value of sample information (EVSI) and the expected value of perfect information (EVPI) (Raiffa and Schlaifer 1968). Information is valued according to the expected improvement it produces in some decision to be made later. In this chapter, the decision to be made later is which alternative to select as the best and to implement in reality. We call this decision the "implementation decision." The value of an implementation decision x is $\theta(x)$ and has expectation $\mu_{n+1}(x)$ under the posterior at time $n+1$. Thus, the expected value of the best implementation decision that can be made at time $n+1$ is $\max_{x\in\{1,2,\ldots,k\}} \mu_{n+1}(x) = \max \mu_{n+1}$. The increment in this value in going from time n to time $n+1$ is $\max \mu_{n+1} - \max \mu_n$ and depends on y_{n+1}. Here, the VOI is the expected value of this increment, under the posterior at time n, under the hypothetical that an alternative is to be selected after a single stage of sampling.

In this framework, the VOI for a set of β samples collected by observing $\vec{y}_{n+1}$ with a general sampling decision $\vec{x}$ at time $n+1$ according to (3.2) can be written

$$V_n(\vec{x},\beta) = \mathbb{E}_n\left[\max \mu_{n+1} \mid \vec{x}_{n+1} = \vec{x}, \beta_{n+1} = \beta\right] - \max \mu_n. \tag{3.8}$$

If the implementation decision is restricted to a set $A_n(\vec{x})$ that may depend upon on $\mathscr{X}_n, \mathscr{Y}_n$ and $\vec{x}$, then the VOI is

$$V_n\left(\vec{x}, A_n(\vec{x}), \beta\right) = \mathbb{E}_n\left[\max\left[\mu_{n+1}\left(A_n(\vec{x})\right)\right] \mid \vec{x}_{n+1} = \vec{x}, \beta_{n+1} = \beta\right] - \max\left[\mu_n\left(A_n(\vec{x})\right)\right]. \tag{3.9}$$

When $A_n(\vec{x}) = \{1,2,\ldots,k\}$, then $V_n(\vec{x},A_n(\vec{x}),\beta) = V_n(\vec{x},\beta)$. This VOI also satisfies a monotonicity property: if $A \subseteq B$ then $V_n(\vec{x},A,\beta) \leq V_n(\vec{x},B,\beta)$. This monotonicity property implies that $V_n(\vec{x},A_n(\vec{x}),\beta)$ is actually a lower bound on $V_n(\vec{x},\beta)$.

There is no restriction on the implementation decision in practice, but we use $V_n(\vec{x},A_n(\vec{x}),\beta)$ as an approximation to $V_n(\vec{x},\beta)$ because it can be computed more

quickly, especially when $|A_n(\vec{x})|$ is small. Methods for choosing $A_n(\vec{x})$ are discussed in Section 3.6.3.

3.3.1 Predictive Distribution for Posterior Means to be Observed

The VOI in (3.8) or (3.9) depends on the predictive distribution for $\mu_{n+1}(A)$ that results from a particular decision to sample $\vec{x}_{n+1}$ for β_{n+1} times, for any given set A. We consider two specific types of sampling decisions $\vec{x}_{n+1}$: observing singletons $\vec{x}_{n+1} = (x_{n+1})$ as in (3.3); and observing the difference between a pair of alternatives $\vec{x}_{n+1} = \left(x_{n+1}^{(1)}, x_{n+1}^{(2)}\right)$ as in (3.4). Observing either the singleton y_n or the difference $y_n^{(1)} - y_n^{(2)}$ admits an analytic expression for $V_n(\vec{x}, A, \beta)$ below. Observing both $y_n^{(1)}$ and $y_n^{(2)}$ together does not: we use the VOI of sampling their difference as a lower bound on the VOI of observing both values. This lower bound proves to be useful in numerical experiments.

For both singletons and differences between pairs, the predictive distribution is

$$\mu_{n+1}(A) \mid \mathscr{X}_n, \mathscr{Y}_n, \vec{x}_{n+1}, \beta_{n+1} \sim \mathscr{N}\left(\mu_n(A), \widetilde{\sigma}_n(\vec{x}_{n+1}, A, \beta_{n+1})\, \widetilde{\sigma}_n(\vec{x}_{n+1}, A, \beta_{n+1})^T\right), \tag{3.10}$$

where $\widetilde{\sigma}_n(\vec{x}_{n+1}, A, \beta_{n+1})$ is a $|A| \times 1$ vector defined respectively in the two cases as

$$\begin{aligned} \widetilde{\sigma}_n(x, A, \beta) &= \frac{\Sigma_n(A, x)}{\sqrt{\beta^{-1}\Lambda(x,x) + \Sigma_n(x,x)}}, \\ \widetilde{\sigma}_n\left(\left(x^{(1)}, x^{(2)}\right), A, \beta\right) &= \frac{\Sigma_n\left(A, x^{(1)}\right) - \Sigma_n\left(A, x^{(2)}\right)}{\sqrt{\beta^{-1}P + Q_n}}, \end{aligned} \tag{3.11}$$

which follows directly from Frazier et al. (2011, Sec. 2.2). Here, $\Sigma_n(A, x)$ is a column vector containing the entries from Σ_n in column x with rows in A, and P and Q_n are

defined by

$$P = \Lambda\left(x^{(1)}, x^{(1)}\right) + \Lambda\left(x^{(2)}, x^{(2)}\right) - 2\Lambda\left(x^{(1)}, x^{(2)}\right),$$
$$Q_n = \Sigma_n\left(x^{(1)}, x^{(1)}\right) + \Sigma_n\left(x^{(2)}, x^{(2)}\right) - 2\Sigma_n\left(x^{(1)}, x^{(2)}\right). \tag{3.12}$$

This expression will be used in Section 3.3.2 to compute the VOI in (3.9) explicitly.

3.3.2 Evaluation of the Value of Information

We now provide explicit expressions for the VOI in (3.9) under observations of singletons and of differences between pairs. From (3.10), we know that when $\mathscr{X}_n$, $\mathscr{Y}_n$, $\vec{x}_{n+1}$ and β_{n+1} are given, $\mu_{n+1}(A)$ is equal in distribution to $\mu_n(A) + \widetilde{\sigma}_n(\vec{x}_{n+1}, A, \beta_{n+1})Z$, where Z is a standard normal random variable. Using this observation in (3.9) shows that

$$V_n\left(\vec{x}, A_n(\vec{x}), \beta\right) = \mathbb{E}_n\left[\max\left[\mu_n\left(A_n(\vec{x})\right) + \widetilde{\sigma}_n\left(\vec{x}, A_n(\vec{x}), \beta\right) Z\right]\right] - \max\left[\mu_n\left(A_n(\vec{x})\right)\right]. \tag{3.13}$$

To compute (3.13), we consider three cases: when $A_n(\vec{x})$ has one, two, or more than two elements. This third case is the most common in the allocation rules developed in Section 3.4.

When $A_n(\vec{x})$ has exactly one element, one can show using the tower property of conditional expectation that $V_n\left(\vec{x}, A_n(\vec{x}), \beta\right) = 0$. In other words, if only one alternative can ever be selected, information has no value.

When $A_n(\vec{x})$ has exactly two elements, computation of $V_n\left(\vec{x}, A_n(\vec{x}), \beta\right)$ is similar to related computations for the VOI in a pairwise comparison (Frazier et al. 2008, Jones et al. 1998, Chick and Inoue 2001a). Namely, let Δ be the absolute value of the difference of $\mu_n(x)$ between the two different $x \in A_n(\vec{x})$, and let s be the absolute value of the

difference of the two components of $\widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)$. Then

$$V_n(\vec{x}, A_n(\vec{x}), \beta) = sf(-\Delta/s),$$

where $f(-z) = \varphi(z) - z\Phi(-z)$, and φ and Φ are the density and cumulative distribution functions, respectively, of a standard normal random variable.

When $A_n(\vec{x})$ contains more than two elements, computation of $V_n(\vec{x}, A_n(\vec{x}), \beta)$ is more involved, but still can be performed analytically. Recalling (3.13), we see that we can write

$$V_n(\vec{x}, A_n(\vec{x}), \beta) = h\left(\mu_n(A_n(\vec{x})), \widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)\right), \tag{3.14}$$

where $h(a,b) = \mathbb{E}[\max_i a(i) + b(i)Z] - \max_i a(i)$ for two vectors a and b of equal length. Frazier et al. (2008) gives an exact algorithm for computing h and Frazier (2013) provides a Matlab implementation. More details are given below in Section 3.6.3.

In situations where some entries in the sampling covariance Λ are negative, independent sampling for the pairs of alternatives corresponding to these entries is preferred over correlated sampling. More generally, the VOI increases as the sampling correlation increases. This is shown by the following lemma, and is used in our sampling algorithm to improve performance.

Lemma 1. *Suppose* $\vec{x} = \left(x^{(1)}, x^{(2)}\right)$. *Let* μ_n, Σ_n, $\Lambda\left(x^{(1)}, x^{(1)}\right)$, $\Lambda\left(x^{(2)}, x^{(2)}\right)$ *be fixed. Then for any A and* β, $V_n(\vec{x}, A, \beta)$ *is an increasing function of the sampling correlation between* $x^{(1)}$ *and* $x^{(2)}$,

$$\rho\left(x^{(1)}, x^{(2)}\right) = \frac{\Lambda\left(x^{(1)}, x^{(2)}\right)}{\Lambda\left(x^{(1)}, x^{(1)}\right)\Lambda\left(x^{(2)}, x^{(2)}\right)}.$$

3.3.3 Knowledge Gradient Factors

The knowledge-gradient (KG) factor is a metric that measures the VOI per sample, when a given alternative $\vec{x}$ is sampled β times before an implementation decision. Qualitatively, it is a rate of information per sample. The allocation rules in Section 3.4 will make use of the KG factor when making a sampling decision at each stage of sampling. The KG factor uses the predictive distribution in (3.10) and the computational cost $c(\vec{x})$ of sampling at $\vec{x}$, measured by the computation time required.

Thus, the KG_β factor at time n for observing the value at a given singleton $x \in \{1,2,\ldots,k\}$ is

$$\nu_n^{KG_\beta}(x) = V_n\left(x, A_n(x), \beta_n\right) \big/ [\beta_n c(x)], \tag{3.15}$$

where β_n and $A_n(\cdot)$ may be chosen in an implementation-specific way (see Section 3.6.3). Similarly, the KG_β factor at time n for observing the difference in value between a pair of alternatives $\left(x^{(1)}, x^{(2)}\right)$ is

$$\nu_n^{KG_\beta}\left(x^{(1)}, x^{(2)}\right) = V_n\left(\left(x^{(1)}, x^{(2)}\right), A_n\left(x^{(1)}, x^{(2)}\right), \beta_n\right) \Big/ \left[\beta_n c\left(\left(x^{(1)}, x^{(2)}\right)\right)\right]. \tag{3.16}$$

If the computation time for a sample does not depend on $\vec{x}$, then $c(\vec{x}) = c|\vec{x}|$, where c is a positive constant cost per sample, and $|\vec{x}|$ is the length of $\vec{x}$. We adopt this model in numerical tests below.

3.4 Allocation Rules

This section discusses allocation rules, which use previous sampling information to decide how to take the next sample or samples, and which appear in Step 3 of the generic

sampling algorithm in Section 3.2. The allocation rules discussed all search over a set of possible sampling decisions to find the one with the largest KG_β factor, but differ in the way in which this search is performed.

Let $\Xi = \{1,2,\ldots,k\} \cup \left\{ \left(x^{(1)},x^{(2)}\right) \in \{1,2,\ldots,k\}^2 : x^{(1)} \neq x^{(2)} \right\}$ denote the set of all singletons and pairs. For each allocation rule below, we let $\Xi_n \subseteq \Xi$ denote a possibly smaller set, and at each iteration n, the allocation rule selects the sampling decision that maximizes the KG_β factor from Section 3.3.3 over this set,

$$\vec{x}_n = \underset{\vec{x}\in\Xi_n}{\operatorname{argmax}} \nu_n^{\mathrm{KG}_\beta}(\vec{x}). \tag{3.17}$$

Certain ways of choosing the Ξ_n will be shown to improve the computation time of the algorithm while retaining theoretical convergence guarantees (in Section 3.5) and good empirical performance (in Section 3.7).

When calculating the KG_β factor $\nu_n^{\mathrm{KG}_\beta}(\vec{x})$, we replace strictly negative entries in the sampling covariance matrix Λ by 0, because Lemma 1 shows that this generates a larger VOI and corresponding KG_β factor. Then, if a pair of alternatives whose sampling covariance was replaced by 0 is selected for simulation by our allocation rule, we use independent sampling rather than CRN to simulate these alternatives. Otherwise, we use CRN when sampling pairs.

The expression (3.17) depends upon the choice for Ξ_n, and implicitly on the choice of β_n and $A_n(\vec{x})$ used to calculate $\nu_n^{\mathrm{KG}_\beta}(\vec{x})$. Thus, different allocation rules are specified by different methods for choosing Ξ_n, β_n, and $A_n(\vec{x})$. We define a class of allocation rules, called **KG_β^2 allocation rules**, to be any that includes at least one singleton and one pair of alternatives in Ξ_n, and includes both $x^{(1)}$ and $x^{(2)}$ in $A_n(\vec{x})$, if $\vec{x} = \left(x^{(1)},x^{(2)}\right)$, for each n. Within this larger class, we now define two more specific types of KG_β^2 allocation rules, which place additional conditions on Ξ_n.

An **idealized** KG_β^2 allocation rule (proposed in Frazier et al. 2011) is one in which $\Xi_n = \Xi$ for each n. Thus, an idealized KG_β^2 allocation rule looks over all the singleton and pairwise-difference KG_β factors and finds the largest one. A specific instance of an idealized KG_β^2 allocation rule would require specifying a choice for β_n and $A_n(\vec{x})$.

When k is large, the exhaustive maximization performed by an idealized KG_β^2 rule is too computationally intensive. Frazier et al. (2011) proposed an alternative to this exhaustive maximization, which checks only singletons and a subset of pairs of alternatives, but even that approach is too computationally intensive when $k \gg 10^3$ and is not easily amenable to theoretical analysis.

To allow for better performance in large problems in a way that also supports theoretical analysis, we propose here a new class of KG_β^2 allocation rules, called **accelerated** KG_β^2 allocation rules, which can be used when the alternatives are embedded in an integer lattice, or some other space that supports local search. An accelerated KG_β^2 allocation rule is one that chooses at least one singleton from $\{1,2,\ldots,k\}$ and at least one pair from $\left\{\left(x^{(1)},x^{(2)}\right) \in \{1,2,\ldots,k\}^2 : x^{(1)} \neq x^{(2)}\right\}$, adding these to Ξ_n. Then, starting at each chosen singleton or pair $\vec{x}$, it applies a function $\tilde{f}$, which we call a "local search function", to produce a point $\tilde{f}(\vec{x})$, and adds this point to Ξ_n as well. The singleton or pair in Ξ_n with the best KG factor is then selected for evaluation, according to (3.17). The function $\tilde{f}$ can be defined in an implementation specific way, but would usually be designed to find a local optimum of the KG factor in the neighborhood of the passed input sampling decision.

Thus, an accelerated KG_β^2 allocation rule is specified by a rule for choosing the starting singletons and pairs, and for β_n, $A_n(\vec{x})$, and $\tilde{f}$. One choice for $\tilde{f}$, implemented using a gradient-based local search appropriate for alternatives corresponding to an integer lattice, is provided in Section 3.6.3. Another choice, the identity map, $\tilde{f}(\vec{x}) = \vec{x}$, results

in a form of random search.

The class of **KG_β allocation rules** is defined analogously to the class of KG^2_β allocation rules, except that only singletons (not pairs) may be sampled. That is, $\Xi_n \subseteq \{1,2,\ldots,k\}$ in (3.17) for KG_β allocation rules. The notions of idealized and accelerated KG_β allocation rules are defined as for the KG^2_β allocation rules above, except that pairs are not included in the search. When $\Xi_n = \{1,\ldots,k\}$, $A_n(\vec{x}) = \{1,\ldots,k\}$, and $\beta_n = 1$, we recover the allocation rule proposed in Frazier et al. (2009).

3.5 Convergence Properties

This section shows that the generic sampling algorithm from Section 3.2, when used with known Λ, μ_0, and Σ_0, and with a KG^2_β allocation rule from Section 3.4 satisfying mild conditions, samples every alternative infinitely often, so that we learn the value of every alternative, and are able to find a global maximum $x^* \in \text{argmax}_x\, \theta(x)$ almost surely in the limit as the number of samples grows without bound. Frazier et al. (2009) proved these consistency results for the idealized KG_β algorithm with $A_n(\vec{x}) = \{1,\ldots,k\}$ and $\beta_n = 1$, and so the results here can be viewed as a generalization to KG^2_β and to algorithms that do not require exhaustive optimization over all alternatives. The presence of sampling correlations, however, require substantially different proof techniques from those used in Frazier et al. (2009).

These results depend on two assumptions and a condition, which are stated precisely below. The first assumption states that we require the parameters governing Λ, μ_0, and Σ_0 to be known and fixed. The second assumption states that there is genuine uncertainty about each alternative's performance. The condition restricts the choice of KG^2_β allocation rule, and is satisfied by the idealized KG^2_β and accelerated KG^2_β

allocation rules from Section 3.4 as long as every $\vec{x} \in \Xi$ is chosen as a starting point for the local search infinitely often, with probability one.

Assumption 1. *μ_0, Σ_0 and Λ are known.*

Assumption 2. *Σ_0 and Λ are positive definite.*

Condition 4. *Each $\vec{x} \in \Xi$ is included in Ξ_n infinitely often, with probability* 1.

We now state our main result: that we become certain of the vector of true means θ eventually, as the conditional variance $\Sigma_n(x,x)$ of $\theta(x)$ converges to 0, and the conditional mean $\mu_n(x)$ converges to $\theta(x)$, for each x; and that the implementation decision that would be chosen if sampling stopped at time n, $\text{argmax}_x \mu_n(x)$, is eventually globally optimal. The proof may be found in Appendix B.1.

Theorem 1. *If Assumptions 1 and 2 hold, and if sampling occurs according to a* KG^2_β *allocation rule satisfying Condition 4, then:* $\lim_{n\to\infty} \Sigma_n(x,x) = 0$ *almost surely for each x;* $\lim_{n\to\infty} \mu_n(x) = \theta(x)$ *almost surely and in* L^2 *for each x; and* $\lim_{n\to\infty} \text{argmax}_x \mu_n(x) = \text{argmax}_x \theta(x)$ *almost surely.*

3.6 Implementation Features and Practicalities

This section discusses practical implementation choices arising in the generic algorithm in Section 3.2 and allocation rules in Section 3.4. This includes the specification of the functional form of the prior distribution and structure of the initial stage of sampling in Step 1, the empirical Bayes estimator used to assess μ_0, Σ_0 and Λ in Step 2, the choice of $A_n(\vec{x})$ and β_n used in KG^2_β allocation rules, and derivations of the gradients of the VOI and KG factors used by accelerated KG^2_β allocation rules. The convergence results

in Section 3.5 do not depend on how these implementation issues are addressed, as long as Assumptions 1, 2 and Condition 4 are valid.

Several of the implementation choices discussed assume that the k alternatives may be represented as elements in a lattice in $\mathbb{Z}^d$. For example, in a manufacturing problem, there may be d decision variables, each of which represents the number of resources (machines, employees with given skill sets, etc.) that combine to define a specific alternative manufacturing system design. That is, for any alternative x, we can specify its grid coordinates $\{\zeta_i(x)\}_{i=1}^d$.

3.6.1 Functional Form of the Prior Distribution and Sampling Covariance (Step 1 of Generic Sampling Algorithm)

Step 1 of the generic sampling algorithm requires specification of the functional form of the sampling covariance and prior distribution for the unknown means, either fully, or more frequently in terms of parameters to be estimated later in Step 2. We discuss this choice here.

The functional form of the sampling covariance Λ is considered first. While several different forms are possible, we assume compound sphericity for simplicity. The compound sphericity assumption means that Λ can be specified with exactly two parameters: a common sampling variance σ_ε^2 on the diagonals and a common sampling correlation across any pair of alternatives, ρ. All off-diagonal elements of Λ are the same. While the compound sphericity assumption is strong, it has been used by others to model the effect of CRN (Schruben and Margolin 1978, Tew and Wilson 1992), including in the context of CRN with kriging (Chen et al. 2012).

We now discuss the functional form of the prior distribution for the unknown means. When the alternatives may be embedded in a lattice, there may be a belief that the performance of two alternatives that are 'near' each other in this lattice are more likely to be similar than the performance of two alternatives that are 'distant' from each other. This motivates the notion that the prior distribution may be a multivariate normal distribution under which the covariance between the values of any two alternatives is a decreasing function of their distance from each other on the lattice. This is analogous to covariance functions used in GP priors over continuous functions. Inspired by this link to GP priors, we adopt the commonly used Gaussian kernel.

$$\Sigma_0\left(x,x'\right) = \sigma_0^2 \exp\left\{-\sum_{i=1}^{d} \alpha_i \left[\zeta_i(x) - \zeta_i(x')\right]^2\right\}. \tag{3.18}$$

Here σ_0^2 is the homogeneous prior variance of the unknown means and $\vec{\alpha} = \{\alpha_i\}_1^d$ is a vector of scaling parameters. We also let η be a parameter for the mean in this model and let $\vec{1}$ be a vector of k ones, so that (3.18) and $\mu_0 = \eta\vec{1}$ define the prior distribution in (3.1).

Specification of the prior distribution parameters μ_0, Σ_0 can therefore be accomplished by specifying σ_0^2, $\vec{\alpha}$, and η. Kernels other than that in (4.11) would be handled similarly.

3.6.2 Initial Stage of Sampling (Step 1 of Generic Sampling Algorithm) and Empirical Bayes Parameter Update (Step 2 of Generic Sampling Algorithm)

Here we discuss the initial stage of sampling performed in Step 1, and the periodic empirical Bayes updates performed in Step 2 of the generic sampling algorithm. These

steps are used when Λ, μ_0 or Σ_0 or some parameters of their functional forms are unknown, and require some estimation.

If an initial stage of sampling is required, we randomly select a set $\vec{x}_{01}$ of N_1 alternatives, sample once from each of them using CRN, sort them in descending order, and then take another sample from each of the first N_2 alternatives, denoted by the vector $\vec{x}_{02}$, using CRN $(N_2 < N_1)$. We initialize the number of stages sampled so far to be $n = 2$ (one for each use of CRN), $\mathscr{X}_2$ to be the row concatenation of $\vec{x}_{01}$ and $\vec{x}_{02}$, and $\mathscr{Y}_{\in}$ to be the outputs at those alternatives.

Once this initialization stage of samples is complete, and also periodically thereafter according to a fixed schedule, we estimate the parameters determining μ_0, Σ_0, and Λ in Step 2 of the generic sampling algorithm using a maximum likelihood estimator (MLE). Appendix B.2 derives a MLE assuming that μ_0, Σ_0, and Λ take the functional form specified in Section 3.6.1, which has parameters σ_0^2, $\vec{\alpha}$, η, σ_ε^2 and ρ. This use of maximum likelihood estimation to estimate parameters within a Bayesian model is known as an empirical Bayes approach, and is common in GP regression. Relaxing the compound sphericity assumption or using a different GP prior in our proposed algorithm simply involves providing an alternative MLE for Λ, μ_0 and Σ_0.

We let N_3 denote the set of times at which the MLE will be performed, so N_3 contains $N_1 + N_2$. If computation time for the allocation rule is unimportant (e.g., because the simulations themselves are very time-consuming), one may perform the MLE before each new stage of sampling, in which case $N_3 = \{N_1+N_2, N_1+N_2+1, N_1+N_2+2, \ldots\}$. In other situations, because computation of the MLE may be time-consuming, it may be beneficial to avoid recomputing the MLE at every stage. In our implementation, we update the MLE more frequently at first when additional samples tend to have more impact on parameter estimates, and then less frequently as more samples are acquired.

If the parameters are known, we may skip these updates by setting $N_3 = \emptyset$.

3.6.3 Local Search Function and Other Implementation Choices in KG Allocation Rules (Step 3 of Generic Sampling Algorithm)

This section discusses implementation-specific choices for $A_n(\vec{x})$ and β_n in idealized and accelerated KG_β and KG^2_β allocation rules. Additionally, for accelerated allocation rules, it discusses the choice of Ξ_n and the local search function $\tilde{f}$.

Except where otherwise noted in our numerical experiments, we set $\beta_n = 1$ and we chose $A_n(\vec{x})$ to be the alternatives in $\vec{x}$ and the best other sampled alternative given the observations available. So, for singletons $\vec{x} = (x)$, we set $A_n(x) = \{x, x_*\}$, where $x_* = \mathrm{argmax}_{x' \in \mathscr{X}_n \setminus \{x\}} \mu_n(x')$. For pairs, $\vec{x} = (x^{(1)}, x^{(2)})$, we set $A_n(\vec{x}) = \left\{x^{(1)}, x^{(2)}, x_*\right\}$, where $x_* = \mathrm{argmax}_{x' \in \mathscr{X}_n \setminus \{x^{(1)}, x^{(2)}\}} \mu_n(x')$.

We now describe the choice of Ξ_n used within accelerated KG^2_β and KG_β allocation rules in our numerical experiments. Denote the best and second best alternative (in terms of posterior mean) after n samples as

$$x_{n,b} = \operatorname*{argmax}_{x \in \mathscr{X}_n} \mu_n(x), \qquad x_{n,s} = \operatorname*{argmax}_{x \in \mathscr{X}_n \setminus \{x_{n,b}\}} \mu_n(x).$$

In accelerated KG_β allocation rules, eligible sampling decisions Ξ_n were $x_{n,b}$, $x_{n,s}$, a randomly chosen singleton, and the values of $\tilde{f}$ applied to those three sampling decisions. In accelerated KG^2_β allocation rules, eligible sampling decisions Ξ_n were $x_{n,b}$, a random singleton, $\left(x_{n,b}, x_{n,s}\right)$, a random pair of alternatives, and the values of $\tilde{f}$ applied to those five sampling decisions.

In first stages of sampling, $\mathscr{X}_n$ may have too few elements for x_*, $x_{n,b}$ or $x_{n,s}$ to be defined. In such a case, a random sampling decision is used instead.

We now describe the local search function $\tilde{f}$ used within accelerated KG^2_β and KG_β allocation rules. This local search function assumes that the alternatives correspond to points on a grid embedded in a continuous space, as discussed in the beginning of Section 3.6, and also assumes that the prior is of the form specified in Section 3.6.1. This structure allows us to determine the gradient of the KG factors, and to use the gradient to locally optimize the KG factor in a neighborhood of $\vec{x}$, where $\vec{x}$ is interpreted as varying continuously. We round that local optimum to the nearest feasible grid point to obtain $\tilde{f}(\vec{x})$.

We first derive the gradient of the VOI, as it is required to determine the gradient of the KG factor. Specifically, we assess the gradient of $V_n(x, A_n(x), \beta)$ in $\mathbb{R}^d$ and of $V_n\left((x^{(1)}, x^{(2)}), A_n\left(x^{(1)}, x^{(2)}\right), \beta\right)$ in $\mathbb{R}^{2d}$, where $A_n(\vec{x})$ is as described above.

We abuse notation slightly by writing the gradient of $V_n(x, A_n(x), \beta)$ in $\mathbb{R}^d$ in terms of derivatives with respect to the d coordinates of x rather than with respect to the $\zeta_i(x)$, in order to simplify notation. Similarly, for pairs $\vec{x}$, we write the gradient of $V_n(\vec{x}, A_n(\vec{x}), \beta)$ in $\mathbb{R}^{2d}$ by referring directly to the alternatives $\vec{x}$ rather than indirectly through the function ζ_i that embeds them in the grid.

First consider the case of the singleton $\vec{x} = x$. Recall that $V_n(x, A_n(x), \beta) = sf(-\Delta/s)$, where $\Delta = |\mu_n(x) - \mu_n(x_*)|, s = |\widetilde{\sigma}_n(x, x, \beta) - \widetilde{\sigma}_n(x, x_*, \beta)|$, and $f(z) = \varphi(z) + z\Phi(z)$. Direct calculation then reveals that

$$\nabla_x [V_n(x, A_n(x), \beta)] = -\Phi(-\Delta/s) \cdot \operatorname{sign}[\mu_n(x) - \mu_n(x_*)] \cdot \nabla_x [\mu_n(x) - \mu_n(x_*)] + \varphi(\Delta/s) \cdot \operatorname{sign}[\widetilde{\sigma}_n(x, x, \beta) - \widetilde{\sigma}_n(x, x_*, \beta)] \cdot \nabla_x [\widetilde{\sigma}_n(x, x, \beta) - \widetilde{\sigma}_n(x, x_*, \beta)]. \tag{3.19}$$

Detailed derivations of $\nabla_x [\mu_n(x')]$ and $\nabla_x [\widetilde{\sigma}_n(x, x', \beta)]$ for arbitrary x' are given in Appendix B.3.

Second, consider the case of the pair $\vec{x} = \left(x^{(1)}, x^{(2)}\right)$. Letting $a = \mu_n(A_n(\vec{x}))$ and

$b = \widetilde{\sigma}(\vec{x}, A_n(\vec{x}), \beta)$ we have from Section 3.3.2 that

$$V_n(\vec{x}, A_n(\vec{x}), \beta) = h(a,b).$$

To support taking the derivative of this quantity, we now recall Algorithms 1 and 2 from Frazier et al. (2009) for computing $h(a,b) = E[\max_i a(i) + b(i)Z] - \max_i a(i)$. We first reorder the components of a and b so that the $b(i)$ are in non-decreasing order and ties in b are broken so that $a(i) \leq a(i+1)$ if $b(i) = b(i+1)$. Then, we remove all those entries i for which $a(i) + b(i)z < \max_{j \neq i} a(j) + b(j)z$ for all values of z (this is accomplished by Algorithm 1 in Frazier et al. (2009)). This gives new vectors a' and b' with $|a'| = |b'| \leq |a| = |b|$. Set $\gamma(i) = \frac{a'(i+1) - a'(i)}{b'(i+1) - b'(i)}$ for $i = 1, 2, \ldots, |a'| - 1$. Then

$$V_n(\vec{x}, A_n(\vec{x}), \beta) = h(a,b) = \sum_{i=1}^{|a'|-1} \left[b'(i+1) - b'(i) \right] f(-|\gamma(i)|)$$

if $|a'| > 1$ and the sum is taken to be 0 if $|a'| = 1$. Computation then reveals that

$$\begin{aligned} \nabla_{\vec{x}} [V_n(\vec{x}, A_n(\vec{x}), \beta)] = & \sum_{i=1}^{|a'|-1} \varphi(\gamma(i)) \nabla_{\vec{x}} [b'(i+1) - b'(i)] \\ & - \Phi(-|\gamma(i)|) \operatorname{sign}[a'(i+1) - a'(i)] \nabla_{\vec{x}} [a'(i+1) - a'(i)]. \end{aligned} \tag{3.20}$$

For each i, $a'(i)$ and $b'(i)$ are equal to $a(j)$ and $b(j)$ for j given by the reordering procedure above, and $a(j)$ and $b(j)$ are the jth components of $a = \mu_n(A_n(\vec{x}))$ and $b = \widetilde{\sigma}(\vec{x}, A_n(\vec{x}), \beta)$ respectively. Thus, $\nabla_{\vec{x}}[a'(i)]$ and $\nabla_{\vec{x}}[b'(i)]$ are equal to $\nabla_{\vec{x}}[\mu_n(x')]$ and $\nabla_{\vec{x}}[\widetilde{\sigma}_n(\vec{x}, x', \beta)]$, where x' is the jth element in $A_n(\vec{x})$. Derivations of these quantities are given in Appendix B.3.

We now consider the gradient of the KG_β factors in $\mathbb{R}^d$. Recalling (3.15) and (3.16), we have

$$\nabla_{\vec{x}} \left[\nu_n^{KG_\beta}(\vec{x}) \right] = \left(\nabla_{\vec{x}} [V_n(\vec{x}, A_n(\vec{x}), \beta)] \cdot c(\vec{x}) - V_n(\vec{x}, A_n(\vec{x}), \beta) \cdot \nabla_{\vec{x}} [c(\vec{x})] \right) / \left(\beta [c(\vec{x})]^2 \right) \tag{3.21}$$

for $\vec{x} = x$ or $\left(x^{(1)}, x^{(2)}\right)$. In the case of homogeneous sampling costs for each alternative ($c(\vec{x}) = c|\vec{x}|$), we have $\nabla_{\vec{x}}[c(\vec{x})] = 0$. Hence (3.21) is determined by preceding results as

$$\nabla_{\vec{x}}\left[\nu_n^{KG_\beta}(\vec{x})\right] = \nabla_{\vec{x}}[V_n(\vec{x}, A_n(\vec{x}), \beta)] / (\beta c(\vec{x})).$$

3.7 Numerical Results

Beyond asymptotic convergence to the optimal solution, we are interested in the rate in which solutions improve for even small numbers of samples. We measure this performance by the expected opportunity cost (the difference between the true best and the estimated best $x_{n,b}$, as defined in Section 3.6.3, at each time n), $\mathbb{E}\left[\max_x \theta_x - \theta_{x_{n,b}}\right]$.

In this section we present numerical results to explore the behavior of the proposed algorithm, allocations rules, and implementation choices from Section 3.6 in order to answer the following questions. Does pairwise sampling with CRN provide an efficiency benefit, even if approximations are made to simplify computations? How much benefit can the KG and KG^2 allocation rules give, on problems with combinatorially large numbers of solutions, as compared to other benchmark algorithms such as a random search which is enhanced with a Gaussian process metamodel (which we call RSGP and describe below) and Industrial Strength COMPASS (Xu et al. 2010).

Except as noted below, the KG and KG^2 allocation rules used the Gaussian process prior for unknown means, compound sphericity assumption for samples, MLE and empirical Bayes estimation, and other parameters as described in Section 3.6. When μ_0, Σ_0 and Λ were not known, the parameters for the initial stage of sampling were $N_1 = 10d, d \leq N_2 \leq 2d$, where d is the dimension of the problem, and we let N_3 contain $N_1 + N_2$ and stage numbers that allowed the period between updates to increase from

30 to 60 as sampling continued. In cases where sampling was done without CRN, we performed maximum likelihood estimation with ρ fixed to 0.

3.7.1 How do the approximations interact?

This section assesses the relative importance of several features and approximations described above: the allocation rule, approximations due to accelerated allocations and parameter estimation, and deviations from the assumed sampling correlation structure under CRN. Specifically, we assess the $12 = 2 \times 3 \times 2$ combinations that result from combining each level of the following three factors:

Allocation: KG_β allocation rule (no CRN); or KG^2_β allocation rule (CRN allowed).

Approximation: Idealized allocation rule with known parameters; accelerated allocation rule with known parameters; or accelerated allocation with unknown parameters $(\sigma_0^2, \vec{\alpha}, \eta, \sigma_\varepsilon^2, \rho)$ fit as in Section 3.6.2.

Sampling with CRN: Samples satisfy compound sphericity (with $\rho(i,j) = 0.25$ for $i \neq j$); or decreasing correlations (with $\rho(i,j) = \exp\left[-(i-j)^2/50\right]$ for $i \neq j$) even though compound sphericity may be (incorrectly) assumed by the parameter fitting.

We do so for randomly generated problem instances with a small (100) number of alternatives. We generate 500 problem instances. In each problem, the 100 alternatives had means distributed as a $\mathcal{N}(\mu_0, \Sigma_0)$ with $\mu_0 = \vec{0}$ and $\Sigma_0(i,j) = 100\exp\left[-(i-j)^2/50\right]$ for $i, j = 1, 2, \ldots, 100$. We assumed a homogeneous sampling variance $\sigma_\varepsilon^2 = 50$. We set $\beta = 1$ and so refer to KG_1 and KG^2_1.

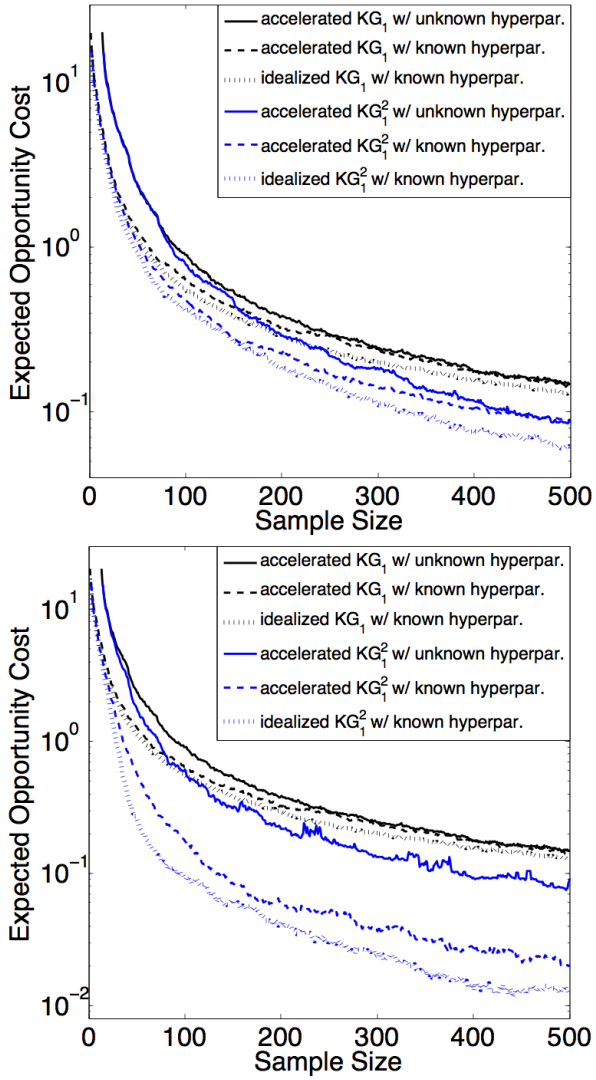

Figure 3.1: Performance of selected algorithms in the grid test problem with compound sphericity (top plot) and decreasing correlations (bottom plot).

Figure 3.1 shows the expected opportunity cost of a potentially incorrect selection, on a logarithmic scale, as a function of the total number of samples. The maximum size of the 95% confidence intervals is 0.28 at sample size 100, 0.09 at sample size 300, and 0.05 at sample size 500.

Not surprisingly, idealized KG allocation rules performed better than their accelerated counterparts (the idealized does an exhaustive, not local, search to maximize the

KG factor). The degree of sub-optimality was not particularly great in any setting where parameters were known. A greater degree of sub-optimality was seen when parameter estimation was used. The deterioration due to parameter estimation was not significant for the KG_1 allocation, even when sphericity did not apply and parameters were (incorrectly) estimated with the sphericity assumption (right panel, top three lines). The degradation in performance due to parameter estimation with the KG_1^2 allocation was not too significant when sphericity was correctly assumed (left panel, bottom three curves).

All else fixed, a KG^2 allocation with CRN improved upon the performance of its corresponding KG allocation with independent sampling. Thus, the ability of sampling pairs with CRN offered an important benefit beyond sampling only one alternative independently at a time (both panels).

Moreover, we observed that the accelerated KG_1^2 allocation rule, even when parameter estimation was used, performed better than the idealized KG_1 allocation, which had the advantage of 'knowing' the true sampling correlation and of doing an exhaustive search over KG factors. Thus, the benefit of CRN outweighed the penalties associated with sub-optimality in the accelerated KG_1^2 allocation rule with unknown parameters, once 200 samples were observed to get stable parameter estimates (even when sphericity was incorrectly assumed by the MLE, right panel).

In experiments not shown here for reasons of space, we found other interesting observations. One, when we set $\rho(i,j) = 0.5$ rather than $\rho(i,j) = 0.25$ for all $i \neq j$, the expected opportunity costs decreased. This is consistent with the benefit offered by sampling pairs being increasing in a (common) sampling correlation ρ. Two, we experimented with the number of randomly selected singletons and pairs that were included in Ξ_n for the accelerated allocations. Increasing that number to 2 or 3 provided a practical improvement in performance in the approximate KG and KG^2 allocations, but the

benefit of adding random points beyond 4 or 5 had little marginal increase. Three, experiments with several values of β_n for KG^2_β allocation rules did not reveal a large difference in performance due to the choice of β_n.

3.7.2 Comparison with RSGP on a Rosenbrock Problem with 10^6 Alternatives

This section explores the performance of the procedures when there are a very large number of alternatives. The problem considered is a discretized version of a 6-dimensional Rosenbrock function with 10^6 alternatives. Each alternative x corresponds to a point in the grid with coordinates $\zeta(x) = \{\zeta_i(x)\}_{i=1}^6 \in [-0.8, -0.5, \ldots, 1.9]^6$ and has value

$$\theta(x) = -\sum_{i=1}^{5} \left[100 \left[\zeta_i(x)^2 - \zeta_{i+1}(x) \right]^2 + \left[\zeta_i(x) - 1 \right]^2 \right].$$

The computation required for idealized KG allocation rules is not practical when there are such a large number of alternatives. This section assesses differences in performance between the accelerated KG_1 and accelerated KG^2_1 allocation rules, as well as a benchmark algorithm that we introduce, called RSGP. The RSGP samples uniformly at random and uses the Gaussian Process model and parameter estimation tools in Section 3.6 to estimate the performance for each alternative by its posterior mean when selecting the best alternative.

The sampling noise satisfies the compound sphericity assumption, with $\sigma^2_\varepsilon = 125$ and $\rho(i, j) = 0.4$ for all $i \neq j$. These values were assumed unknown in this test, and the empirical Bayes approach described in Section 3.6.2 was used to estimate the GP prior and sampling covariance.

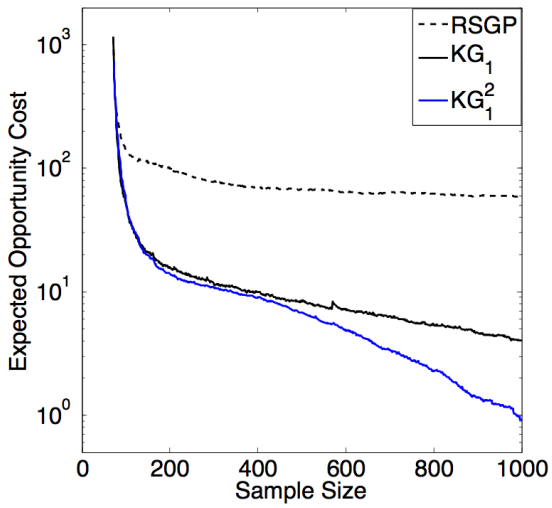

Figure 3.2: Expected opportunity cost for a benchmark algorithm, (Random Search with Gaussian Processes, RSGP), accelerated KG_1 (KG_1) and accelerated KG_1^2 (KG_1^2) allocation rules as a function of the total number of samples (on discrete Rosenbrock function).

Figure 3.2 shows the opportunity cost, averaged over 200 sample paths, of the accelerated KG_1 and accelerated KG_1^2 allocation rules, and the benchmark RSGP. Both KG allocation rules dramatically outperformed RSGP. This is because the KG factors steered sampling to areas that more efficiently identified local extrema. The KG_1 and KG_1^2 performed similarly through about 500 samples, but KG_1^2 provided better solutions thereafter. Exploring sample paths indicates that this was because both KG_1 and KG_1^2 initially identified regions of good local extrema, which occurred at about the same rate. Then, when good local extrema were found, the use of CRN helped KG_1^2 find better solutions more quickly, as compared to KG_1, near such local extrema.

3.7.3 Comparison with ISC on the Assemble to Order Problem

We now compare the accelerated KG_1 and KG_1^2 allocation rules with a well-known algorithm, Industrial Strength COMPASS (ISC, developed by Xu et al. 2010). We do so

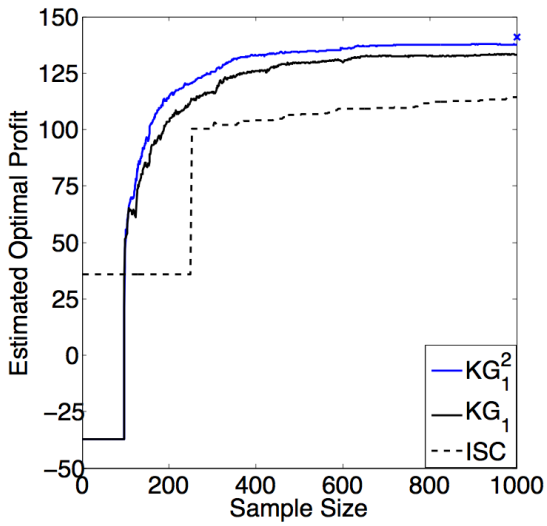

Figure 3.3: Performance of the accelerated KG_1^2 and accelerated KG_1 allocation rules, and Industrial Strength COMPASS (ISC) for the assemble-to-order (ATO) problem.

for the Assemble to Order (ATO) problem described in Hong et al. (2012 Accessed July 2013), which is a variation on the problem studied by Hong and Nelson (2006), and has a combinatorially large number (21^8) of alternatives.

In the ATO problem, orders for 5 different products arrive according to independent Poisson processes with constant arrival rates. Products are made up of a collection of items of 8 different types. Items are either key items or non-key items. If any of the key items are out of stock then the product order is lost. If all key items are in stock, then the order is assembled from all key items and the available non-key items. Each item sold brings a profit, and each item in inventory incurs a holding cost per unit time. There is an inventory capacity 20 for each item. Items are produced one at a time on dedicated machines. The production time for each item is normally distributed, truncated at 0. The system operates under a continuous-review base stock policy under which each item k has a target base stock b_k, and each demand for an item triggers a replenishment order for that item. Each simulation replication starts from a fully stocked system with no

orders in production, has a warm-up period of 20 time units, then captures statistics for the next 50 time units of operation. The goal is to maximize the expected total profit per unit time by selecting the target inventory level vector $b = (b_1, b_2, \ldots, b_8)$. See Hong et al. (2012 Accessed July 2013) for more details and code.

We calculate each algorithm's performance by collecting the true expected total profit (estimated in a post-processing step through exhaustive simulation) of the algorithm's current solution, as a function of the sample size. We then average this value over 100 independent sample paths for each algorithm. We fix the starting solution of KG_1 and KG_1^2 to the inventory capacity, and randomize the initial solution of ISC over the feasible set $\{b : 0 \leq b_k \leq 20, b_k \in \mathbb{Z}\}$. Thus, KG_1 and KG_1^2 were forced to start searching with a worse initial alternative to sample than did ISC, on average.

Figure 3.3 shows the average performance of the three algorithms. This average performance jumped when algorithms finished their initialization phases (Step 1 of the KG algorithms), which occurred at 250 samples for ISC and 95 samples for the two KG algorithms. The height of the x at the right edge of the plot (at $x = 1000$) gives the value (141, estimated through exhaustive simulation) of the best solution found by all sample paths across all three algorithms. This best solution was discovered by a KG algorithm. The true optimal solution is unknown. The accelerated KG_1^2 allocation rule outperformed the accelerated KG_1 allocation rule, which in turn outperformed ISC for this problem, in terms of achieving a higher quality solution with fewer samples. It is also important to consider the total amount of computation time required to reach a given solution quality. ISC required an average of 27 minutes of computation time to complete, taking 1084 samples on average. Its average profit upon completion was 115.53. To reach this same level of solution quality achieved by ISC, KG_1 took 279 samples on average and required 9 minutes of computation time, while KG_1^2 took 203 samples on

average and required 5.5 minutes of computation time. The two KG algorithms required fewer samples and less computation time than did ISC, with KG_1^2 delivering additional efficiency above and beyond that delivered by KG_1.

While the KG algorithms outperformed ISC in terms of total computation time to reach a given level of solution quality on the ATO problems, algorithms like KG_β and KG_β^2 that rely on kriging or Gaussian-process regression may consume substantial computational resources in deciding where to sample, which may make them less suitable for problems in which simulation can be performed very quickly. When simulation samples come from a complex, long-running simulator, this is relatively unimportant, and algorithms like KG_β^2 that find good solutions in few samples also work well in terms of overall computation time.

Figure 3.4 shows the time taken in a single sample path of the KG_β^2 algorithm. It shows that the CPU time per sampling decision increased over time, with a baseline level of computation due to gradient-based optimization of the KG factor, and spikes at regular intervals due to the empirical Bayes update of parameters. These spikes, which are so prominent in the right-hand panel of Figure 3.4, would also be present in any algorithm using kriging with adaptively updated parameter estimates. The increase with sample size in both the time to perform gradient-based optimization of the KG factor, and to perform empirical Bayes updates, was due to the increasing size of the matrices being manipulated for maximum likelihood estimation and for kriging-based prediction.

These points suggest potential future research directions: how to balance frequency of parameter updates to improve performance with the cost of computing them; how to speed up and improve parameter estimation; adaptation or development of localized submodels for kriging approximation to reduce the number of samples included in local gradient search to optimize KG factors; how much time to spend on the local search

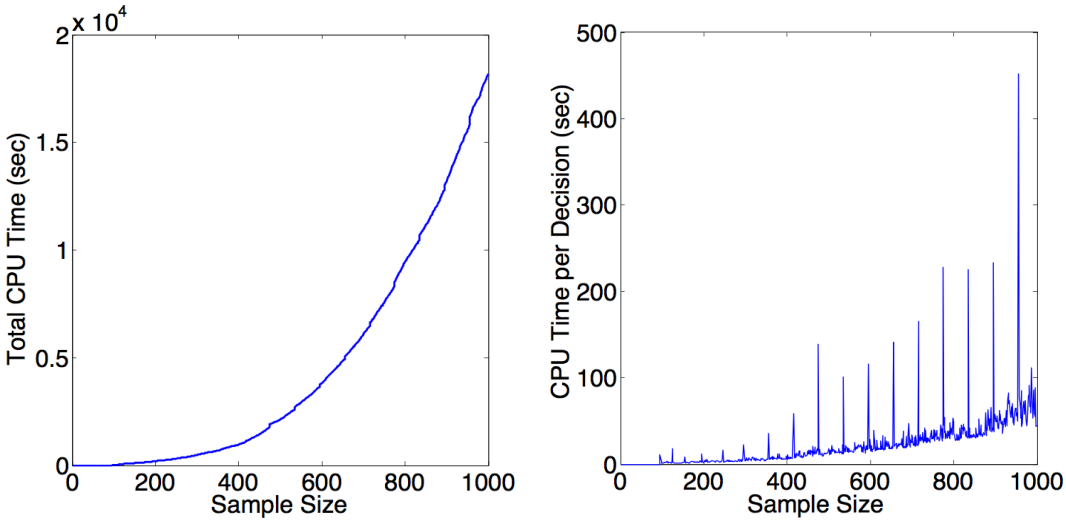

Figure 3.4: CPU time spent in a sample path of KG_1^2, as a function of the sample size, on the ATO problem.

(balancing some improvement versus perfect improvement in these steps). Related to this last point, we did derive and test second-order methods (not shown) to find local optimizers of the KG factors but found they did not give CPU cost per iteration benefits relative to Matlab's fminsearch and simple gradient search on some test problem.

In summary, our algorithms demonstrate superior efficiency compared to others in problems with large solution spaces and when samples are moderately to very computationally expensive.

3.8 Conclusions

We contributed to the area of discrete optimization via simulation, where the value of the best alternative is to be estimated by simulation, by developing a fully sequential algorithm based on new value of information tools. Those tools are able to take advantage of both correlated prior beliefs and correlated sampling distributions. We gave easy-to-

verify conditions under which almost sure convergence to the optimal solution can be guaranteed. The implementation presented here takes advantage of machine learning tools that enable exploring combinatorially large solution spaces, with run times that are a low order polynomial in the number of samples observed (which is much better than a low order polynomial in the size of the solution space). We also derived 'accelerated' versions of the algorithms that use local search when alternatives can be embedded in a continuous space. That acceleration takes advantage of gradient information about the Bayesian value of information, rather than the more common technique of using gradient information about the response surface, to improve practical performance. Numerical results show that there is a distinct benefit for being able to use both correlated prior beliefs and correlated sampling in simulation optimization using the Bayesian value of information framework.

CHAPTER 4

OPTIMIZATION OF COMPUTATIONALLY EXPENSIVE SIMULATIONS WITH GAUSSIAN PROCESSES AND PARAMETER UNCERTAINTY: APPLICATION TO CARDIOVASCULAR SURGERY

Motivated by an application in cardiovascular surgery with parameter uncertainty, we develop a new method for optimization of an objective function whose value is the average of the output of a computationally expensive simulator, where the input is varied across some low-dimensional space. We use Bayesian methods, in which inference based on a Gaussian process prior learns the behavior of the computationally expensive simulator across the input space, and tracks our uncertainty about values at unevaluated points. We then use value of information calculations to decide at which inputs it would be most valuable to evaluate the simulator next.

The application that we consider is the design of idealized bypass graft models under uncertain shape design variables, model geometry and boundary conditions, and unsteady flow, using a simulation of blood-flow in the graft. Our goal is to compute the optimal graft attachment angles that minimize the area of low wall-shear stress (WSS). Previously, non-Bayesian methods (surrogate management framework SMF) (Booker et al. 1999, Marsden et al. 2008) were coupled to cardiovascular simulations to perform robust shape optimization (Marsden et al. 2004, Sankaran and Marsden 2010, 2011). To account for uncertainties, a stochastic collocation method (Sankaran and Marsden 2011) was coupled with SMF (Sankaran et al. 2010). This method converges to a mesh local optima for Lipschitz continuous functions. We have previously demonstrated in Sankaran and Marsden (2010) that accounting for implementation and measurement uncertainties affects the optimal graft attachment angle.

In this work, we investigate the expected performance of the design variables under low-dimensional uncertainties. In defining our objective, we consider not only the areas of low WSS, which is an output of our simulation, but also the uncertainties associated with implementation, inflow velocity and stenosis radius. The random output variable is thus a deterministic function of a low-dimensional random vector. Evaluation of this deterministic function is expensive, and its derivative information is unavailable. Our goal is to optimize the expectation of this output variable (or its variant) by allocating simulation effort efficiently across different values of the random vector.

These types of problems arise in many applications of simulation optimization. e.g., the robust optimization of the design of biomechanical devices (Chang et al. 1999, Santner et al. 2001), where they incorporate environmental variables such as multiple loading conditions, and employ an empirical best linear unbiased prediction of the structural response.

In its attempt to evaluate the expectation (integral) of an implicit function, this work is closely related to the Bayesian Quadrature (O'Hagan 1991) or the Bayesian Monte Carlo method (Rasmussen and Ghahramani 2003), which models the integrand using Gaussian process (GP) (Rasmussen and Williams 2006), and then performs inference about the integral by taking advantage of the analytical convenience of the GP models.

To design a strategy that samples efficiently, we employ a Bayesian approach, in which we begin with a GP prior distribution on the response function, updating this prior distribution based on sampling information, evaluate the expectation of the response function under uncertain model inputs and unsteady flow, and use "value of information" computations to decide how to best allocate sampling effort.

4.1 Problem Formulation

In this section we formulate the Bayesian shape optimization problem of an idealized bypass graft model with unsteady flow. This problem is studied in Sankaran and Marsden (2010), which uses the stochastic collocation technique to incorporate and study the effects of input uncertainties, and applies a derivative-free SMF optimization method to perform robust shape design.

In this problem, the design variables are the target anastomosis angles x_1 and x_2 given to the surgeon. Given these target values, the actual angles of a bypass graft constructed in a surgery are not x_1 and x_2, but instead $\theta_1 = x_1 + \delta_1$ and $\theta_2 = x_2 + \delta_2$, where δ_1 and δ_2 are the implementation errors introduced during surgery. As shown in Figure 4.1, we denote by r and v the stenosis radius and the inflow velocity respectively. We then write $x = (x_1, x_2)$, $\delta = (\delta_1, \delta_2)$, $\theta = x + \delta$ and $\omega = (r, v)$.

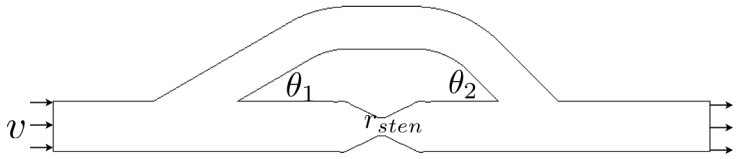

Figure 4.1: Schematic of the bypass graft surgery with the two attachment angles, inlet velocity and stenosis radius shown.

We assume that the area of low WSS is fully determined by the actual anastomosis angles θ, the stenosis radius r, and the inflow velocity v. Given θ and $\omega = (r, v)$, we denote by $f(\theta, \omega)$ the corresponding area of low WSS. We can use simulation to evaluate $f(\theta, \omega)$ exactly. However, each evaluation is time-consuming, requiring several hours of parallel computation, limiting how many times we may perform this evaluation.

To optionally include risk aversion into our objective function, we define a utility

function U by

$$U(\theta,\omega) = -f(\theta,\omega) \quad \text{or} \quad U(\theta,\omega) = e^{-\alpha\cdot f(\theta,\omega)},$$

where $\alpha > 0$ is a parameter that models aversion to risk, with larger values of α corresponding to more aversion to risk. The second definition can be used to control the standard deviation (sensitivity) of f due to input uncertainties.

For analytical convenience, we suppose that our probability distributions over ω and δ are independent and normal (ω may be truncated at 0). Denote by $p(\delta,\omega)$ their joint pdf, which is assumed known.

Our overarching goal is to find the target anastomosis angles x that maximize the expected value of $U(\cdot,\cdot)$, i.e., we want to solve

$$\max_x g(x), \tag{4.1}$$

where

$$g(x) := \iint U(x+\delta,\omega)\, p(\delta,\omega)\, \mathrm{d}\delta\, \mathrm{d}\omega \tag{4.2}$$

is the expected utility that results from using target values x.

4.2 Statistical Inference and Value of Information Analysis

To support the solution to the optimization problem, we use Bayesian statistics to provide an estimate of $U(\theta,\omega)$ across *all* points (θ,ω), based on those points at which U has actually been evaluated. This statistical framework also provides uncertainties associated with these estimates. This is useful because evaluating U is time-consuming, and so we cannot simply evaluate it at each point of interest. Applying Bayesian Quadrature techniques, these estimates of U, and their associated uncertainties, then imply estimates

and uncertainties of $g(x)$ across the domain of x. In this section, we first describe the statistical framework in which this estimation takes place. We then describe a value of information analysis based upon this statistical framework, in which we quantify the value of evaluating U at a given set of previously unevaluated values. This quantification of the value of information will then be used later in Section 4.3 to create an algorithm for solving (4.1).

We work in a Bayesian framework, in which we place a Gaussian process (GP) prior distribution over the function U. For an overview of GP priors see Rasmussen and Williams (2006).

$$U(\cdot,\cdot) \sim \mathrm{GP}\left(\mu_0(\cdot,\cdot), \Sigma_0(\cdot,\cdot,\cdot,\cdot)\right),$$

where

$$\mu_0 \colon (\theta, \omega) \mapsto \mathbb{R},$$
$$\Sigma_0 \colon \left(\theta, \omega, \theta', \omega'\right) \mapsto \mathbb{R},$$

and Σ_0 is a positive semi-definite function. A typical choice of Σ_0 is the square exponential covariance function (see Section 4.4.1). At each time $n = 1, 2, \ldots$, our algorithm will evaluate some point (θ_n, ω_n), and observe the resulting objective, $y_n = U(\theta_n, \omega_n)$. Define $D_n = \{\theta_{1:n}, \omega_{1:n}, y_{1:n}\}$ to contain all of this data. The posterior distribution of U at time n is then

$$U(\cdot,\cdot) \mid D_n \sim \mathrm{GP}\left(\mu_n(\cdot,\cdot), \Sigma_n(\cdot,\cdot,\cdot,\cdot)\right),$$

where μ_n and Σ_n can be computed using standard results from Bayesian linear regression (see, e.g., Rasmussen and Williams (2006) or Gelman et al. (2004)). Section 4.4.2 gives explicit expressions for μ_n and Σ_n.

Denote by $\mathbb{E}_n$ and Cov_n the expectation and covariance conditioned on D_n, respectively. That is, $\mathbb{E}_n$ and Cov_n are the expectation and covariance under the posterior

distribution at time n. We then relate the posterior distribution on U to the posterior distribution on the function g via Bayesian Quadrature. First, the posterior mean of the function g at an arbitrary point x can be calculated by interchanging integration over the values of $g(x)$ with the integration defining $g(x)$ in (4.2) via Fubini's theorem to obtain,

$$\mathbb{E}_n\left[g\left(x\right)\right] = \iint \mu_n\left(x+\delta,\omega\right) p\left(\delta,\omega\right) \mathrm{d}\delta\, \mathrm{d}\omega, \tag{4.3}$$

A similar computation provides the covariance between $g(x)$ and $g(x')$ at two arbitrary points x and x' in the following expression.

$$\mathrm{Cov}_n\left[g\left(x\right), g\left(x'\right)\right] = \iiiint \Sigma_n\left(x+\delta,\omega,x'+\delta',\omega'\right) p(\delta,\omega)\, p\left(\delta',\omega'\right) \mathrm{d}\delta\, \mathrm{d}\omega\, \mathrm{d}\delta'\mathrm{d}\omega'. \tag{4.4}$$

Note that taking $x = x'$ gives an expression for the variance.

We will frequently refer to the posterior mean of $g(x)$, and so for brevity we introduce the notation

$$a_n\left(x\right) = \mathbb{E}_n[g(x)], \tag{4.5}$$

which is defined in terms of $\mu_n(\cdot,\cdot)$ by (4.3). Section 4.4.3 gives an explicit expression for $a_n(x)$. Then, if we were to stop after n evaluations of the simulator and choose the solution to (4.1) with the best estimated value, we would choose

$$x_n^* = \operatorname*{argmax}_x \mathbb{E}_n\left[g\left(x\right)\right] = \operatorname*{argmax}_x a_n\left(x\right).$$

In a formal sense, this solution is Bayes-optimal when we are neutral with respect to the risk introduced by our uncertainty about the simulation's output.

We now conduct an analysis to determine the expected solution quality that will result from a single additional evaluation of the simulator. The improvement in solution quality is then the value of the information provided by this additional evaluation.

Consider a given time n, and a given candidate point (θ,ω) to evaluate at time $n+1$. The expected quality of the best solution we can obtain *after* we observe the

sample $y_{n+1} = U(\theta, \omega)$ that results from this evaluation is $\max_x a_{n+1}(x)$. This quantity is unknown at time n, as it depends on the outcome y_{n+1}. If we calculate its expected value at time n, and take the difference between this expected solution quality and the solution quality $\max_x a_n(x)$ that we have at time n, then we obtain the value of the information achieved from measuring (θ, ω) at time $n+1$,

$$V_n(\theta, \omega) = \mathbb{E}_n\Big[\max_x a_{n+1}(x) \,\Big|\, \theta_{n+1}{=}\theta, \omega_{n+1}{=}\omega\Big] - \max_x a_n(x).$$

The algorithm we present in Section 4.3 seeks to evaluate the simulator at the point maximizing the value of information. That is, we want to evaluate at time $n+1$

$$(\theta_{n+1}, \omega_{n+1}) = \operatorname*{argmax}_{\theta,\omega} V_n(\theta, \omega). \tag{4.6}$$

We now show how to compute $V_n(\theta, \omega)$. To perform this computation, we must first determine the distribution of $a_{n+1}(x)$ conditioned on D_n and $(\theta_{n+1}, \omega_{n+1})$ for an arbitrary x. The following lemma describes this distribution.

Lemma 2. *Define*

$$b_n(x, \theta_{n+1}, \omega_{n+1}) = \Bigg[\iiiint \left[\Sigma_n\left(x{+}\delta, \omega, x{+}\delta', \omega'\right) - \Sigma_{n+1}\left(x{+}\delta, \omega, x{+}\delta', \omega'\right)\right] \cdot p(\delta, \omega)\, p\left(\delta', \omega'\right) \mathrm{d}\delta\, \mathrm{d}\omega\, \mathrm{d}\delta'\, \mathrm{d}\omega'\Bigg]^{1/2}. \tag{4.7}$$

Then

$$a_{n+1}(x) \mid D_n, \theta_{n+1}, \omega_{n+1} \sim \mathcal{N}\left(a_n(x), b_n^2(x, \theta_{n+1}, \omega_{n+1})\right). \tag{4.8}$$

Section 4.4.3 gives an explicit expression for $b_n(x, \theta_{n+1}, \omega_{n+1})$. Denote by $\mathscr{X}$ the set of design variables x under consideration. We assume that $\mathscr{X}$ is discrete and finite. Define the following two vectors

$$\begin{aligned} \vec{a}_n &= \{a_n(x) : x \in \mathscr{X}\}, \\ \vec{b}_n(\theta, \omega) &= \{b_n(x, \theta_{n+1}, \omega_{n+1}) \,|\, \theta_{n+1} = \theta, \omega_{n+1} = \omega : x \in \mathscr{X}\}. \end{aligned} \tag{4.9}$$

Then

$$V_n(\theta,\omega) = h\left(\vec{a}_n, \vec{b}_n(\theta,\omega)\right), \tag{4.10}$$

where

$$h\left(\vec{a},\vec{b}\right) := \mathbb{E}\left[\max_i a_i + b_i Z\right] - \max_i a_i,$$

and Z is a standard normal variable. An algorithm for computing h is given in Frazier et al. (2008). The derivative of $V_n(\theta,\omega)$ with respect to θ and ω, denoted by $\nabla_\theta V_n(\theta,\omega)$ and $\nabla_\omega V_n(\theta,\omega)$, is also available, and is described in Section 4.4.4. We can then solve (4.6) using multi-start gradient ascent.

4.3 Algorithm

We now summarize the algorithm that implements this value of information approach.

1. Evaluate U at a number of randomly chosen (θ,ω). Fit a GP prior (see Section 4.4.1) to U based on these evaluations, using maximum likelihood estimation.
2. At each time $n \geq 0$:
   (a) If the stopping rule is met, go to Step 4; else go to Step 2b.
   (b) Update $\vec{a}_n$, $\vec{b}_n(\cdot,\cdot)$, $V_n(\cdot,\cdot)$ and $\nabla V_n(\cdot,\cdot)$ according to (4.9), (4.10) and Section 4.4.3, 4.4.4.
   (c) Maximize $V_n(\cdot,\cdot)$ using multi-start gradient ascent. Let $(\theta_{n+1},\omega_{n+1})$ be the maximizer, and evaluate $U(\theta_{n+1},\omega_{n+1})$.
3. Increase n and return to Step 2.
4. Report $x_n^* = \operatorname{argmax} a_n(x)$ as our final solution.

4.4 Detailed Computations

In this section, we provide explicit expressions for the quantities introduced in the previous sections. We first describe the GP model in Section 4.4.1, and then compute $\mu_n(\cdot,\cdot)$, $\Sigma_n(\cdot,\cdot,\cdot,\cdot)$ in Section 4.4.2, $a_n(\cdot)$, $b_n(\cdot,\theta_{n+1},\omega_{n+1})$ in Section 4.4.3, and $\nabla V_n(\cdot,\cdot)$ in Section 4.4.4.

4.4.1 Gaussian Process Priors

In a GP prior, the covariance between $U(\theta,\omega)$ and $U(\theta',\omega')$ for some

$$\theta = \begin{bmatrix} \theta^{(1)} \\ \vdots \\ \theta^{(d_1)} \end{bmatrix},\ \omega = \begin{bmatrix} \omega^{(1)} \\ \vdots \\ \omega^{(d_2)} \end{bmatrix},\ \theta' = \begin{bmatrix} \theta'^{(1)} \\ \vdots \\ \theta'^{(d_1)} \end{bmatrix},\ \omega' = \begin{bmatrix} \omega'^{(1)} \\ \vdots \\ \omega'^{(d_2)} \end{bmatrix}$$

($d_1 = d_2 = 2$), i.e, $\Sigma_0(\theta,\omega,\theta',\omega')$, is a decreasing function of the distance between (θ,ω) and (θ',ω'). In this work, we use the following square exponential covariance function:

$$\Sigma_0\left(\theta,\omega,\theta',\omega'\right) = \sigma_0^2 \cdot \exp\left(-\sum_{k=1}^{d_1}\alpha_1^{(k)}\left[\theta^{(k)}-\theta'^{(k)}\right]^2 - \sum_{k=1}^{d_2}\alpha_2^{(k)}\left[\omega^{(k)}-\omega'^{(k)}\right]^2\right), \qquad (4.11)$$

where σ_0^2 is the common prior variance, and $\alpha_1^{(1)},\ldots,\alpha_1^{(d_1)},\alpha_2^{(1)},\ldots,\alpha_2^{(d_2)}$ are the length scales. Values of these parameters are usually obtained using maximum likelihood estimation from the observations of U. This and other commonly used covariance functions, e.g., the Matern covariance function, are carefully discussed in Rasmussen and Williams (2006) Section 4.

The mean of a GP prior is usually a linear regression function. Typical choices for $\mu_0(\cdot,\cdot)$ include

1. zero order polynomial (constant): $\mu_0(\theta,\omega) \equiv \xi$,

2. first order polynomial (linear):

$$\mu_0(\theta,\omega) = \sum_{k=1}^{d_1} \xi_1^{(k)} \theta^{(k)} + \sum_{k=1}^{d_2} \xi_2^{(k)} \omega^{(k)},$$

3. second order polynomial (quadratic), etc.,

where $\xi, \xi_1^{(1)}, \ldots, \xi_1^{(d_1)}, \xi_2^{(1)}, \ldots, \xi_2^{(d_2)}$ are the coefficients of the polynomials ("basis functions"). We use the generalized least squares estimates of these coefficients in practice (see Huang et al. (2006) or Rasmussen and Williams (2006) Section 2, 5).

To validate the GP model for our bypass graft surgery application, we apply leave-one-out cross-validation of the model with different covariance functions and regression functions using 137 observations from the cardiovascular simulation. As an example, Figure 4.2 shows the validation results of a GP prior with covariance (4.11) and a constant mean. We see that this model fits the data sufficiently well except for a very small number of outliers.

4.4.2 $\mu_n(\cdot,\cdot)$ and $\Sigma_n(\cdot,\cdot,\cdot,\cdot)$

We briefly describe the GP posterior distribution of U in this subsection. Define

$$\widetilde{Y} = \begin{bmatrix} y_1 - \mu_0(\theta_1,\omega_1) \\ \vdots \\ y_n - \mu_0(\theta_n,\omega_n) \end{bmatrix}, \; T_n = \begin{bmatrix} \Sigma_0(\theta_1,\omega_1,\theta_1,\omega_1) & \cdots & \Sigma_0(\theta_1,\omega_1,\theta_n,\omega_n) \\ \vdots & \ddots & \vdots \\ \Sigma_0(\theta_n,\omega_n,\theta_1,\omega_1) & \cdots & \Sigma_0(\theta_n,\omega_n,\theta_n,\omega_n) \end{bmatrix}, \tag{4.12}$$

and

$$t_n(\cdot,\cdot) = \begin{bmatrix} \Sigma_0(\cdot,\cdot,\theta_1,\omega_1) & \cdots & \Sigma_0(\cdot,\cdot,\theta_n,\omega_n) \end{bmatrix} T_n^{-1}. \tag{4.13}$$

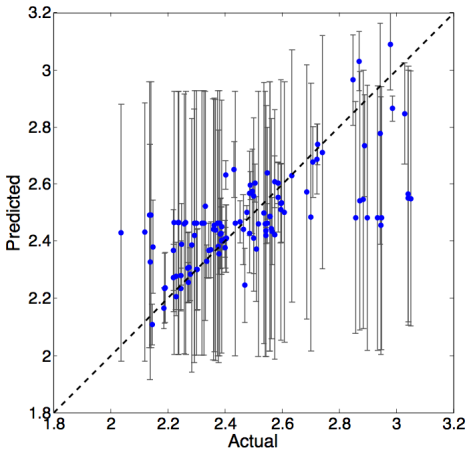

Figure 4.2: Leave-one-out cross-validation of the Gaussian process prior with covariance (4.11) and a constant mean, using 137 observations from the cardiovascular simulation. Each dot compares the actual value of an observation against its predicted value from the other observations. Each error bar is the 95%-confidence interval of the corresponding prediction.

Then for arbitrary θ, ω and θ', ω', by the Kalman filter equations (see, e.g., Gelman et al. (2004) Section 14.6), we have

$$\mu_n(\theta,\omega) = \mu_0(\theta,\omega) + t_n(\theta,\omega)\widetilde{Y}, \tag{4.14}$$

$$\Sigma_n\left(\theta,\omega,\theta',\omega'\right) = \Sigma_0\left(\theta,\omega,\theta',\omega'\right) - t_n(\theta,\omega)\begin{bmatrix}\Sigma_0(\theta',\omega',\theta_1,\omega_1)\\ \vdots \\ \Sigma_0(\theta',\omega',\theta_n,\omega_n)\end{bmatrix}. \tag{4.15}$$

4.4.3 $a_n(\cdot)$ and $b_n(\cdot,\theta_{n+1},\omega_{n+1})$

Based on Section 4.4.1 and 4.4.2, we now explicitly compute $a_n(\cdot)$ and $b_n(\cdot,\theta_{n+1},\omega_{n+1})$, which can then support the calculation of the value of information (4.10).

Suppose that

$$\delta^{(k)} \sim \mathcal{N}\left(\mu_1^{(k)}, 1/\beta_1^{(k)}\right), \qquad k = 1, \ldots, d_1,$$
$$\omega^{(k)} \sim \mathcal{N}\left(\mu_2^{(k)}, 1/\beta_2^{(k)}\right), \ \omega^{(k)} \geq 0, \qquad k = 1, \ldots, d_2,$$

and that $\delta^{(1)}, \ldots, \delta^{(d_1)}, \omega^{(1)}, \ldots, \omega^{(d_2)}$ are mutually independent.

Define

$$S_0(x) = \iint \mu_0(x+\delta, \omega)\, p(\delta, \omega)\, \mathrm{d}\delta\, \mathrm{d}\omega, \tag{4.16}$$

and for $i = 1, \cdots, n+1$,

$$S_i(x) = \iint \Sigma_0(x+\delta, \omega, \theta_i, \omega_i)\, p(\delta, \omega)\, \mathrm{d}\delta\, \mathrm{d}\omega. \tag{4.17}$$

Then by (4.5), (4.3) and (4.12)-(4.17), we have

$$\begin{aligned} a_n(x) &= S_0(x) + \iint t_n(x+\delta, \omega)\, \widetilde{Y}\, p(\delta, \omega)\, \mathrm{d}\delta\, \mathrm{d}\omega \\ &= S_0(x) + \begin{bmatrix} S_1(x) & \cdots & S_n(x) \end{bmatrix} T_n^{-1} \widetilde{Y}. \end{aligned} \tag{4.18}$$

By (4.15) and the Sherman-Morrison-Woodbury formula (see, e.g., Golub and Van Loan (1996)), we can write

$$\Sigma_n\left(\theta, \omega, \theta', \omega'\right) - \Sigma_{n+1}\left(\theta, \omega, \theta', \omega'\right) = \frac{\Sigma_n(\theta, \omega, \theta_{n+1}, \omega_{n+1})\, \Sigma_n(\theta', \omega', \theta_{n+1}, \omega_{n+1})}{\Sigma_n(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1})}.$$

Plug this and (4.12)-(4.17) into (4.7), then we have

$$\begin{aligned} b_n(x, \theta_{n+1}, \omega_{n+1}) &= \frac{\iint \Sigma_n(x+\delta, \omega, \theta_{n+1}, \omega_{n+1})\, p(\delta, \omega)\, \mathrm{d}\delta\, \mathrm{d}\omega}{\sqrt{\Sigma_n(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1})}} \\ &= \frac{S_{n+1}(x) - \begin{bmatrix} S_1(x) & \cdots & S_n(x) \end{bmatrix} T_n^{-1} \alpha}{\sqrt{\Sigma_0(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1}) - \alpha^T T_n^{-1} \alpha}}, \end{aligned} \tag{4.19}$$

where

$$\alpha = \begin{bmatrix} \Sigma_0(\theta_1, \omega_1, \theta_{n+1}, \omega_{n+1}) \\ \vdots \\ \Sigma_0(\theta_n, \omega_n, \theta_{n+1}, \omega_{n+1}) \end{bmatrix}$$

and ^{T} denotes matrix transposition.

Given a GP prior with covariance (4.11) and a constant mean ξ, we now give an explicit expression for $S_i(x)$ ($i = 0, 1, \ldots, n+1$) . By plugging in (4.11) and $p(\delta, \omega)$, we have

$$S_i(x) = \sigma_0^2 \cdot \prod_{k=1}^{d_1} \left[\sqrt{\frac{\beta_1^{(k)}}{2\pi}} \cdot A_i^{(k)} \right] \cdot \prod_{k=1}^{d_2} \left[\sqrt{\frac{\beta_2^{(k)}}{2\pi}} \cdot B_i^{(k)} \right]$$

for $i = 1, \ldots, n+1$, and $S_0(x) = \xi$, where

$$A_i^{(k)} = \int_{-\infty}^{\infty} \exp\left(-\alpha_1^{(k)} \left[x^{(k)} + \delta^{(k)} - \theta_i^{(k)} \right]^2 - \frac{\beta_1^{(k)}}{2} \left[\delta^{(k)} - \mu_1^{(k)} \right]^2 \right) \mathrm{d}\delta^{(k)},$$

$$B_i^{(k)} = \int_{0}^{\infty} \exp\left(-\alpha_2^{(k)} \left[\omega^{(k)} - \omega_i^{(k)} \right]^2 - \frac{\beta_2^{(k)}}{2} \left[\omega^{(k)} - \mu_2^{(k)} \right]^2 \right) \mathrm{d}\omega^{(k)}.$$

Simple algebra then yields

$$A_i^{(k)} = \sqrt{\frac{\pi}{\alpha_1^{(k)} + \frac{1}{2}\beta_1^{(k)}}} \cdot \exp\left[-\alpha_1^{(k)} [x^{(k)} - \theta_i^{(k)}]^2 - \frac{\beta_1^{(k)} [\mu_1^{(k)}]^2}{2} + \frac{\left[\beta_1^{(k)} \mu_1^{(k)} + 2\alpha_1^{(k)} [x^{(k)} - \theta_i^{(k)}] \right]^2}{4\alpha_1^{(k)} + 2\beta_1^{(k)}} \right]$$

and

$$B_i^{(k)} = \frac{\Phi\left(\frac{\beta_2^{(k)} \mu_2^{(k)} + 2\alpha_2^{(k)} \omega_i^{(k)}}{\sqrt{2\alpha_2^{(k)} + \beta_2^{(k)}}} \right)}{\sqrt{\pi^{-1} \left[\alpha_2^{(k)} + \frac{1}{2}\beta_2^{(k)} \right]}} \cdot \exp\left[-\alpha_2^{(k)} [\omega_i^{(k)}]^2 - \frac{1}{2}\beta_2^{(k)} [\mu_2^{(k)}]^2 + \frac{[\beta_2^{(k)} \mu_2^{(k)} + 2\alpha_2^{(k)} \omega_i^{(k)}]^2}{4\alpha_2^{(k)} + 2\beta_2^{(k)}} \right],$$

where Φ is the standard normal CDF.

4.4.4 $\nabla V_n(\cdot, \cdot)$

We briefly describe the algorithm in Frazier et al. (2008) for computing h here to provide notation and context that supports the computation of $\nabla_\theta V_n(\theta, \omega)$ and $\nabla_\omega V_n(\theta, \omega)$. A MATLAB implementation of this algorithm in given in Frazier (2013).

First, $h\left(\vec{a},\vec{b}\right)$ does not change if we reorder the components of the inputs. Thus, without loss of generality, we assume that the b_i are in non-decreasing order and ties in b are broken so that $a_i \leq a_{i+1}$ if $b_i = b_{i+1}$. Then, we remove all those entries i for which $a_i + b_i z < \max_{j\neq i} a_i + b_i z$ for all values of z. An algorithm for doing this is given in Algorithm 1 in Frazier et al. (2009). This gives new vectors $\vec{a}'$ and $\vec{b}'$ with $|\vec{a}'| = |\vec{b}'| \leq |\vec{a}| = |\vec{b}|$, where $|\cdot|$ denotes the length of a vector. Then,

$$h\left(\vec{a},\vec{b}\right) = \sum_{i=1}^{|\vec{a}'|-1} \left(b'_{i+1} - b'_i\right) f\left(-|c_i|\right), \tag{4.20}$$

where

$$\begin{aligned} f(-z) &:= \varphi(z) - z\Phi(-z), \\ c_i &:= -\frac{a'_{i+1} - a'_i}{b'_{i+1} - b'_i} \quad \text{for } i = 1,\ldots,|\vec{a}'|-1, \end{aligned} \tag{4.21}$$

and φ and Φ are the standard normal PDF and CDF.

Now let $\vec{a}'$ and $\vec{b}'$ be the reordering of $\vec{a}_n$ and $\vec{b}_n(\theta,\omega)$ respectively in the acceptance set of Algorithm 1 in Frazier et al. (2009). Then if $|\vec{a}'| = 1$, $V_n(\theta,\omega) = h\left(\vec{a}_n,\vec{b}_n(\theta,\omega)\right) = 0$, and $\nabla V_n(\theta,\omega) = \vec{0}$. Otherwise,

$$\begin{aligned} \nabla V_n(\theta,\omega) &= -\nabla h\left(\vec{a}_n,\vec{b}_n(\theta,\omega)\right) \\ &= \sum_{i=1}^{|\vec{a}'|-1} \left(b'_{i+1}-b'_i\right)\Phi(-|c_i|)\nabla|c_i| - \left(\nabla b'_{i+1} - \nabla b'_i\right) f(-|c_i|) && (4.22) \\ &= \sum_{i=1}^{|\vec{a}'|-1} \left(\nabla b'_{i+1} - \nabla b'_i\right)\left[-|c_i|\,\Phi\left(-|c_i|\right) - f\left(-|c_i|\right)\right] && (4.23) \\ &= \sum_{i=1}^{|\vec{a}'|-1} \left(\nabla b'_i - \nabla b'_{i+1}\right)\varphi\left(|c_i|\right), \end{aligned}$$

where (4.22) follows from (4.20) and $\nabla f = \Phi$; (4.23) follows since $\nabla a'_i = 0$ for all i, and by the definition in (4.21),

$$\nabla|c_i| = \frac{-\left|a'_{i+1} - a'_i\right|\left(\nabla b'_{i+1} - \nabla b'_i\right)}{\left(b'_{i+1} - b'_i\right)^2}.$$

It then suffices to compute $\nabla b_i'$ for all i, or equivalently, $\nabla_{\theta_{n+1}} b_n(x, \theta_{n+1}, \omega_{n+1})$ and $\nabla_{\omega_{n+1}} b_n(x, \theta_{n+1}, \omega_{n+1})$ for all x. Now let ∇ denote the gradient w.r.t. θ_{n+1} or ω_{n+1}. By (4.19), it is clear that

$$\begin{aligned}\nabla b_n(x, \theta_{n+1}, \omega_{n+1}) = \gamma_1 \left(\nabla S_{n+1}(x) - \nabla\left(\alpha^T\right) T_n^{-1} \begin{bmatrix} S_1(x) \\ \vdots \\ S_n(x) \end{bmatrix} \right) \\ - \frac{1}{2}\gamma_1^3 \gamma_2 \left[\nabla\Sigma_0(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1}) - 2\nabla\left(\alpha^T\right) T_n^{-1}\alpha\right],\end{aligned} \tag{4.24}$$

where

$$\begin{aligned}\gamma_1 &= \left[\Sigma_0(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1}) - \alpha^T T_n^{-1}\alpha\right]^{-1/2}, \\ \gamma_2 &= S_{n+1}(x) - \begin{bmatrix} S_1(x) & \cdots & S_n(x) \end{bmatrix} T_n^{-1}\alpha, \\ \nabla\left(\alpha^T\right) &= \Big[\nabla\Sigma_0(\theta_1, \omega_1, \theta_{n+1}, \omega_{n+1}) \cdots \nabla\Sigma_0(\theta_n, \omega_n, \theta_{n+1}, \omega_{n+1})\Big].\end{aligned}$$

With a GP prior (4.11), we can write (4.24) explicitly by plugging in

$$\nabla_{\theta_{n+1}}\Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}) = \begin{cases} \vec{0}, & \text{i=n+1,} \\ 2\alpha_1(\theta_i - \theta_{n+1})\Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}), & \text{i=1,...,n,} \end{cases}$$

$$\nabla_{\omega_{n+1}}\Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}) = \begin{cases} \vec{0}, & \text{i=n+1,} \\ 2\alpha_2(\omega_i - \omega_{n+1})\Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}), & \text{i=1,...,n,} \end{cases}$$

and

$$\begin{aligned}\nabla_{\theta_{n+1}^{(k)}} S_{n+1}(x) &= 2\alpha_1^{(k)} S_{n+1}(x)\left[x^{(k)} - \theta_{n+1}^{(k)} - \nu_1\right], \\ \nabla_{\omega_{n+1}^{(k)}} S_{n+1}(x) &= 2\alpha_2^{(k)} S_{n+1}(x)\left[\frac{\varphi(\nu_2)/\Phi(\nu_2) + \nu_2}{\sqrt{2\alpha_2^{(k)} + \beta_2^{(k)}}} - \omega_{n+1}^{(k)}\right],\end{aligned}$$

where

$$\nu_1 = \frac{\beta_1^{(k)}\mu_1^{(k)} + 2\alpha_1^{(k)}\left(x^{(k)} - \theta_{n+1}^{(k)}\right)}{2\alpha_1^{(k)} + \beta_1^{(k)}}, \qquad \nu_2 = \frac{\beta_2^{(k)}\mu_2^{(k)} + 2\alpha_2^{(k)}\omega_{n+1}^{(k)}}{\sqrt{2\alpha_2^{(k)} + \beta_2^{(k)}}}.$$

4.5 Numerical Results

We now explore the performance of the sampling algorithm proposed in Section 4.3.

4.5.1 A Two-Dimensional Test Problem

We first consider a simplified test problem, where the design variable x and the environmental variable ω are both one-dimensional, and there is no implementation error ($\theta \equiv x$). We assume $\omega \sim \mathcal{N}(1, 1/9)$ and the utility is

$$U(\theta, \omega) = -100(\omega - \theta^2)^2 - (1-\theta)^2. \tag{4.25}$$

In this example, the underlying objective function g has a closed form: $g(x) = (1-x)^2 + 100\left[(1-x^2)^2 + 1/9\right]$.

We plot, in the first row of Figure 4.3, U and g, where the color in the left contour darkens as the value of U decreases, and the gray dot in the right plot gives the true maximizer of g, i.e., the underlying best solution.

We compare our VOI-based design against the uniform design, in which each (θ_n, ω_n) is selected independently and uniformly at random. Both algorithms use Bayesian Quadrature to evaluate g, where each sample path of these algorithms we fit a GP prior distribution with covariance (4.11) and a constant mean to U after 10 initial random evaluations, and re-fit it after each additional evaluation.

To illustrate this statistical model, we plot our posterior beliefs on U and g at a sequence of times in a random sample path of the proposed algorithm in Figure 4.3(rows 2 to 5). The black dot in each plot on the left is the sampling decision, and the black dot in each plot on the right is the corresponding implementation decision (the estimated

best solution). In this sample path, as the posterior means converge to the underlying mean, our estimated best solution converges to the true best solution.

We measure the performance of the two designs by their expected opportunity cost $\mathbb{E}\left[\max_x g(x) - g\left(x_n^*\right)\right]$ at each time n, using the analytical expression for g. As shown in Figure 4.4, the proposed algorithm significantly outperforms random search, which demonstrates the advantage of VOI-based sampling methods over conventional sampling methods when performing simulation optimization with input uncertainties.

4.5.2 A Four-Dimensional Test Problem

We now consider a four-dimensional test problem where

$$\begin{aligned}
&\theta_1 \sim \mathcal{N}(x_1, 1/9), \qquad \theta_2 \sim \mathcal{N}(x_2, 1/36), \\
&v \sim \mathcal{N}(0, 1/9), \qquad r \sim \mathcal{N}(2, 4/9), \\
&U(\theta_1, \theta_2, v, r) = \left[\theta_1^2 + (\theta_1 - v)^2\right] \cdot \left[\theta_2^2 + (\theta_2 - r)^2\right],
\end{aligned}$$

and g can be written in closed form (omitted here).

In addition to the Bayesian designs considered for the two-dimensional test problem, we also compare with a (non-Bayesian) SMF method from Sankaran and Marsden (2010). This method also uses kriging (i.e., a GP) to infer U, but instead of using the posterior on g implied by the posterior on U, it uses stochastic collocation Sankaran and Marsden (2011), which samples in batches (the batch size is 9 in this example) to estimate g in each stage.

Figure 4.5 shows the relative performance of the three designs in this test problem. While our Bayesian Quadrature VOI-based design significantly outperforms the other two, both Bayesian Quadrature algorithms demonstrate better performance than

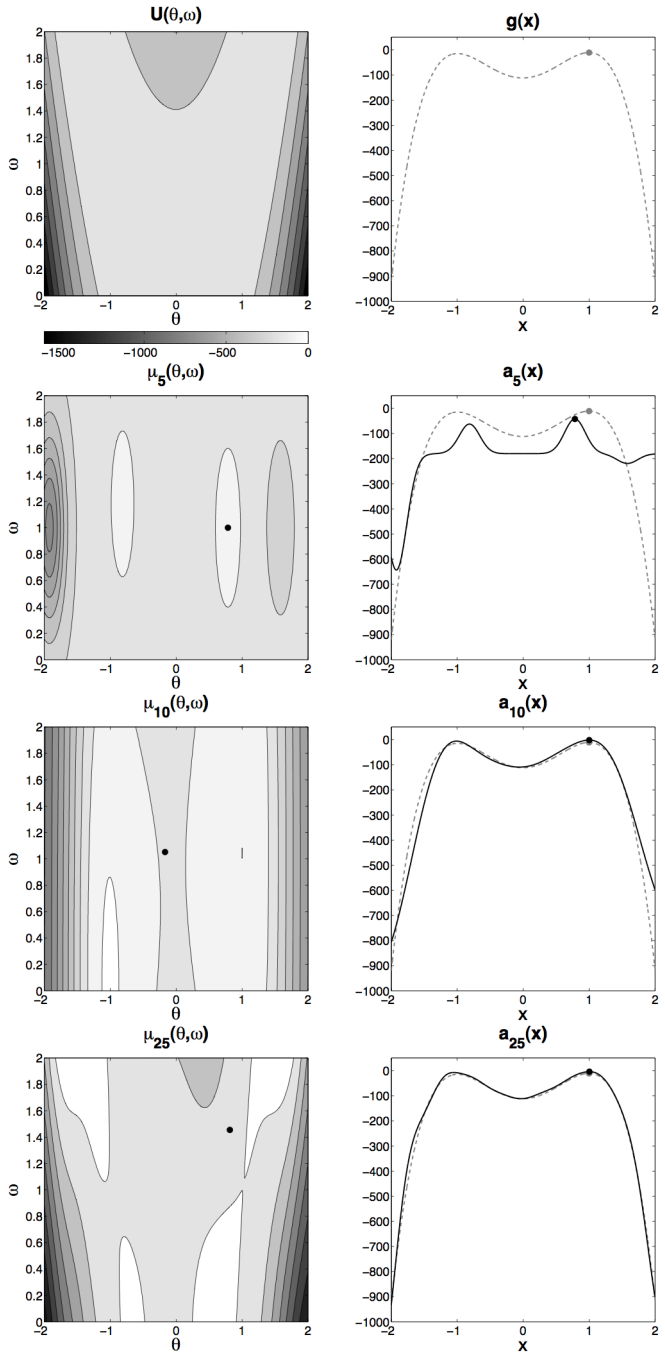

Figure 4.3: Sample path of the proposed algorithm on a two-dimensional test problem. The first row plots U (left) and g (right). The other rows plot the posterior mean of U, i.e., μ_n, on the left, and the posterior mean of g, i.e., a_n, on the right, at times $n = 5, 10, 25$ respectively.

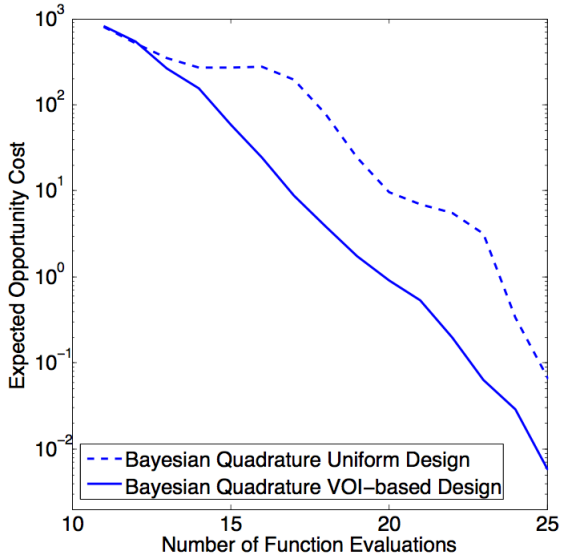

Figure 4.4: Performance of the algorithms in the two-dimensional test problem.

SMF with stochastic collocation. Moreover, after 100 function evaluations, our design provides higher average solution quality than is possible with 500 function evaluations using SMF.

While batching significantly increases the number of function evaluations SMF requires to reach a good solution, SMF with stochastic collocation has the following distinct merits:

1. SMF is a consistent method, in the sense that as sampling effort grows to infinity, its estimated solution converges to the true solution. Our algorithm is not consistent, at least on continuous design spaces, because we discretize the design space to perform VOI-based calculations, and our estimated solution can be no better than the best point in the discretized space. In future work we will explore refining this discretization over time to provide a consistent method.

2. The batches of simulations required by stochastic collocation can be performed in parallel. Our algorithm, on the other hand, is fully sequential.

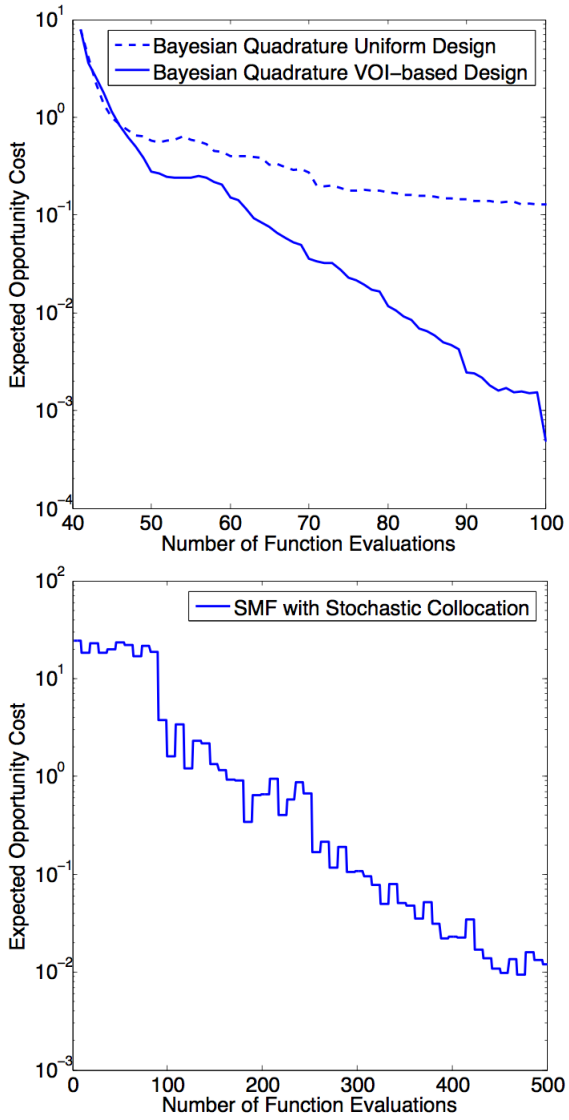

Figure 4.5: Performance of the algorithms in the four-dimensional test problem.

4.5.3 Shape Design of Idealized Bypass Grafts

We also tested the proposed algorithm on the bypass graft shape design problem introduced in Section 4.1, where we used the "anastomosis over a stenosis" simulation performed by Sankaran and Marsden (2010), and assumed the following probability

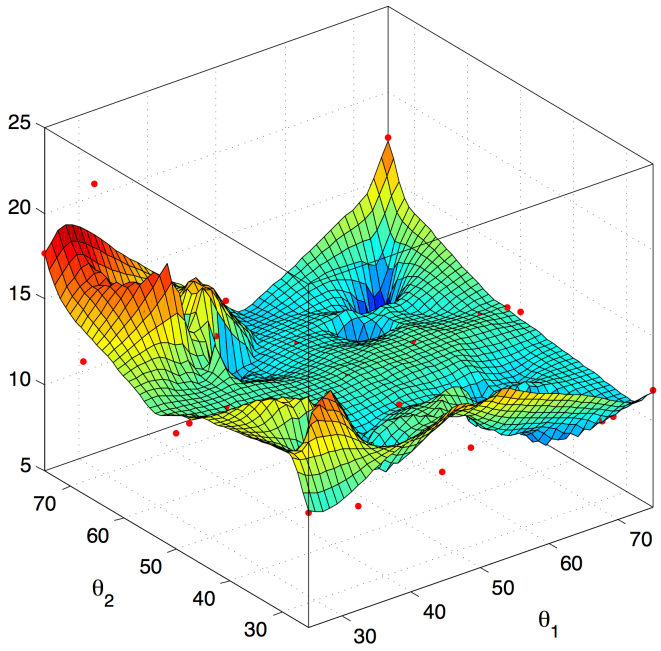

Figure 4.6: 3D plot of the simulation output function (area of low WSS) given $v = 1$ and $r = 0.425$, with red dots showing the projected sampling decisions (evaluated graft attachment angles) in a sample path of the proposed algorithm on the graft design problem.

densities for the simulation inputs,

$$\theta_1 \sim \mathcal{N}(x_1, 25), \quad \theta_2 \sim \mathcal{N}(x_2, 25),$$
$$v \sim \mathcal{N}(1, 0.04), \quad r \sim \mathcal{N}(0.425, 0.000625).$$

As an initial implementation, we adopted a coarse grid mesh size (about 950,000 elements) in the simulations. We plot in Figure 4.6 and Figure 4.7 results we observed from one sample path. In future work, we will use the desired fine mesh size instead, and compare the performance of our algorithm against SMF.

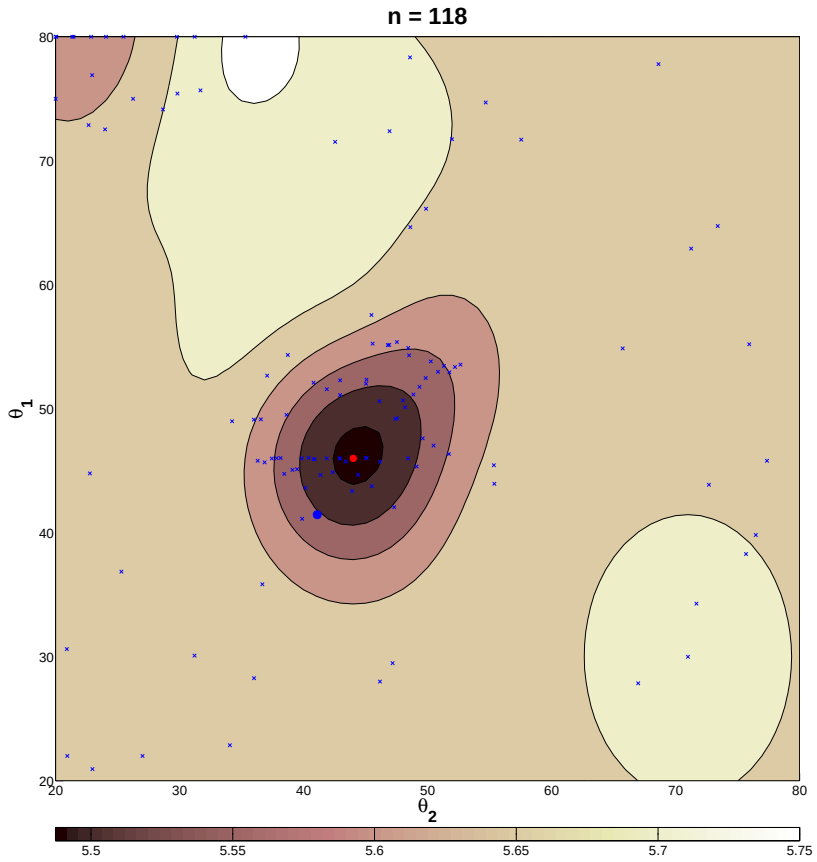

Figure 4.7: Contour plot of the response surface, i.e., posterior estimate of the expected area of low WSS, at time $n = 118$, in a sample path of the proposed algorithm on the graft design problem. The red dot gives the estimated optimal graft attachment angles; the blue dots represent the projected sampling decisions (evaluated angles) in the sample path.

4.6 Conclusions

In this work, we employ a Bayesian approach to optimize computationally expensive simulations under input uncertainties. The simulation output variable, whose expectation is being optimized, is a deterministic function of a low-dimensional random vector. By placing a Gaussian process prior on this deterministic function, we are able to use Bayesian Quadrature to estimate it and its expectation, and apply value of information

computations to allocate simulation effort efficiently.

As demonstrated in the numerical experiments, the proposed algorithm significantly reduces the number of expensive function evaluations required to find a good solution, as compared with other commonly used sampling procedures.

This work is motivated by the shape design of idealized bypass graft models under implementation uncertainty and unsteady flow. Yet the proposed method is appropriate for many other applications of simulation-based optimization where we want to investigate the expected performance of design variables under parameter uncertainties.

CHAPTER 5

UPPER BOUNDS ON THE BAYES-OPTIMAL PROCEDURE FOR RANKING & SELECTION WITH INDEPENDENT NORMAL PRIORS

We consider the Bayesian ranking and selection (R&S) problem (see Section 1.1). For R&S problems with more than a few alternatives, Bayes-optimal sequential procedures become computationally infeasible, due to the curse of dimensionality (Powell 2007). Thus, work in Bayesian R&S has focused in large part on developing sub-optimal procedures. These procedures are evaluated sometimes through theoretical investigations, but also by empirical comparison with previously developed procedures in simulation experiments. One can view the performance of each newly proposed procedure as a lower bound on the value of a Bayes-optimal procedure, and as more procedures are proposed, we may hope these lower bounds will get closer to this Bayes-optimal value.

In this chapter, we focus on a complimentary approach: computing upper bounds on the value of a Bayes-optimal procedure. We focus on one version of the sequential Bayesian R&S problem, independent normal samples with known variance with an infinite horizon and a cost per sample, which was previously considered in Frazier and Powell (2008) and Chick and Frazier (2012). For this problem, we use a Lagrangian relaxation technique to obtain a computable upper bound on the value of a Bayes-optimal procedure. Our computational procedures build on recent work for the problem of sequential Bayesian multiple comparisons with a known standard, for which the Bayes-optimal procedure can be computed efficiently (Xie and Frazier 2013a).

This allows computing an optimality gap, which is the distance between this upper bound and the expected performance of the best existing procedure (which may depend on the specific problem parameters used). This may be used to inform judgments of the value of continued algorithmic development. If this gap is small for a given set

of problem parameters, it tells us that future procedures can improve only by a small margin over the current state-of-the-art. If this gap is large, this may be because existing procedures are far from optimal or because the upper bound is loose, or both. Being able to compute gaps as a function of problem's parameters will allow future researchers to focus development of improved procedures and upper bounds on regions of the problem parameter space where the gap is large.

The mathematical approach that we follow can be viewed as a Lagrangian relaxation of a stochastic dynamic program, which was used in Whittle (1980) to study restless bandit problems, and is also treated in Gittins et al. (2011). Our focus on obtaining upper bounds for sequential decision-making problems is also similar in spirit to recent work on information relaxations in Brown et al. (2010), Brown and Smith (2011), Haugh and Kogan (2004), and Rogers (2002).

We begin in Section 5.1 by formulating the problem. We then describe our upper bound in Sections 5.2 and how to compute it in Section 5.3. In Section 5.4 we describe some special cases in which the bound is tight. In Section 5.5 we apply this bound to a variety of problems. In Section 5.6 we offer concluding remarks.

5.1 The Bayesian Ranking & Selection Problem

We would like to select the best among k alternative systems. We assume that samples from alternative x are normally distributed, with mean θ_x and variance λ_x, and independence across time and across alternatives. The means θ_x are unknown, while the sampling variances λ_x are assumed known. We let $\theta = (\theta_1, \ldots, \theta_k)$. We place a Bayesian prior distribution upon the unknown sampling means,

$$\theta_x \sim \mathcal{N}\left(\mu_{0,x}, \sigma_{0,x}^2\right), \qquad x = 1, \ldots, k,$$

with independence across alternatives. Our goal is to find the alternative with the largest mean θ_x, i.e., to find $x^* \in \text{argmax}_x \theta_x$, and our main challenge in Bayesian R&S is to allocate simulation effort efficiently, so as to best support making this determination.

We assume no fixed computational budget, and instead assume that each sample of alternative x carries a cost $c_x > 0$ that may vary across alternatives. This assumption may be inappropriate when simulations are performed on hardware owned by the simulation analyst, and is instead intended to model the cost structure of on-demand computation purchased through existing cloud computing services, in which the user pays a cost per hour of CPU time consumed.

We index time by $n = 1, 2, \ldots$, and perform our simulations sequentially. At each time n, based on the samples observed so far, we either choose to stop sampling (see below), or we choose an alternative x_n to sample, paying a cost c_{x_n}, and observing a sampled value y_n, $y_n | x_n, \theta \sim \mathcal{N}(\theta_{x_n}, \lambda_{x_n})$. The posterior distribution that results from a sequence of observations obtained in this way is

$$\theta_x \mid x_1, y_1, \ldots, x_n, y_n \sim \mathcal{N}\left(\mu_{n,x}, \sigma^2_{n,x}\right), \qquad x = 1, \ldots, k, \quad n = 1, 2, \ldots,$$

where $\mu_{n,x}$ and $\sigma^2_{n,x}$ can be computed recursively from $\mu_{n-1,x}$, $\sigma^2_{n-1,x}$, x_n, and y_n (see, e.g., DeGroot (1970) or equation 2 in Frazier (2012)). We define $x^*_n \in \text{argmax}_x \mu_{n,x}$.

At time n, if we choose to stop sampling, then we select an alternative as the best based on the previously collected samples, and receive a reward equal to the true value of that alternative. We call $\widehat{x_*}$ the selected alternative, so that the reward received from this selection is $\theta_{\widehat{x_*}}$. We call τ the total number of samples taken. We assume that $\widehat{x_*} = x^*_\tau$, and one can show formally that this choice is the best possible, as measured by expected reward under the prior (see, e.g., Frazier et al. (2008)).

A procedure, or policy, for Bayesian R&S is then comprised of a sampling rule, for

choosing each x_n based on the previous samples $(x_m, y_m : m < n)$, and a stopping rule, for choosing at each time n based on this same information whether to continue sampling or not, and thus implicitly for choosing the number of samples taken τ. (The selection rule is assumed to be $\widehat{x_*} = x^*_\tau$, as stated above.) We refer to such a policy with the notation π.

In Bayesian R&S, we measure the quality of a policy π by the expected net reward under the prior distribution, $\mathbb{E}^\pi \left[\theta_{\widehat{x_*}} - \sum_{n=1}^{\tau} c_{x_n}\right]$. where the expectation is taken both over randomness due to the stochasticity of the samples, and to the uncertainty about θ, and is written using the notation $\mathbb{E}^\pi$ when it depends upon the policy π. This reward includes both the reward due to selection, $\theta_{\widehat{x_*}}$, and the sampling costs, $\sum_{n=1}^{\tau} c_{x_n}$. We use the notation E^π_n to indicate the conditional expectation, with respect to the information available at time n, $(x_m, y_m : m \leq n)$.

This formulation of the sequential Bayesian R&S problem with independent normal samples, known sampling variance, independent normal prior, infinite horizon, and sampling costs, follows that of Frazier and Powell (2008), Chick and Frazier (2012), and is quite similar to the model in Chick and Gans (2009), which assumes a discount factor, and to the model in Frazier et al. (2008), which assumes a finite horizon and no discounting.

With this formulation, the expected value of a Bayes-optimal sampling policy is then

$$r := \sup_\pi \mathbb{E}^\pi \left[\theta_{x^*_\tau} - \sum_{n=1}^{\tau} c_{x_n}\right], \tag{5.1}$$

which depends implicitly on the number of alternatives k, and, the vectors composed of the prior mean $\mu_{0,x}$, prior variance $\sigma^2_{0,x}$, sampling variance λ_x, and sampling cost c_x of each alternative $x = 1, \ldots, k$.

This value r, understood as the solution to a stochastic dynamic programming prob-

lem, is characterized by the dynamic programming equations, e.g., as described in Chick and Frazier (2012), but actually computing r using existing methods is intractable except when k is very small. This intractability is caused (1) by the fact that the state space of the dynamic program is the set of all possible values of the vector of posterior means $(\mu_{nx} : x = 1, \ldots, k)$ and the vector of posterior variances $(\sigma^2_{nx} : x = 1, \ldots, k)$, which has $2k$ dimensions; and (2) by the fact that computation required to solve a dynamic program scales badly with the dimensionality of its state space — a phenomenon which is referred to as the curse of dimensionality (see, e.g., Powell (2007)). Our contribution in this chapter is to provide a tractable method for computing an upper bound on r.

One simple upper bound is immediately apparent from (5.1). Sampling costs are positive, $c_x > 0$, so we have $\sum_{n=1}^{\tau} c_{x_n} \geq 0$. Also, $\theta_{x^*_\tau} \leq \max_x \theta_x$. Thus,

$$r \leq \mathbb{E}\left[\max_x \theta_x\right] := \mathrm{UB}^s$$

where the expectation does not depend on π, and so we use the notation $\mathbb{E}$ rather than $\mathbb{E}^\pi$. UB^s can be computed via numerical integration or Monte Carlo. This upper bound was used as a benchmark in Chick and Frazier (2012).

In the following sections, we will provide a tighter and more sophisticated upper bound than UB^s.

5.2 Upper Bound on the Bayes-Optimal Value: Step 1 (Decomposition)

In this section, we provide an upper bound on (5.1) in terms of a stochastic dynamic program with a special structure that admits solution through decomposition, avoiding

the curse of dimensionality. Later, in Section 5.3, we show how to exploit this structure to allow efficient computation.

We first derive the dynamic program upper bound using a direct approach in Section 5.2.1, giving the bound below in equation (5.5). We then show an alternative derivation using a Lagrangian relaxation in Section 5.2.2, which is more complicated, but relates the upper bound to previous literature, and may also suggest generalizations.

5.2.1 Direct Approach

First, it is convenient to rewrite r as

$$r := \sup_{\pi} \mathbb{E}^{\pi} \left[\theta_{x_\tau^*} - \sum_{n=1}^{\tau} c_{x_n} \right] = \sup_{\pi} \mathbb{E}^{\pi} \left[\max_x \mu_{\tau,x} - \sum_{n=1}^{\tau} c_{x_n} \right], \tag{5.2}$$

where the second equation holds since

$$\mathbb{E}^{\pi} \theta_{x_\tau^*} = \mathbb{E}^{\pi} \left[\mathbb{E}_\tau^{\pi} \theta_{x_\tau^*} \right] = \mathbb{E}^{\pi} \left[\mu_{\tau,x_\tau^*} \right] = \mathbb{E}^{\pi} \left[\max_x \mu_{\tau,x} \right]$$

by the tower property of conditional expectation.

We now state the following lemma, which bounds the reward received from the selection decision, and whose proof involves simple algebraic manipulations.

Lemma 3. *For any $d \in \mathbb{R}$,*

$$\max_x \mu_{\tau,x} \leq d + \sum_{x=1}^{k} (\mu_{\tau,x} - d)^+. \tag{5.3}$$

This inequality holds with equality if and only if $\mu_{\tau,x_\tau^{**}} \leq d \leq \mu_{\tau,x_\tau^*}$, where $x_\tau^{**} = \operatorname{argmax}_{x \neq x_\tau^*} \mu_{\tau,x}$.

Proof. The right-hand side of (5.3) can be rewritten as

$$d + \sum_{x:\ \mu_{\tau,x} \geq d} (\mu_{\tau,x} - d).$$

Consider two cases. If $\max_x \mu_{\tau,x} \geq d$, then this quantity is greater than or equal to $d + \max_x \mu_{\tau,x} - d$, which is equal to the left-hand side of (5.3). If not, so $\max_x \mu_{\tau,x} < d$, then the right-hand side of (5.3) is equal to d, which is greater than the left-hand side of (5.3) by our supposition. In both cases, the right-hand side of (5.3) is greater than or equal to the left-hand side.

Furthermore, $\sum_{x=1}^{k} (\mu_{\tau,x} - d)^+ = \max_x \mu_{\tau,x} - d$ if and only if $\mu_{\tau,x_\tau^{**}} \leq d \leq \mu_{\tau,x_\tau^*}$, hence the result follows. □

It follows from (5.2) and Lemma 3 that, for any $d \in \mathbb{R}$,

$$r \leq d + \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} (\mu_{\tau,x} - d)^+ - \sum_{n=1}^{\tau} c_{x_n} \right] := R(d), \tag{5.4}$$

where we have defined the quantity $R(d)$.

We will see in Section 5.3 that $R(d)$ can be computed efficiently. Moreover, since the bound (5.4) holds for any $d \in \mathbb{R}$, it follows that

$$r \leq \inf_d R(d) = \inf_d \left\{ d + \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} (\mu_{\tau,x} - d)^+ - \sum_{n=1}^{\tau} c_{x_n} \right] \right\} := \mathrm{UB}^*. \tag{5.5}$$

We will see in Section 5.3 that, in addition to being able to compute $R(d)$ efficiently for any d, we can also take the infimum efficiently over d to calculate UB^*. The bound UB^* is the main focus of this chapter.

5.2.2 Lagrangian Approach

Suppose that we enlarge the set of decisions made by each policy π to include an additional variable $a_x \in [0,1]$ for each alternative x, whose value is determined at time τ when sampling stops. Let Π^0 be the set of such policies satisfying the constraint

$\sum_{x=1}^{k} a_x = 1$ almost surely. That is, $\Pi^0 = \left\{\pi : \sum_{x=1}^{k} a_x = 1\right\}$. It follows immediately that

$$r = \sup_{\pi \in \Pi^0} \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} a_x \mu_{\tau,x} - \sum_{n=1}^{\tau} c_{x_n}\right]. \tag{5.6}$$

We now apply a Lagrangian relaxation to (5.6). Let Π^1 be the set of policies that relaxes the constraint on a_x to hold only in expectation, and not almost surely, so $\Pi^1 = \left\{\pi : \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} a_x\right] = 1\right\}$. It follows that $\Pi^0 \subseteq \Pi^1$. Thus, since, taking a supremum over a larger set provides an upper bound, we know that for any $d \in \mathbb{R}$,

$$\begin{aligned}
r &\leq \sup_{\pi \in \Pi^1} \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} a_x \mu_{\tau,x} - \sum_{n=1}^{\tau} c_{x_n}\right] \\
&= \sup_{\pi \in \Pi^1} \left\{\mathbb{E}^{\pi}\left[\sum_{x=1}^{k} a_x \mu_{\tau,x} - \sum_{n=1}^{\tau} c_{x_n}\right] - d\left[\mathbb{E}^{\pi}\left(\sum_{x=1}^{k} a_x\right) - 1\right]\right\} \\
&= d + \sup_{\pi \in \Pi^1} \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} a_x (\mu_{\tau,x} - d) - \sum_{n=1}^{\tau} c_{x_n}\right] = d + \sup_{\pi \in \Pi^1} \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} (\mu_{\tau,x} - d)^+ - \sum_{n=1}^{\tau} c_{x_n}\right] \\
&= d + \sup_{\pi \in \Pi} \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} (\mu_{\tau,x} - d)^+ - \sum_{n=1}^{\tau} c_{x_n}\right]
\end{aligned}$$

In the first equality, we have used $E^{\pi}[\sum_{x=1}^{k} a_x] = 1$ for all $\pi \in \Pi^1$. In the second equality, we have used the linearity of expectation and the fact that d does not depend on π to rearrange terms. In the third equality, we have used that the optimal choice of a_x in this supremum is to choose $a_x = 1$ when $\mu_{\tau,x} - d$ is positive, and $a_x = 0$ when it is negative. In the fourth and last equality, we have switched Π^1 to Π because the value whose supremum being taken does not depend on a_x, making it sufficient to consider $\pi \in \Pi$.

We have derived the same upper bound UB^* in (5.5), where d has played the role of a Lagrange multiplier on the constraint $E^{\pi}[\sum_{x=1}^{k} a_x] = 1$, using a technique similar to that used in Whittle (1980), Gittins et al. (2011).

5.3 Upper Bound on the Bayes-Optimal Value: Step 2 (Computation)

In this section we give a tractable method for computing the upper bound UB* on the expected value of the Bayes-optimal policy. This method has two components. First, we show that $R(d)$ can be computed efficiently for any given d, using a method developed in Xie and Frazier (2013a). Second, we show that $d \mapsto R(d)$ is convex in d, allowing efficient computation of UB* with a standard method for minimization of a one-dimensional convex function that uses only function values, such as Fibonacci search or golden section search (Kiefer 1953).

5.3.1 Computation of $R(d)$

As written in (5.4), computation of $R(d)$ requires solving a dynamic program whose state space includes every possible value of the $2k$-dimensional vector $(\mu_{nx}, \sigma^2_{nx} : x = 1,\ldots,k)$, for which memory requirements and computation time scale exponentially in k, making computation intractable except when k is very small.

The essential idea behind this technique is to use the fact that alternatives are independent of each other, and costs are additive, to rewrite $R(d)$ as the sum of the values of k different sub-problems, each of which is much easier to solve than the original problem,

$$R(d) = \sum_{x=1}^{k} R_x(d), \tag{5.7}$$

where $R_x(d)$ is

$$R_x(d) = d/k + \sup_{\pi \in \Pi^x} \mathbb{E}^{\pi} \left[(\mu_{\tau,x} - d)^+ - \sum_{n=1}^{\tau} c_x \right],$$

and Π^x is the set of policies with $x_n = x$ for all x, i.e., that only measure alternative x. Calculating $R_x(d)$ requires solving a dynamic program whose state space is only two-dimensional, as it contains only $\mu_{n,x}$ and $\sigma^2_{n,x}$ for a single x. Solving such a low-dimensional dynamic program *is* tractable, and the computation to solve k 2-dimensional dynamic programs scales only linearly in k.

Figure 5.1 shows R_x as a function of d, with k taking values $1, 2, 3$ and 100, and the other parameters fixed to $\mu_{0x} = 0$, $\sigma^2_{0,x} = 1$, $\lambda_x = 10$, $c_x = e^{-3}$. The figure suggests that R_x is convex in d, foreshadowing the result on convexity of R (though not R_x) to come in Section 5.3.2.

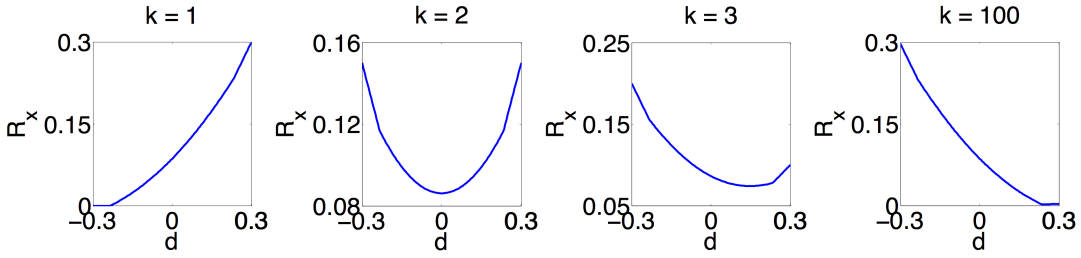

Figure 5.1: $R_x(d)$ as a function of d, when $\mu_{0x} = 0$, $\sigma^2_{0,x} = 1$, $\lambda_x = 10$, $c_x = e^{-3}$, for different values of k. From left to right, $k = 1, 2, 3, 100$. $R(d) = \sum_{x=1}^{k} R_x(d)$, and our upper bound on the value of a Bayes-optimal procedure is $\mathrm{UB}^* = \inf_d R(d)$.

The decomposition (5.7) was previously reported, and justified formally, in Xie and Frazier (2013a), which considered the related problem of multiple-comparisons with a known standard (MCS). The quantity $R(d) - d$ is actually the value of a Bayes-optimal procedure for a variant of this MCS problem, which Xie and Frazier (2013a) refers to as the variant for normal sampling, linear terminal payoff, and infinite horizon.

Here, we briefly describe this variant of the MCS problem. It arises when we sample exactly as in Section 5.1, paying a cost for each sample as before, but when sampling stops, our goal is not to find the alternative with the best true mean θ_x, but is instead

to find the set of alternatives whose true means are above a threshold d, $\{x : \theta_x \geq d\}$. At time τ, in this problem, the decision-maker chooses a set of alternatives, and earns a reward of $\theta_x - d$ for every alternative selected, and a reward of 0 for every alternative not selected. The Bayes-optimal way to make this selection decision is to choose to include an alternative x iff $0 \leq E_\tau[\theta_x - d] = \mu_{\tau,x} - d$, making the optimal set of alternatives to include $B_\tau := \{x : \mu_{\tau,x} - d \geq 0\}$. When the selection decision is made in this way, the resulting expected reward is $\mathbb{E}^\pi\left[\sum_{x\in B_\tau}(\theta_x - d)\right] = \mathbb{E}^\pi\left[\mathbb{E}_\tau\left[\sum_{x\in B_\tau}(\theta_x - d)\right]\right] = \mathbb{E}^\pi\left[\sum_{x\in B_\tau}(\mu_{\tau,x} - d)\right] = \mathbb{E}^\pi\left[\sum_{x=1}^k (\mu_{\tau,x} - d)^+\right]$, by the tower property of conditional expectation. Including the sampling costs, the value of a Bayes-optimal policy for this variant of the MCS problem is $\sup_{\pi\in\Pi}\mathbb{E}^\pi\left[\sum_{x=1}^k (\mu_{\tau,x} - d)^+ - \sum_{n=1}^\tau c_{x_n}\right]$, which is exactly $R(d) - d$.

It is also useful for development in Section 5.4 to define $V_x(d) = R_x(d) - d/k - (\mu_{0,x} - d)^+$. We have subtracted from $R_x(d)$ the term d/k, as well as the value $(\mu_{0,x} - d)^+$ that we would receive in expectation in the MCS problem if we were forced to stop immediately and estimate whether θ_x is above d or below d. Thus, $V_x(d)$ can be seen as the optimal incremental reward that can be obtained through sampling, in an MCS problem with a single alternative. Xie and Frazier (2013a), in its discussion of MCS with linear terminal payoff, normal sampling and infinite horizon, shows that V_x is non-negative, symmetric, maximized at $d = \mu_{0,x}$, with bounded support. The methods described in Xie and Frazier (2013a) actually compute $V_x(d)$, from which $R_x(d)$ can be determined via

$$R_x(d) = d/k + (\mu_{0,x} - d)^+ + V_x(d).$$

5.3.2 Convexity of $R(d)$

We now show that R is convex, allowing efficient computation of $\mathrm{UB}^* = \inf_d R(d)$ with Fibonacci search or golden section search.

Proposition 1. *R is a convex function.*

Proof. Let Π denote the complete set of policies. Since point-wise supremum preserves convexity, it suffices to show that $g\colon \mathbb{R} \times \Pi \mapsto \mathbb{R}$, defined by

$$g(d,\pi) = \mathbb{E}^{\pi}\left[\sum_{x=1}^{k} (\mu_{\tau,x} - d)^{+} - \sum_{n=1}^{\tau} c_{x_n}\right],$$

is convex in d for any given π. Now for a given π,

$$g(d,\pi) = \int h(d,\vec{\omega})\, p^{\pi}(\vec{\omega})\, \mathrm{d}\vec{\omega},$$

where

$$\vec{\omega} = \left(\tau, x_1, \ldots, x_\tau, \mu_{\tau,1}, \ldots, \mu_{\tau,k}\right), \qquad h(d,\vec{\omega}) = \sum_{x=1}^{k} (\mu_{\tau,x} - d)^{+} - \sum_{n=1}^{\tau} c_{x_n},$$

and $p^{\pi}(\vec{\omega})$ is the probability distribution of $\vec{\omega}$ given the specified priors and the sampling policy π. Since h is convex in d for any given $\vec{\omega}$, its integral (infinite sum) g is also convex in d. □

5.4 Special Cases in which the Upper Bound is Tight

In general, the upper bound UB^* is not tight. However, the following theorems present two special cases in which the upper bound UB^* is tight, i.e., in which it is equal to the optimal expected value r.

Theorem 2. *If $k = 1$, then $UB^s = UB^* = r$.*

Proof. First note that $\mathrm{UB}^s = \mathbb{E}[\theta_x] = \mu_{0,x}$, and that the optimal policy is to stop without taking any samples at $\tau = 0$, with $r = \mathbb{E}[\theta_x] = \mu_{0,x}$.

$$R(d) = d + (\mu_{0,x} - d)^+ + V_x(d) = \begin{cases} d + V_x(d), & \text{if } d \geq \mu_{0,x} \\ \mu_{0,x} + V_x(d), & \text{otherwise} \end{cases}.$$

Since V_x is symmetric and maximized at $d = \mu_{0,x}$, we know

$$\mathrm{UB}^* = \inf_d R(d) = R(-\infty) = \mu_{0,x} + V_x(-\infty) = \mu_{0,x}.$$

□

Theorem 3. *If $k = 2$ and $\sigma_{0,1}^2 = 0$, then $UB^* = r$.*

Proof. Since alternative 1 has known value $\mu_{0,1}$, the optimal sampling policy only samples from alternative 2, and $\mu_{n,1} = \mu_{0,1}$ for all n. It follows from (5.2) that

$$r = \sup_\pi \mathbb{E}^\pi \left[\max\left\{\mu_{0,1}, \mu_{\tau,2}\right\} - \sum_{n=1}^{\tau} c_{x_n} \right] = \mu_{0,1} + \sup_\pi \mathbb{E}^\pi \left[(\mu_{\tau,2} - \mu_{0,1})^+ - \sum_{n=1}^{\tau} c_{x_n} \right].$$

By (5.5), we know

$$r \leq \mathrm{UB}^* \leq \mu_{0,1} + \sup_\pi \mathbb{E}^\pi \left[(\mu_{\tau,2} - \mu_{0,1})^+ - \sum_{n=1}^{\tau} c_{x_n} \right] = r.$$

□

5.5 Numerical Results

In this section we apply the technique in Section 5.3 for computing the proposed upper bound UB^* on several test problems, to bound the optimality gaps of existing R&S procedures.

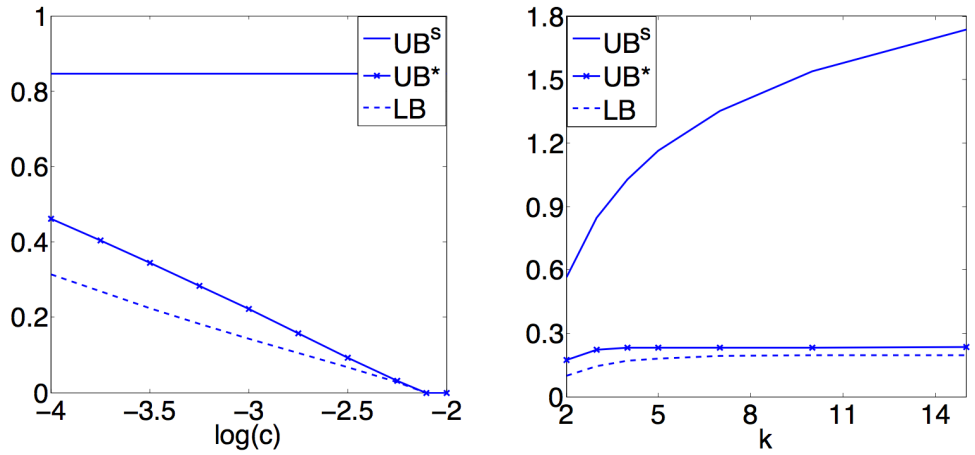

Figure 5.2: Upper bounds UB^* and UB^s and lower bound LB on the value of a Bayes-optimal procedure for R&S problems with $\mu_{0x} = 0$, $\sigma^2_{0,x} = 1$, $\lambda_x = 10$, and $c_x = c$ for all x. UB^s is a simple upper bound computed by supposing that the best alternative is revealed without sampling, and UB^* is computed using the methods described in this chapter. LB is the expected value of the best existing procedure for the given problem parameters, among a collection of procedures tested, as computed using Monte Carlo simulation. The left plot fixes $k = 3$ and varies c from e^{-4} to e^{-2}. The right plot fixes $c = e^{-3}$ and varies k.

In our numerical experiments, we implement a number of benchmarking policies, each of which is the combination of a sampling rule among KG_1 (Frazier and Powell 2008), KG_* (Frazier and Powell 2010), ESP_b (Chick and Frazier 2012), and a stopping rule among $\text{EOC}_{c,k}$ (Chick and Frazier 2012), KG_* (Frazier and Powell 2010), ESP_b (Chick and Frazier 2012). The best expected value of these policies serve as a lower bound on the expected value of the Bayes-optimal policy. We denote this lower bound by LB.

First, in Figure 5.2, we consider a collection of problems with homogeneous priors $\mu_{0x} = 0$, $\sigma^2_{0,x} = 1$, on the unknown means, homogeneous sampling variances $\lambda_x = 10$, and homogeneous sampling costs, $c_x = c$ for all x. We first fix $k = 3$ and vary $\log(c)$ (where log indicates the natural logarithm) within $[-4, -2]$, and then fix $\log(c) = -3$ and vary k within $[2, 15]$. Figure 5.2 shows the resulting upper bounds UB^s, UB^*, and

the lower bound LB, on the Bayes-optimal value.

Figure 5.2 shows that the proposed upper bound UB^* improves dramatically over the naive upper bound UB^s. Moreover, the optimality gap provided by UB^* vanishes as c increases, and stabilizes as k increases. We hypothesize that the optimality gap vanishes as c increases because, when c is large, both the Bayes-optimal R&S procedure, and the Bayes-optimal MCS procedure used to compute UB^*, stop sampling immediately, without taking any samples. When both procedures stop immediately, then $\mu_{\tau,x} = \mu_{0,x}$, and the bound is tight.

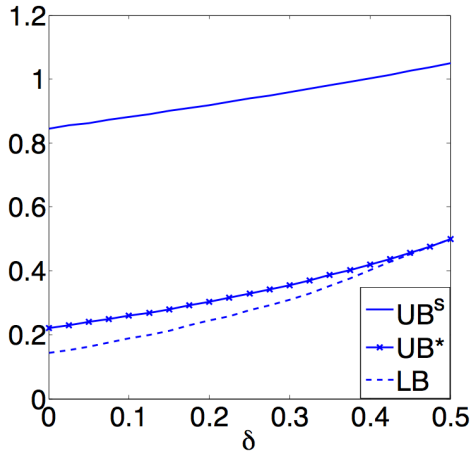

Figure 5.3: Upper bounds UB^* and UB^s and lower bound LB on the value of the Bayes-optimal procedure for R&S problems with heterogeneous priors. We set $\sigma^2_{0,x} = 1$, $\lambda_x = 10$, $c_x = e^{-3}$ for all x, $k = 3$, $\mu_{0,1} = \mu_{0,2} = 0$, and $\mu_{0,3} = \delta$, and plot bounds as a function of δ.

Second, in Figure 5.3, we consider problems with non-homogeneous priors on the unknown means. We set $\sigma^2_{0,x} = 1$, $\lambda_x = 10$, $c_x = e^{-3}$ for all x, fix $k = 3$, $\mu_{0,1} = \mu_{0,2} = 0$, and vary $\mu_{0,3} = \delta$ between $[0, 0.5]$. UB^* gives a significantly tighter bound than does UB^s, and the gap vanishes for sufficiently large δ. We hypothesize that the optimality gap vanishes for large δ because a large difference between the best and second-best *prior* allows a well-chosen value of d to be between the best and second-best values of $\mu_{\tau,x}$ with a probability close to 1, and when this occurs the bound in Lemma 3 is tight.

5.6 Conclusions

We have provided a computationally tractable method for computing upper bounds on the value of a Bayes-optimal procedure for the Bayesian R&S problem with independent normal samples, an independent normal prior, and an infinite horizon with a cost per sample. These upper bounds can be used to judge how far from optimality existing procedures are, for a given set of problem parameters, and can be used to judge where future algorithmic development can be directed.

CHAPTER 6

CONCLUSIONS

In this thesis we have considered a number of sequential learning problems, including multiple comparisons with a standard in Chapter 2, discrete optimization via simulation in Chapter 3, optimization of expensive simulations under parameter uncertainties in Chapter 4, and ranking and selection in Chapter 5.

We make two types of decisions when solving these problems:

1. implementation decisions upon sampling, e.g., which alternative should we select as the best, or which subset of alternatives should we report as being feasible;
2. sequential sampling decisions for allocating simulation effort, e.g., which alternative(s) should we sample at each point in time, and when should we stop sampling.

By formulating these problems within a Bayesian statistical framework, the Bayes-optimal (average-case optimal) implementation decisions become immediately available. Our main challenge lies in making adaptive sampling decisions, for which the Bayes-optimal solution requires solving stochastic dynamic programs and suffers the curse of dimensionality.

Using results from multi-armed bandits, in which a simple index policy is optimal, and decomposition techniques that break high-dimensional dynamic programs into low-dimensional ones, we provide tractable methods for computing the Bayes-optimal sampling procedures for MCS (in Chapter 2), and for computing an upper bound on the value of the Bayes-optimal procedure for R&S (in Chapter 5).

For DOvS problems in Chapter 3 and 4 where the Bayes-optimal sampling policies are computationally intractable, we provide suboptimal procedures that take advantage

of correlations among the values of alternatives (e.g., with Gaussian processes), sampling correlations (e.g., with CRN), and value of information calculations (managing the exploration versus exploitation tradeoff). These procedures improve over existing methods by significantly reducing the number of function evaluations required on average to reach a given solution quality, and are most appropriate for use when working with medium to long-running stochastic simulations, for which the savings on function evaluations outweighs the extra computation time in the algorithm itself.

APPENDIX A

APPENDIX OF CHAPTER 2

A.1 Proof of Table 2.1.

A.1.1 0-1 Terminal Payoff

$h_{0x}(s) = \mathbb{E}[r_0(x;\theta_x) \mid \eta_x \sim \mathscr{D}(s)] = m_{0x} \cdot \mathbb{P}\{x \notin \mathbb{B} \mid \eta_x \sim \mathscr{D}(s)\} = m_{0x}[1 - p_x(s)]$. Similarly $h_{1x}(s) = m_{1x} p_x(s)$. B_n follows directly from definition (2.1).

Payoff Condition 1 holds because function g defined by $g(t) \mapsto \max\{m_{0x}(1-t), m_{1x}\}$ is a convex function and that $\mathbb{E}_0[p_{x_1}(S_{1,x_1})] = \mathbb{E}_0\left[\mathbb{E}_1\left[\mathbf{1}_{\{x_1 \in \mathbb{B}\}}\right]\right] = \mathbb{E}_0\left[\mathbf{1}_{\{x_1 \in \mathbb{B}\}}\right] = p_{x_1}(S_{0,x_1})$. Hence by Jensen's inequality, we have $h_{x_1}(S_{0,x_1}) = g(p_{x_1}(S_{0,x_1})) = g(\mathbb{E}_0[p_{x_1}(S_{1,x_1})]) \leq \mathbb{E}_0[g(p_{x_1}(S_{1,x_1}))] = \mathbb{E}_0[h_{x_1}(S_{1,x_1})]$.

Since for any x and $s \in \Lambda$,

$$\begin{aligned} h_x(s) &= \max\{h_{0x}(s), h_{1x}(s)\} = h_{0x}(s) \cdot \mathbf{1}_{\{s \notin B(x)\}} + h_{1x}(s) \cdot \mathbf{1}_{\{s \in B(x)\}} \\ &\leq m_{0x} \cdot \mathbf{1}_{\{s \notin B(x)\}} + m_{1x} \cdot \mathbf{1}_{\{s \in B(x)\}}, \end{aligned}$$

hence for any n,

$$\mathbb{E}[h_x(S_{n,x}) \mid S_{0,x} = s, x_1 = \cdots = x_n = x] - h_x(s) \leq m_{0x} \cdot \mathbf{1}_{\{s \notin B(x)\}} + m_{1x} \cdot \mathbf{1}_{\{s \in B(x)\}} - h_x(s).$$

Thus Payoff Condition 2 holds with H_x specified in the table.

A.1.2 Linear Terminal Payoff

$h_{0x}(s) = \mathbb{E}\left[r_0(x;\theta_x) \mid \eta_x \sim \mathscr{D}(s)\right] = m_{0x}\left[d_x - \mu_x(s)\right]$. Similarly $h_{1x}(s) = m_{1x}\left[\mu_x(s) - d_x\right]$. B_n follows directly from definition (2.1). Hence

$$h_x(s) = m_{0x}\left[\mu(s) - d_x\right]^- + m_{1x}\left[\mu(s) - d_x\right]^+.$$

Payoff Condition 1 holds because $\mathbb{E}_0\left[\mu_{1x_1}\right] = \mathbb{E}_0\left[\mathbb{E}_1\left(\theta_{x_1}\right)\right] = \mathbb{E}_0\left[\theta_{x_1}\right] = \mu_{0x_1}$. Hence by Jensen's inequality $(\mathbb{E}X)^+ \leq \mathbb{E}(X^+)$ and $(\mathbb{E}X)^- \leq \mathbb{E}(X^-)$, we have

$$\begin{aligned} h_{x_1}(S_{0,x_1}) &= m_{0x}\left(\mu_{0,x_1} - d_x\right)^- + m_{1x}\left(\mu_{0,x_1} - d_x\right)^+ \\ &\leq \mathbb{E}_0\left[m_{0x}\left(\mu_{1,x_1} - d_x\right)^- + m_{1x}\left(\mu_{1,x_1} - d_x\right)^+\right] = \mathbb{E}_0\left[h_{x_1}\left(S_{1,x_1}\right)\right]. \end{aligned}$$

Now for any n and x, by Jensen's inequality,

$$h_x\left(S_{n,x}\right) = m_{0x}\left(\mu_{nx} - d_x\right)^- + m_{1x}\left(\mu_{nx} - d_x\right)^+ \leq \mathbb{E}_n\left[m_{0x}\left(\theta_x - d_x\right)^- + m_{1x}\left(\theta_x - d_x\right)^+\right].$$

Hence

$$\begin{aligned} &\mathbb{E}\left[h_x\left(S_{n,x}\right) \mid S_{0,x} = s, x_1 = \cdots = x_n = x\right] \\ &\leq \mathbb{E}\left\{\mathbb{E}_n\left[m_{0x}\left(\theta_x - d_x\right)^- + m_{1x}\left(\theta_x - d_x\right)^+\right] \,\middle|\, S_{0,x} = s, x_1 = \cdots = x_n = x\right\} \\ &= \mathbb{E}\left[m_{0x}\left(\theta_x - d_x\right)^- + m_{1x}\left(\theta_x - d_x\right)^+ \,\middle|\, \eta_x \sim \mathscr{D}(s)\right]. \end{aligned}$$

Thus Payoff Condition 2 holds with H_x specified in the table.

A.2 Proof of Proposition 1.

Proof. By the tower property of conditional expectation,

$$\mathbb{E}^\pi\left[r\left(B_{\tau\wedge T};\theta,d\right)\right] = \mathbb{E}^\pi\left[\mathbb{E}^\pi_{\tau\wedge T}\left[r\left(B_{\tau\wedge T};\theta,d\right)\right]\right] = \mathbb{E}^\pi\left[\sum_{x=1}^{k} h_x\left(S_{\tau\wedge T,x}\right)\right].$$

We can therefore write (2.5) as follows,

$$V(\mathbf{s}) = \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} h_x(S_{\tau \wedge T,x}) - \sum_{n=1}^{\tau \wedge T} c_{x_n} \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right].$$

We restructure this into a sequence of single period rewards. For a fixed policy π and hence a stopping rule τ, since T is geometrically distributed with parameter $1-\alpha$ and independent of the sampling filtration,

$$\begin{aligned}
&\mathbb{E}^{\pi} \left[\sum_{x=1}^{k} h_x(S_{\tau \wedge T,x}) - \sum_{n=1}^{\tau \wedge T} c_{x_n} \right] \\
&= \mathbb{E}^{\pi} \left[\sum_{t=1}^{\tau} \left[(1-\alpha)\alpha^{t-1} \left(\sum_{x=1}^{k} h_x(S_{t,x}) - \sum_{n=1}^{t} c_{x_n} \right) \right] + \alpha^{\tau} \left(\sum_{x=1}^{k} h_x(S_{\tau,x}) - \sum_{n=1}^{\tau} c_{x_n} \right) \right] \\
&= \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} \left[\sum_{t=1}^{\tau} [(1-\alpha)\alpha^{t-1} h_x(S_{t,x})] + \alpha^{\tau} h_x(S_{\tau,x}) \right] - \sum_{t=1}^{\tau} \left[(1-\alpha)\alpha^{t-1} \sum_{n=1}^{t} c_{x_n} \right] \right. \\
&\qquad \left. - \alpha^{\tau} \sum_{n=1}^{\tau} c_{x_n} \right] \\
&= \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} \left[h_x(S_{0,x}) + \sum_{t=1}^{\tau} \alpha^{t-1} [h_x(S_{t,x}) - h_x(S_{t-1,x})] \right] - (1-\alpha) \sum_{n=1}^{\tau} \left[c_{x_n} \sum_{t=n}^{\tau} \alpha^{t-1} \right] \right. \\
&\qquad \left. - \alpha^{\tau} \sum_{n=1}^{\tau} c_{x_n} \right] \\
&= \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} h_x(S_{0,x}) + \sum_{t=1}^{\tau} \alpha^{t-1} [h_{x_t}(S_{t,x_t}) - h_{x_t}(S_{t-1,x_t})] - \sum_{t=1}^{\tau} \alpha^{t-1} c_{x_t} \right] \\
&= \mathbb{E}^{\pi} \left[\sum_{x=1}^{k} h_x(S_{0,x}) + \sum_{t=1}^{\tau} \alpha^{t-1} [-c_{x_t} + h_{x_t}(S_{t,x_t}) - h_{x_t}(S_{t-1,x_t})] \right].
\end{aligned}$$

The second to last equation follows from simple computation and the fact that at each time $n = 1, 2, \ldots, \tau$, for all non-selected alternatives $x \neq x_n$, we have $S_{n,x} = S_{n-1,x}$.

Define for all $n \geq 1$

$$R_n = -c_{x_n} + h_{x_n}(S_{n,x_n}) - h_{x_n}(S_{n-1,x_n}). \tag{A.1}$$

We then have

$$V(\mathbf{s}) = R_0(\mathbf{s}) + \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{n=1}^{\tau} \alpha^{n-1} R_n \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right]. \tag{A.2}$$

We see from (A.2) that a problem with stopping rule τ that provides a fixed initial reward $R_0(\mathbf{s})$ and a discounted single period reward $\alpha^{n-1}R_n$ at each time $n \geq 1$ is equivalent to the original problem. Since R_0 does not affect the optimal policy, we may subtract it from the value function and instead think of V as the optimal expected incremental reward over R_0. This provides the equivalent problem,

$$V(\mathbf{s}) = \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{n=1}^{\tau} \alpha^{n-1} R_n \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right], \tag{A.3}$$

where we have redefined V to correspond to this equivalent problem in a slight abuse of notation. A policy π that attains the supremum in (A.3) also attains the supremum in (A.2) and (2.5).

For later work, it is convenient to make one additional transformation in which we replace the random variables R_n in (A.3) with deterministic reward functions of the alternatives' states. From (A.3) and the tower property, we know

$$V(\mathbf{s}) = \sup_{\pi} \mathbb{E}^{\pi} \left[\sum_{n=1}^{\tau} \alpha^{n-1} \mathbb{E}^{\pi}[R_n | \mathbf{S}_{n-1}] \,\middle|\, \mathbf{S}_0 = \mathbf{s} \right].$$

Now $\mathscr{R}_x(s) = \mathbb{E}[R_1 \mid S_{0,x} = s, x_1 = x]$ by (2.7). Hence

$$\mathbb{E}^{\pi}[R_n | \mathbf{S}_{n-1}] = \mathscr{R}_{x_n}(S_{n-1,x_n}), \tag{A.4}$$

and then (2.6) follows.

□

A.3 Proof of Proposition 2.

Proof. Under Payoff Condition 1, $\nu_x \geq -c_x$ follows directly from $\mathscr{R}_x \geq -c_x$.

Now under both conditions, we take the following expectations with respect to the sub-problem with a single alternative x. It then follows from (A.1) and (A.4) that

$$\mathscr{R}_x(S_{n,x}) = \mathbb{E}\left[R_{n+1}|S_n\right]. \tag{A.5}$$

Since $\mathscr{R}_x + c_x \geq 0$ and $0 < \alpha < 1$, we write

$$\begin{aligned}
\nu_x(s) + c_x &= \max_{\tau>0} \mathbb{E}\left[\frac{\sum_{n=1}^{\tau} \alpha^{n-1}[\mathscr{R}_x(S_{n-1,x}) + c_x]}{\sum_{n=1}^{\tau} \alpha^{n-1}} \middle| S_{0,x} = s\right] \\
&\leq \max_{\tau>0} \mathbb{E}\left[\sum_{n=1}^{\tau} [\mathscr{R}_x(S_{n-1,x}) + c_x] \;\middle|\; S_{0,x} = s\right] = \mathbb{E}\left[\sum_{n=0}^{\infty} [\mathscr{R}_x(S_{n,x}) + c_x] \;\middle|\; S_{0,x} = s\right] \\
&= \sum_{n=0}^{\infty} \mathbb{E}\left[\mathscr{R}_x(S_{n,x}) + c_x \;\middle|\; S_{0,x} = s\right] \\
&= \sum_{n=1}^{\infty} \mathbb{E}\left[R_n + c_x \;\middle|\; S_{0,x} = s\right] \\
&= \sum_{n=1}^{\infty} \mathbb{E}\left[h_x(S_{n,x}) - h_x(S_{n-1,x}) \;\middle|\; S_{0,x} = s\right] \\
&= \sum_{n=1}^{\infty} \left\{\mathbb{E}\left[h_x(S_{n,x}) \;\middle|\; S_{0,x} = s\right] - \mathbb{E}\left[h_x(S_{n-1,x}) \;\middle|\; S_{0,x} = s\right]\right\} \\
&= \lim_{n\to\infty} \mathbb{E}\left[h_x(S_{n,x}) \;\middle|\; S_{0,x} = s\right] - h_x(s) \leq H_x(s),
\end{aligned}$$

Here, the third line uses the Monotone Convergence Theorem; the fourth line uses (A.5) and the tower property; and the fifth line uses (A.1). The limit in the last equality exists since $\{h_x(S_{n,x})\}_{n\geq 0}$ is a sub-martingale by Payoff Condition 1 and hence $\left\{\mathbb{E}\left[h_x(S_{n,x}) \;\middle|\; S_{0,x} = s\right]\right\}_n$ is an increasing sequence; and the last inequality follows from Payoff Condition 2.

□

A.4 Proof of Proposition 3.

Proof. We receive a zero reward if we take $\tau_x = 0$. Thus $V_x \geq 0$. By (2.7), we can write (2.9) as $V_x(s) = \sup_{\tau_x}\left\{-c_x\tau_x + \mathbb{E}\left[h_x(S_{\tau_x,x}) \mid S_{0,x} = s, x_1 = \cdots = x_{\tau_x} = x\right] - h_x(s)\right\}$.

Since $\tau_x \geq 0$, we know that $V_x(s) \leq H_x(s)$ under Payoff Condition 2. $\square$

A.5 Proof of Proposition 4.

Proof. Since $c_x > 0$, we know that N_x exists under Payoff Condition 3. Applying Payoff Condition 3 to (2.7) gives $\mathscr{R}_x \leq -c_x + \widetilde{H}_x$. Hence for all $n \geq N_x$ and $s \in PS(x;n)$, $\mathscr{R}_x(s) \leq 0$. Now in the sub-problem with single alternative x and initial state $S_{0,x} = s \in PS(x;N_x)$, we know that $S_{t,x} \in PS(x;N_x+t)$ for all $t \geq 0$, and hence $\mathscr{R}_x(S_{t,x}) \leq 0$ for all $t \geq 0$. It follows that

$$V_x(s) = \sup_{\tau_x} \mathbb{E}\left[\sum_{t=0}^{\tau_x-1} \mathscr{R}_x(S_{t,x}) \,\middle|\, S_{0,x} = s\right] = 0, \quad \forall s \in PS(x;N_x)$$

Thus $\tau_x^* \leq N_x$. $\square$

A.6 Proof of Theorem 1.

Proof. For any arbitrary policy π with stopping time τ, we denote the number of times we sample from each alternative x by m_x. Then $\tau = \sum_{x=1}^{k} m_x$. Denote the collection of times when we sample from x, $\{1 \leq n \leq \tau : x_n = x\}$, by $\{n_i^x\}_{1 \leq i \leq m_x}$.

Since the reward for each period only depends on the alternative being sampled during that period, and the states of all the other alternatives remain frozen, we know that the order of the sequence of sampling decisions does not affect the expected total reward. Hence the original problem can be naturally decomposed into k sub-problems

as follows.

$$\begin{aligned}
&\mathbb{E}^{\pi}\left[\sum_{n=1}^{\tau}\mathscr{R}_{x_n}(S_{n-1,x_n})\,\middle|\,\mathbf{S}_0=\mathbf{s}\right]=\sum_{x=1}^{k}\left\{\mathbb{E}^{\pi}\left[\sum_{i=1}^{m_x}\mathscr{R}_x(S_{n_i^x-1,x})\,\middle|\,S_{0,x}=s_x\right]\right\}\\
&=\sum_{x=1}^{k}\left\{\mathbb{E}^{\pi}\left[\sum_{n=1}^{m_x}\mathscr{R}_x(S_{n-1,x})\,\middle|\,S_{0,x}=s_x,x_1=\cdots=x_{m_x}=x\right]\right\}\leq\sum_{x=1}^{k}V_x(s_x),
\end{aligned}\tag{A.6}$$

where the last inequality follows from (2.9). Thus

$$V(\mathbf{s})=\sup_{\pi}\mathbb{E}^{\pi}\left[\sum_{n=1}^{\tau}\mathscr{R}_{x_n}(S_{n-1,x_n})\,\middle|\,\mathbf{S}_0=\mathbf{s}\right]\leq\sum_{x=1}^{k}V_x(s_x).$$

On the other hand, if we adopt a policy satisfying $x_{n+1}\in\{x:S_{n,x}\in\mathbb{C}_x\}$ for all $n\geq 0$ and $\tau=\inf\{n\geq 0:S_{n,x}\notin\mathbb{C}_x,\forall x\}$, then for each x, m_x is exactly the number of samples from alternative x needed for the state of x to leave $\mathbb{C}_x$ for the first time. Hence in each decomposed sub-problem with single alternative x, m_x is an optimal solution equivalent to τ_x^*. As a result,

$$\mathbb{E}^{\pi}\left[\sum_{n=1}^{m_x}\mathscr{R}_x(S_{n-1,x})\,\middle|\,S_{0,x}=s_x,x_1=\cdots=x_{m_x}=x\right]=V_x(s_x),$$

the inequality in (A.6) becomes equality, and $V(\mathbf{s})=\sum_{x=1}^{k}V_x(s_x)$. This also shows that any policy satisfying the conditions is optimal. □

A.7 Proof of Proposition 5.

Proof. We apply ideas similar to those in the proof of Theorem 1. First, $\tau^*=\sum_{x=1}^{k}m_x$. Under the optimal policy, since m_x is the number of samples from alternative x needed for the state of x to leave $\mathbb{C}_x$ for the first time, its distribution is the same as the distribution of τ_x in the decomposed sub-problem with single alternative x. It follows that $m_x\leq N_x$, for each x. Thus $\tau^*\leq\sum_{x=1}^{k}N_x$. □

A.8 Remark 1.

Consider the distribution of $\mathbf{S}_1$ given $\mathbf{S}_0 = (\mathbf{a}, \mathbf{b})$ and $x_1 = x$. Since $\mathbb{P}_0\{y_1 = 1\} = \mathbb{E}_0[y_1] = \mathbb{E}_0[\mathbb{E}_0[y_1 \mid \theta]] = \mathbb{E}_0[\theta_{x_1}] = \mu_{0x_1}$, we immediately have the following expressions:

$$\mathbb{P}\{\mathbf{S}_1 = (\mathbf{a}+\mathbf{e}_x, \mathbf{b}) \mid \mathbf{S}_0 = (\mathbf{a}, \mathbf{b}), x_1 = x\} = \mathbb{P}\{y_1 = 1 \mid \mathbf{S}_0 = (\mathbf{a}, \mathbf{b}), x_1 = x\} = a_x/(a_x + b_x),$$

$$\mathbb{P}\{\mathbf{S}_1 = (\mathbf{a}, \mathbf{b}+\mathbf{e}_x) \mid \mathbf{S}_0 = (\mathbf{a}, \mathbf{b}), x_1 = x\} = \mathbb{P}\{y_1 = 0 \mid \mathbf{S}_0 = (\mathbf{a}, \mathbf{b}), x_1 = x\} = b_x/(a_x + b_x).$$

A.9 Proof of Table 2.2.

A.9.1 Preparatory Material

Use Stirling's approximation, for large a and b,

$$B(a,b) \sim \sqrt{2\pi}\frac{a^{a-\frac{1}{2}}b^{b-\frac{1}{2}}}{(a+b)^{a+b-\frac{1}{2}}}.$$

More generally, we have the following lemma.

Lemma 4. *For $a, b \geq 1$,*

$$B(a,b) \geq \sqrt{2\pi}\frac{a^{a-\frac{1}{2}}b^{b-\frac{1}{2}}}{(a+b)^{a+b-\frac{1}{2}}}.$$

Proof. By Stirling's asymptotic series (see, e.g., Abramowitz and Stegun (1964) and Sloane (2007)), we write

$$\Gamma(z) = e^{-z}z^{z-\frac{1}{2}}\sqrt{2\pi}e^{\lambda_z}, \quad \text{with } \frac{1}{12z+1} < \lambda_z < \frac{1}{12z}.$$

Hence

$$B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)} = \sqrt{2\pi}\exp\{\lambda_a + \lambda_b - \lambda_{a+b}\}\frac{a^{a-\frac{1}{2}}b^{b-\frac{1}{2}}}{(a+b)^{a+b-\frac{1}{2}}}.$$

The result then follows since for $a, b \geq 1$,

$$\begin{aligned}\lambda_a + \lambda_b - \lambda_{a+b} &> \frac{1}{12a+1} + \frac{1}{12b+1} - \frac{1}{12(a+b)} \\ &= \frac{(12a+\frac{1}{2})^2 + (12b+\frac{1}{2})^2 + 144ab - \frac{3}{2}}{12(a+b)(12a+1)(12b+1)} > 0.\end{aligned}$$

□

Lemma 5. *For $a, b \geq 1$ and $d_x \in (0,1)$,*

$$\frac{d_x^a (1-d_x)^b}{B(a,b)} \leq \frac{1}{2}\sqrt{\frac{a+b}{2\pi}}.$$

Proof. Denote $t = a+b$ and $\mu = \frac{a}{a+b}$. Then by Lemma 4,

$$\frac{d_x^a (1-d_x)^b}{B(a,b)} \leq \frac{d_x^{\mu t}(1-d_x)^{(1-\mu)t} t^{t-\frac{1}{2}}}{\sqrt{2\pi}(\mu t)^{\mu t - \frac{1}{2}}[(1-\mu)t]^{(1-\mu)t-\frac{1}{2}}} = \sqrt{\frac{\mu(1-\mu)t}{2\pi}}\left[\left(\frac{d_x}{\mu}\right)^{\mu}\left(\frac{1-d_x}{1-\mu}\right)^{1-\mu}\right]^t.$$

Define function $g(\cdot)$ on $(0,1)$ by

$$g(u) = \left(\frac{d_x}{u}\right)^u \left(\frac{1-d_x}{1-u}\right)^{1-u}.$$

Then since $\frac{d}{du}[\log g(u)] = \log \frac{d_x(1-u)}{u(1-d_x)}$, it is easy to see that $g(\cdot)$ is unimodal on $(0,1)$ with peak at $u = d_x$. Finally, since $\mu \in (0,1)$, we know $\sqrt{\mu(1-\mu)} \leq \frac{1}{2}$, and hence

$$\frac{d_x^a (1-d_x)^b}{B(a,b)} \leq \frac{1}{2}\sqrt{\frac{t}{2\pi}}\,[g(\mu)]^t \leq \frac{1}{2}\sqrt{\frac{t}{2\pi}}\,[g(d_x)]^t = \frac{1}{2}\sqrt{\frac{t}{2\pi}} = \frac{1}{2}\sqrt{\frac{a+b}{2\pi}}.$$

A.9.2 0-1 Terminal Payoff

We first state the following property of the regularized incomplete beta function $I_d(\cdot,\cdot)$,

$$I_d(a+1,b) = I_d(a,b) - \frac{d^a(1-d)^b}{aB(a,b)}; \quad I_d(a,b+1) = I_d(a,b) + \frac{d^a(1-d)^b}{bB(a,b)}.$$

Hence by Remark 1, if

$$I_{d_x}(a,b) \geq \frac{m_{1x}}{m_{0x}+m_{1x}} + \frac{d_x^a(1-d_x)^b}{aB(a,b)},$$

then

$$\begin{aligned}
&\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) \\
&= \frac{a}{a+b} h_x(a+1,b) + \frac{b}{a+b} h_x(a,b+1) - h_x(a,b) \\
&= m_{0x}\left[\frac{a}{a+b} I_{d_x}(a+1,b) + \frac{b}{a+b} I_{d_x}(a,b+1) - I_{d_x}(a,b)\right] \\
&= m_{0x}\left[-\frac{a}{a+b} \cdot \frac{d_x^a (1-d_x)^b}{aB(a,b)} + \frac{b}{a+b} \cdot \frac{d_x^a (1-d_x)^b}{bB(a,b)}\right] = 0.
\end{aligned}$$

Similarly, if

$$I_{d_x}(a,b) \le \frac{m_{1x}}{m_{0x}+m_{1x}} - \frac{d_x^a (1-d_x)^b}{bB(a,b)},$$

we have $\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) = 0$.

Now, if

$$\frac{m_{1x}}{m_{0x}+m_{1x}} \le I_{d_x}(a,b) < \frac{m_{1x}}{m_{0x}+m_{1x}} + \frac{d_x^a (1-d_x)^b}{aB(a,b)},$$

then

$$\begin{aligned}
&\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) \\
&= \frac{a}{a+b} m_{1x}[1 - I_{d_x}(a+1,b)] + m_{0x}\left[\frac{b}{a+b} I_{d_x}(a,b+1) - I_{d_x}(a,b)\right] \\
&= \frac{a}{a+b}\left[m_{1x} - (m_{0x}+m_{1x}) I_{d_x}(a,b)\right] + \frac{m_{0x}+m_{1x}}{a+b} \cdot \frac{d_x^a (1-d_x)^b}{B(a,b)} \\
&\le \frac{m_{0x}+m_{1x}}{a+b} \cdot \frac{d_x^a (1-d_x)^b}{B(a,b)}.
\end{aligned}$$

Similarly, if

$$\frac{m_{1x}}{m_{0x}+m_{1x}} - \frac{d_x^a (1-d_x)^b}{bB(a,b)} < I_{d_x}(a,b) \le \frac{m_{1x}}{m_{0x}+m_{1x}},$$

we still have

$$\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) \le \frac{m_{0x}+m_{1x}}{a+b} \cdot \frac{d_x^a (1-d_x)^b}{B(a,b)}. \tag{A.7}$$

Thus (A.7) holds for all $(a,b) \in \Lambda = [1,+\infty) \times [1,+\infty)$. Now applying Lemma 5, we know

$$\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) \le \frac{m_{0x}+m_{1x}}{2\sqrt{2\pi(a+b)}} := \widetilde{H}_x(a,b).$$

Since in the sub-problem with single alternative x, $a_{nx}+b_{nx}=a_{0x}+b_{0x}+n$, and hence $PS(x;n)=\{(a,b)\in\Lambda : a+b\geq 2+n\}$. It follows that

$$\lim_{n\to\infty}\left[\sup_{(a,b)\in PS(x;n)}\widetilde{H}_x(a,b)\right]=\lim_{n\to\infty}\left[\frac{m_{0x}+m_{1x}}{2\sqrt{2\pi(2+n)}}\right]=0.$$

Thus Payoff Condition 3 holds and

$$N_x=\min\left\{n\geq 0:\frac{m_{0x}+m_{1x}}{2\sqrt{2\pi(2+n)}}\leq c_x\right\}=\left(\lceil\frac{(m_{0x}+m_{1x})^2}{8\pi c_x^2}\rceil-2\right)^+.$$

A.9.3 Linear Terminal Payoff

First suppose $a\geq b$, then $\frac{a}{a+b}\geq\frac{a+1}{a+b+1}$. Since for all $x,y\in\mathbb{R}, x^+-y^+\leq|x-y|$, we have

$$\begin{aligned}
&\mathbb{E}\left[h_x(S_{1,x})\mid S_{0,x}=(a,b),x_1=x\right]-h_x(a,b)\\
=&\frac{a}{a+b}h_x(a+1,b)+\frac{b}{a+b}h_x(a,b+1)-h_x(a,b)\\
=&\frac{a}{a+b}\left[m_{0x}\left(d_x-\frac{a+1}{a+b+1}\right)^+ +m_{1x}\left(\frac{a+1}{a+b+1}-d_x\right)^+\right]\\
&+\frac{b}{a+b}\left[m_{0x}\left(d_x-\frac{a}{a+b+1}\right)^+ +m_{1x}\left(\frac{a}{a+b+1}-d_x\right)^+\right]\\
&-\left[m_{0x}\left(d_x-\frac{a}{a+b}\right)^+ +m_{1x}\left(\frac{a}{a+b}-d_x\right)^+\right]\\
=&\frac{a\cdot m_{0x}}{a+b}\left[\left(d_x-\frac{a+1}{a+b+1}\right)^+-\left(d_x-\frac{a}{a+b}\right)^+\right]\\
&+\frac{a\cdot m_{1x}}{a+b}\left[\left(\frac{a+1}{a+b+1}-d_x\right)^+-\left(\frac{a}{a+b}-d_x\right)^+\right]\\
&+\frac{b\cdot m_{0x}}{a+b}m_{0x}\left[\left(d_x-\frac{a}{a+b+1}\right)^+-\left(d_x-\frac{a}{a+b}\right)^+\right]\\
&+\frac{b\cdot m_{1x}}{a+b}\left[\left(\frac{a}{a+b+1}-d_x\right)^+-\left(\frac{a}{a+b}-d_x\right)^+\right]\\
\leq&\frac{a\cdot m_{0x}}{a+b}\left|\frac{a+1}{a+b+1}-\frac{a}{a+b}\right|+0+\frac{b\cdot m_{0x}}{a+b}\left|\frac{a}{a+b+1}-\frac{a}{a+b}\right|+0\\
=&\frac{2ab\cdot m_{0x}}{(a+b)^2(a+b+1)}.
\end{aligned}$$

Similarly if $a < b$, then

$$\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) \leq \frac{ab(m_{0x}+m_{1x})}{(a+b)^2(a+b+1)}.$$

In both cases, we have

$$\begin{aligned}
&\mathbb{E}\left[h_x(S_{1,x}) \mid S_{0,x} = (a,b), x_1 = x\right] - h_x(a,b) \\
&\leq \frac{ab\left[\max\{m_{0x}, m_{1x}\} + m_{0x}\right]}{(a+b)^2(a+b+1)} \leq \frac{\max\{m_{0x}, m_{1x}\} + m_{0x}}{4(a+b+1)} := \widetilde{H}_x(a,b).
\end{aligned}$$

Since $PS(x;n) = \{(a,b) \in \Lambda : a+b \geq 2+n\}$, we have

$$\lim_{n\to\infty}\left[\sup_{(a,b)\in PS(x;n)} \widetilde{H}_x(a,b)\right] = \lim_{n\to\infty}\left[\frac{\max\{m_{0x}, m_{1x}\} + m_{0x}}{4(3+n)}\right] = 0.$$

Thus Payoff Condition 3 holds and

$$N_x = \min\left\{n \geq 0 : \frac{\max\{m_{0x}, m_{1x}\} + m_{0x}}{4(3+n)} \leq c_x\right\} = \left(\left\lceil \frac{\max\{m_{0x}, m_{1x}\} + m_{0x}}{4c_x} \right\rceil - 3\right)^+.$$

□

A.10 Remark 2.

From here we let Φ and φ denote the standard normal cdf and pdf respectively. Define the following events for a standard normal random variable Z:

$$\begin{aligned}
A^i &= \left\{\mu + \tilde{\sigma}_x(\beta)Z \in \left[\mu_x^i(\beta+\beta_x^\varepsilon) - \delta/2, \mu_x^i(\beta+\beta_x^\varepsilon) + \delta/2\right]\right\} \text{ for all } i, \\
A &= \left\{\mu + \tilde{\sigma}_x(\beta)Z \notin \left[\underline{\mu}_x(\beta+\beta_x^\varepsilon), \overline{\mu}_x(\beta+\beta_x^\varepsilon)\right]\right\}.
\end{aligned}$$

Then

$$
\begin{aligned}
&\mathbb{E}\left[V_x\left(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon}\right)\right]\\
\approx&\sum_i\left\{\mathbb{E}\left[V_x\left(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon}\right)\mid A^i\right]\cdot\mathbb{P}(A^i)\right\}+\mathbb{E}\left[V_x\left(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon}\right)\mid A\right]\cdot\mathbb{P}(A)\\
\approx&\sum_i\left\{V_x\left(\mu_x^i(\beta+\beta_x^{\varepsilon}),\beta+\beta_x^{\varepsilon}\right)\cdot\mathbb{P}(A^i)\right\}+0\cdot\mathbb{P}(A)\\
=&\sum_i\left\{V_x\left(\mu_x^i(\beta+\beta_x^{\varepsilon}),\beta+\beta_x^{\varepsilon}\right)\right.\\
&\qquad\left.\cdot\left[\Phi\left(\frac{\mu_x^i(\beta+\beta_x^{\varepsilon})+\delta/2-\mu}{\tilde{\sigma}_x(\beta)}\right)-\Phi\left(\frac{\mu_x^i(\beta+\beta_x^{\varepsilon})+\delta/2-\mu}{\tilde{\sigma}_x(\beta)}\right)\right]\right\}.
\end{aligned}
$$

A.11 Remark 3.

We only need to give explicit expressions for $\mathbb{E}\left[h_x\left(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon}\right)\right]$.

A.11.1 0-1 Terminal Payoff

Let $\rho=d_x-\mu$ and $\xi=\tilde{\sigma}_x^{-1}(\beta)\left[\rho-\Phi^{-1}\left(\frac{m_{1x}}{m_{0x}+m_{1x}}\right)/\sqrt{\beta+\beta_x^{\varepsilon}}\right]$, then

$$
\begin{aligned}
&\mathbb{E}\left[h_x\left(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon}\right)\right]\\
=&\,\mathbb{E}\left[\max\left\{m_{0x}\cdot\Phi\left(\sqrt{\beta+\beta_x^{\varepsilon}}\left(d_x-\mu-\tilde{\sigma}_x(\beta)Z\right)\right),\right.\right.\\
&\qquad\left.\left.m_{1x}\left[1-\Phi\left(\sqrt{\beta+\beta_x^{\varepsilon}}\left(d_x-\mu-\tilde{\sigma}_x(\beta)Z\right)\right)\right]\right\}\right]\\
=&\,m_{0x}\int_{-\infty}^{\xi}\Phi\left(\sqrt{\beta+\beta_x^{\varepsilon}}\left(\rho-\tilde{\sigma}_x(\beta)z\right)\right)\varphi(z)dz\\
&+m_{1x}\int_{\xi}^{+\infty}\left[1-\Phi\left(\sqrt{\beta+\beta_x^{\varepsilon}}\left(\rho-\tilde{\sigma}_x(\beta)z\right)\right)\right]\varphi(z)dz\\
=&\,m_{0x}\cdot X+m_{1x}\cdot Y,
\end{aligned}
$$

where X and Y are defined to be the first and second integral respectively in the second line. Let Z_1 and Z_2 be two independent standard normal random variables. Then

$$
X=\mathbb{P}\left[Z_1\le\sqrt{\beta+\beta_x^{\varepsilon}}\left(\rho-\tilde{\sigma}_x(\beta)Z_2\right),Z_2\le\xi\right]=\mathbb{P}\left[(Z_2,Z_3)\le(\xi,0)\right],
$$

where $Z_3 := Z_1 - \sqrt{\beta+\beta_x^\varepsilon}\,(\rho-\tilde{\sigma}_x(\beta)Z_2) \sim \mathcal{N}\left(-\rho\sqrt{\beta+\beta_x^\varepsilon}, 1+\beta_x^\varepsilon/\beta\right)$ and $\mathrm{Cov}(Z_2,Z_3)=\sqrt{\beta_x^\varepsilon/\beta}$. It follows that X can be evaluated from the cdf of a bivariate normal distribution. A similar argument can be applied to Y.

A.11.2 Linear Terminal Payoff

$$\begin{aligned}
&\mathscr{R}_x(\mu,\beta)+c_x = \mathbb{E}\left[h_x(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^\varepsilon)\right]-h_x(\mu,\beta)\\
&= m_{0x}\left\{\mathbb{E}\left[(d_x-\mu-\tilde{\sigma}_x(\beta)Z)^+\right]-(d_x-\mu)^+\right\}\\
&\quad + m_{1x}\left\{\mathbb{E}\left[(\mu+\tilde{\sigma}_x(\beta)Z-d_x)^+\right]-(\mu-d_x)^+\right\}\\
&= m_{0x}\cdot\Delta\left(d_x-\mu,\tilde{\sigma}_x^{-2}(\beta)\right)+m_{1x}\cdot\Delta\left(\mu-d_x,\tilde{\sigma}_x^{-2}(\beta)\right),
\end{aligned} \tag{A.8}$$

which can be computed using Lemma 7.

A.12 Proof of Table 2.3

A.12.1 Preparatory Material

Lemma 6. *Let* $u_0 := \Phi^{-1}\left(\frac{m_{1x}}{m_{0x}+m_{1x}}\right)$. *Define functions* $g_x(\cdot)$, $m_x(\cdot)$, $s_x(\cdot)$, $t_x(\cdot)$, $s(\cdot)$ *at* $(-\infty,+\infty)$ *and* $g(\cdot)$ *at* $(-\infty,+\infty)^2$ *by*

$$
\begin{aligned}
g_x(u) &= \max\left\{m_{0x}\Phi(u), m_{1x}\left[1-\Phi(u)\right]\right\},\\
m_x(u) &= m_{0x}\mathbf{1}_{\{u\geq u_0\}} + m_{1x}\mathbf{1}_{\{u<u_0\}},\\
s_x(u) &= \begin{cases} \sup_v\left[\frac{m_{1x}\Phi(-v)-m_{0x}\Phi(u)}{u-v}\right] & \text{if } u\geq u_0,\\ \sup_v\left[\frac{m_{0x}\Phi(v)-m_{1x}\Phi(-u)}{v-u}\right] & \text{if } u<u_0,\end{cases}\\
t_x(u) &= \begin{cases} m_{0x}\sup_{v\geq u}\left[\frac{\Phi(v)-\Phi(u)}{v-u}\right] & \text{if } u\geq u_0,\\ m_{1x}\sup_{v\leq u}\left[\frac{\Phi(v)-\Phi(u)}{v-u}\right] & \text{if } u<u_0,\end{cases}\\
s(u) &= \max\left\{s_x(u), t_x(u)\right\},\\
g(u,v) &= g_x(u) + |v-u|s(u).
\end{aligned}
$$

Then for all u and v,

1. $0 \leq s(u) \leq \max\{m_{0x}, m_{1x}\}/\sqrt{2\pi}$,
2. $|u|s(u) \leq \max\{m_{0x}, m_{1x}\}(1+1/\sqrt{2\pi e})$,
3. $g_x(v) \leq g(u,v)$.

Proof. • If $u \geq u_0$, then $g_x(u) = m_{0x}\Phi(u)$ and $m_x(u) = m_{0x}$. Also, if $m_{0x}\Phi(u) \geq m_{1x}$, then $s_x(u) = 0$; otherwise there exists a maximum $v^* < u$ satisfying the first order condition, i.e.,

$$s_x(u) = \frac{m_{1x}\Phi(-v^*) - m_{0x}\Phi(u)}{u-v^*} = m_{1x}\varphi(v^*). \tag{A.9}$$

If $u \geq 0$, then $t_x(u) = m_{0x}\varphi(u)$; otherwise there exists some $v^{**} > u$ such that

$$t_x(u) = m_{0x}\frac{\Phi(v^{**}) - \Phi(u)}{v^{**} - u} = m_{0x}\varphi(v^{**}). \tag{A.10}$$

- Similarly if $u < u_0$, then $g_x(u) = m_{1x}\Phi(-u)$ and $m_x(u) = m_{1x}$. If $m_{1x}\Phi(-u) \geq m_{0x}$, then $s_x(u) = 0$; otherwise there exists some $v^* > u$ such that

$$s_x(u) = \frac{m_{0x}\Phi(v^*) - m_{1x}\Phi(-u)}{v^* - u} = m_{0x}\varphi(v^*).$$

  If $u \leq 0$, then $t_x(u) = m_{1x}\varphi(u)$; otherwise there exists $v^{**} < u$ such that

$$t_x(u) = m_{1x}\frac{\Phi(v^{**}) - \Phi(u)}{v^{**} - u} = m_{1x}\varphi(v^{**}).$$

It follows that $s(u) \leq \max\{m_{0x}, m_{1x}\}\varphi(0) = \max\{m_{0x}, m_{1x}\}/\sqrt{2\pi}$. Hence the **first** inequality holds.

For the **second** inequality, we notice that function $u \mapsto |u|\varphi(u)$ is maximized at $u = \pm 1$ with maximum value $\varphi(1) = 1/\sqrt{2\pi e}$. Assume $u \geq u_0$. We first show $|u|s_x(u) \leq \max\{m_{0x}, m_{1x}\}(1 + 1/\sqrt{2\pi e})$. This result holds immediately when $m_{0x}\Phi(u) > m_{1x}$, i.e., $s_x(u) = 0$. Otherwise by (A.9),

$$\begin{aligned} |u|s_x(u) &\leq m_{1x}\Phi(-v^*) - m_{0x}\Phi(u) + m_{1x}|v^*|\varphi(v^*) \\ &\leq m_{1x}(1 + 1/\sqrt{2\pi e}) \leq \max\{m_{0x}, m_{1x}\}(1 + 1/\sqrt{2\pi e}). \end{aligned}$$

We then show $|u|t_x(u) \leq \max\{m_{0x}, m_{1x}\}(1 + 1/\sqrt{2\pi e})$. This result holds immediately when $u \geq 0$. Otherwise by (A.10),

$$\begin{aligned} |u|t_x(u) &= m_{0x}\Phi(v^{**}) - m_{0x}\Phi(u) + m_{0x}|v^{**}|\varphi(v^{**}) \\ &\leq m_{0x}(1 + 1/\sqrt{2\pi e}) \leq \max\{m_{0x}, m_{1x}\}(1 + 1/\sqrt{2\pi e}). \end{aligned}$$

The case when $u < u_0$ is similar.

We now show the **last** inequality. WLOG, suppose $m_{0x} \leq m_{1x}$. Then $u_0 \geq 0$.

For $u \geq u_0$, we have $g(u,v) = m_{0x}\Phi(u) + |v-u| \max\{s_x(u), m_{0x}\varphi(u)\}$.

1. For $v \geq u$, by the Mean Value Theorem, $\Phi(v) = \Phi(u) + (v-u)\cdot\varphi(w)$, where $w \in (u,v)$ and $\varphi(w) < \varphi(u)$. Hence $g_x(v) = m_{0x}\Phi(v) < m_{0x}[\Phi(u) + |v-u|\varphi(u)] \leq g(u,v)$.
2. For $u_0 \leq v < u$, $g_x(v) = m_{0x}\Phi(v) < m_{0x}\Phi(u) < g(u,v)$.
3. For $v < u_0$, $g_x(v) = m_{1x}\Phi(-v) = m_{0x}\Phi(u) + (u-v)\left[\frac{m_{1x}\Phi(-v) - m_{0x}\Phi(u)}{u-v}\right] \leq m_{0x}\Phi(u) + |v-u| s_x(u) \leq g(u,v)$.

For $0 < u < u_0$ (if $u_0 > 0$), $g(u,v) = m_{1x}\Phi(-u) + |v-u| \max\{s_x(u), t_x(u)\}$.

1. For $v \leq u$, $g_x(v) = m_{1x}\Phi(-v) = m_{1x}\left[\Phi(-u) + (u-v)\left[\frac{\Phi(-v) - \Phi(-u)}{u-v}\right]\right] \leq m_{1x}\Phi(-u) + |v-u| t_x(u) \leq g(u,v)$.
2. For $u < v \leq u_0$, $g_x(v) = m_{1x}\Phi(-v) < m_{1x}\Phi(-u) = g_x(u) \leq g(u,v)$.
3. For $v > u_0$, $g_x(v) = m_{0x}\Phi(v) = m_{1x}\Phi(-u) + (v-u)\left[\frac{m_{0x}\Phi(v) - m_{1x}\Phi(-u)}{v-u}\right] \leq m_{1x}\Phi(-u) + |v-u| s_x(u) \leq g(u,v)$.

For $u \leq 0$, we can show $g_x(\cdot) \leq g(u,\cdot)$ similar to the case when $u \geq u_0$.

□

Lemma 7. *Define* $\Delta\colon (-\infty,+\infty) \times (0,+\infty] \to [0,+\infty)$ *by*

$$\Delta(\mu,\beta) := \left[\mathbb{E}\left(X^+\right) - (\mathbb{E}X)^+ | X \sim \mathcal{N}(\mu, 1/\beta)\right].$$

Then

$$\Delta(\mu,\beta) = \begin{cases} \mu\Phi\left(\mu\sqrt{\beta}\right) + \sqrt{\beta}^{-1}\varphi\left(\mu\sqrt{\beta}\right), & \text{if } \mu \leq 0; \\ \mu\left[\Phi\left(\mu\sqrt{\beta}\right) - 1\right] + \sqrt{\beta}^{-1}\varphi\left(\mu\sqrt{\beta}\right). & \text{if } \mu > 0. \end{cases}$$

For any fixed $\beta > 0$, $\Delta(\cdot,\beta)$ *has the following properties:*

- *it is strictly increasing on* $(-\infty, 0]$ *and strictly decreasing on* $[0, +\infty)$*;*
- *it reaches its maximum at* $\mu = 0$*, with maximum value* $1/\sqrt{2\pi\beta}$*;*
- *it converges to* 0 *as* $\mu \to +\infty$ *and* $\mu \to -\infty$*.*

Proof. For any fixed $\beta > 0$ and $\mu \in \mathbb{R}$, suppose $X \sim \mathcal{N}(\mu, 1/\beta)$, then by Clark (1961),

$$\mathbb{E}\left(X^+\right) = \mu\Phi\left(\mu\sqrt{\beta}\right) + \sqrt{\beta}^{-1}\varphi\left(\mu\sqrt{\beta}\right) \tag{A.11}$$

where Φ and φ are the standard normal cdf and pdf respectively. Since $\Phi'(x) = \varphi(x)$, $\varphi'(x) = -x\varphi(x)$, it follows that

$$[\mathbb{E}\left(X^+\right)]'_\mu = \Phi\left(\mu\sqrt{\beta}\right) > 0,$$

which indicates that $E(X^+)$ is a strictly increasing function of μ. Similarly,

$$\mathbb{E}\left(X^-\right) = \mathbb{E}\left(X^+\right) - \mu = \mu\left[\Phi\left(\mu\sqrt{\beta}\right) - 1\right] + \sqrt{\beta}^{-1}\varphi\left(\mu\sqrt{\beta}\right), \tag{A.12}$$

$$[\mathbb{E}\left(X^-\right)]'_\mu = \Phi\left(\mu\sqrt{\beta}\right) - 1 < 0,$$

hence $E\left(X^-\right)$ is a strictly decreasing function of μ.

Now consider the objective function

$$\begin{aligned}\Delta(\mu,\beta) &= \mathbb{E}\left(X^+\right) - \left[\mathbb{E}\left(X^+\right) - \mathbb{E}\left(X^-\right)\right]^+ \\ &= \begin{cases} \mathbb{E}\left(X^+\right) = \mathbb{E}\left(X^-\right), & \text{if } \mu = 0; \\ \mathbb{E}\left(X^+\right), & \text{if } \mathbb{E}\left(X^+\right) < \mathbb{E}\left(X^-\right), \text{ i.e., } \mu < 0; \\ \mathbb{E}\left(X^-\right), & \text{if } \mathbb{E}\left(X^+\right) > \mathbb{E}\left(X^-\right), \text{ i.e., } \mu > 0. \end{cases}\end{aligned}$$

It follows that $\Delta(\cdot, \beta)$ is strictly increasing on $(-\infty, 0]$, strictly decreasing on $[0, +\infty)$, and maximized at $\mu = 0$, where

$$\Delta(0,\beta) = \left[\mathbb{E}\left(X^+\right)|X \sim \mathcal{N}(0, 1/\beta)\right] = 1/\sqrt{\beta} \cdot \left[\mathbb{E}\left(X^+\right)|X \sim \mathcal{N}(0,1)\right] = 1/\sqrt{2\pi\beta}.$$

Finally, since for a standard normal r.v. $Z \in L_1$, we have $0 = \lim_{x\to+\infty} x\mathbb{P}(Z > x) = \lim_{x\to-\infty} x\mathbb{P}(Z \le x) = \lim_{x\to+\infty} x(1-\Phi(x)) = \lim_{x\to-\infty} x\Phi(x)$. Thus by (A.11) and (A.12),

$$\lim_{\mu\to-\infty} \Delta(\mu,\beta) = \lim_{\mu\to-\infty} \left[\mathbb{E}\left(X^+\right)|X \sim \mathcal{N}(\mu, 1/\beta)\right] = \sqrt{\beta}^{-1}\left[\lim_{x\to-\infty} x\Phi(x) + \varphi(-\infty)\right] = 0;$$

$$\lim_{\mu\to+\infty} \Delta(\mu,\beta) = \lim_{\mu\to+\infty} \left[\mathbb{E}\left(X^-\right)|X \sim \mathcal{N}(\mu, 1/\beta)\right] = \sqrt{\beta}^{-1}\left[\lim_{x\to+\infty} x(\Phi(x)-1) + \varphi(+\infty)\right]$$
$$= 0.$$

□

A.12.2 0-1 Terminal Payoff

Payoff Condition 3

Denote $\rho = d_x - \mu$. Applying Lemma 6, we have

$$\begin{aligned}
&\mathbb{E}\left[h_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon)\right] - h_x(\mu,\beta) \\
&= \mathbb{E}\left[g_x\left(\sqrt{\beta+\beta_x^\varepsilon}(\rho - \tilde{\sigma}_x(\beta)Z)\right)\right] - g_x\left(\sqrt{\beta}\rho\right) \\
&\le \mathbb{E}\left[g\left(\sqrt{\beta}\rho, \sqrt{\beta+\beta_x^\varepsilon}(\rho - \tilde{\sigma}_x(\beta)Z)\right)\right] - g_x\left(\sqrt{\beta}\rho\right) \\
&= \mathbb{E}\left[\left|\left(\sqrt{\beta+\beta_x^\varepsilon} - \sqrt{\beta}\right)\rho - \sqrt{\beta+\beta_x^\varepsilon}\cdot\tilde{\sigma}_x(\beta)Z\right|\right] \cdot s\left(\sqrt{\beta}\rho\right) \\
&\le \left[\left(\sqrt{1+\beta_x^\varepsilon/\beta} - 1\right)\sqrt{\beta}|\rho| + \sqrt{\beta_x^\varepsilon/\beta}\cdot\mathbb{E}|Z|\right] \cdot s\left(\sqrt{\beta}\rho\right) \\
&= \left(\sqrt{1+\beta_x^\varepsilon/\beta} - 1\right)\cdot\sqrt{\beta}|\rho| s\left(\sqrt{\beta}\rho\right) + \sqrt{\beta_x^\varepsilon/\beta}\cdot\mathbb{E}|Z| s\left(\sqrt{\beta}\rho\right) \\
&\le \max\{m_{0x}, m_{1x}\}\left[\left(\sqrt{1+\beta_x^\varepsilon/\beta} - 1\right)\left(1 + 1/\sqrt{2\pi e}\right) + \pi^{-1}\sqrt{\beta_x^\varepsilon/\beta}\right] \\
&= \widetilde{H}_x(\mu,\beta).
\end{aligned}$$

Now

$$\lim_{n\to\infty}\left[\sup_{(\mu,\beta)\in PS(x;n)}\widetilde{H}_x(\mu,\beta)\right]$$
$$=\max\{m_{0x},m_{1x}\}\lim_{n\to\infty}\left[\left(\sqrt{1+1/n}-1\right)\left(1+1/\sqrt{2\pi e}\right)+1/\left(\sqrt{n}\pi\right)\right]=0,$$

where $PS(x;n)=\{(\mu,\beta)\in\Lambda:\beta\geq n\cdot\beta_x^{\varepsilon}\}$. Thus Payoff Condition 3 holds. Notice that the expression inside the brackets is a continuous, strictly decreasing function of n. Simple algebra then yields

$$N_x=\min\left\{n:\max\{m_{0x},m_{1x}\}\left[\left(\sqrt{1+1/n}-1\right)\left(1+1/\sqrt{2\pi e}\right)+1/\left(\sqrt{n}\pi\right)\right]\leq c_x\right\}$$
$$=\lceil\frac{A_2+\sqrt{A_2^2-4A_1A_3}}{2A_1}\rceil.$$

Special Condition 1

We write

$$\begin{aligned}H_x(\mu,\beta)=m_{0x}\cdot\mathbf{1}_{\left\{p_x(\mu,\beta)\leq\frac{m_{0x}}{m_{0x}+m_{1x}}\right\}}+m_{1x}\cdot\mathbf{1}_{\left\{p_x(\mu,\beta)\geq\frac{m_{0x}}{m_{0x}+m_{1x}}\right\}}\\-\max\left\{m_{0x}\left[1-p_x(\mu,\beta)\right],m_{1x}\cdot p_x(\mu,\beta)\right\}.\end{aligned}\tag{A.13}$$

For any fixed β, $p_x(\mu,\beta)=1-\Phi\left(\sqrt{\beta}(d_x-\mu)\right)$ converges to 1 as $\mu\to+\infty$, and converges to 0 as $\mu\to-\infty$. In both cases, we have $H_x(\mu,\beta)\to 0$ and thus Special Condition 1 holds.

Special Condition 2

Fix $\beta>0$. Let $u_0:=\Phi^{-1}\left[\frac{m_{1x}}{m_{0x}+m_{1x}}\right]$. Then

$$p_x(\mu,\beta)\leq\frac{m_{0x}}{m_{0x}+m_{1x}}\Longleftrightarrow\sqrt{\beta}(d_x-\mu)\geq u_0.$$

Case I $c_x < 2\max\{m_{0x}, m_{1x}\}$.

We know that for any $\alpha \in (0,1)$, $I_\alpha = [-z_\alpha, z_\alpha] := [-\Phi^{-1}(1-\alpha/2), \Phi^{-1}(1-\alpha/2)]$ is the $100(1-\alpha)\%$ confidence interval for the standard normal distribution. By Proposition 3 and (A.13),

$$\begin{aligned}
&\mathbb{E}\left[V_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon)\right] \le \mathbb{E}\left[H_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon)\right] \\
&= (1-\alpha)\mathbb{E}\left[H_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon) \mid Z \in I_\alpha\right] + \alpha\mathbb{E}\left[H_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon) \mid Z \in I_\alpha^c\right] \\
&\le (1-\alpha)\mathbb{E}\left[H_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon) \mid Z \in I_\alpha\right] + \alpha \cdot \max\{m_{0x}, m_{1x}\}.
\end{aligned} \tag{A.14}$$

Pick $\alpha = c_x / \left[2\max\{m_{0x}, m_{1x}\}\right]$.

When $\mu \le d_x - \tilde{\sigma}_x(\beta)z_\alpha - u_0/\sqrt{\beta + \beta_x^\varepsilon}$, we have $\sqrt{\beta + \beta_x^\varepsilon}\,(d_x - \mu - \tilde{\sigma}_x(\beta)Z) \ge u_0$ for all $Z \in I_\alpha$, hence

$$\begin{aligned}
&\mathbb{E}\left[H_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon) \mid Z \in I_\alpha\right] \\
&= m_{0x}\mathbb{E}\left[1 - \Phi\left(\sqrt{\beta + \beta_x^\varepsilon}(d_x - \mu - \tilde{\sigma}_x(\beta)Z)\right) \mid Z \in I_\alpha\right] \\
&\le m_{0x}\left[1 - \Phi\left(\sqrt{\beta + \beta_x^\varepsilon}(d_x - \mu - \tilde{\sigma}_x(\beta)z_\alpha)\right)\right],
\end{aligned}$$

which goes to 0 as $\mu \to -\infty$.

Moreover, Special Condition 1 indicates that $\mathscr{R}_x(\mu, \beta) \to -c_x$ as $\mu \to -\infty$.

(2.16) and (A.14) then yields

$$\begin{aligned}
\limsup_{\mu \to +\infty} L_x(\mu, \beta, V_x) &\le \lim_{\mu \to +\infty} \mathscr{R}_x(\mu, \beta) + \limsup_{\mu \to +\infty} \mathbb{E}\left[V_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon)\right] \\
&\le -c_x + c_x/2 < 0.
\end{aligned}$$

We can therefore find some $\underline{\mu}_x(\beta) \le d_x - \tilde{\sigma}_x(\beta)z_\alpha - u_0/\sqrt{\beta + \beta_x^\varepsilon}$ such that for all $\mu \le \underline{\mu}_x(\beta)$, $L_x(\mu, \beta, V_x) < 0$ and hence $V_x(\mu, \beta) = 0$.

Similarly, we can find some $\overline{\mu}_x(\beta) \geq d_x + \tilde{\sigma}_x(\beta) z_\alpha$ such that $V_x(\mu,\beta) = 0$ for all $\mu \geq \underline{\mu}_x(\beta)$.

Case II $c_x \geq 2\max\{m_{0x}, m_{1x}\}$. Then

$$\mathbb{E}\left[V\left(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon\right)\right] \leq \mathbb{E}\left[H_x\left(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^\varepsilon\right)\right] \leq \max\{m_{0x}, m_{1x}\} \leq c_x/2.$$

It follows that

$$\limsup_{\mu \to +\infty} L_x(\mu, \beta, V_x) \leq -c_x/2 < 0.$$

Hence we can also find $\overline{\mu}_x(\beta)$ and $\underline{\mu}_x(\beta)$ satisfying Special Condition 2.

A.12.3 Linear Terminal payoff

Special Condition 1

Applying Lemma 7,

$$\begin{aligned} H_x(\mu,\beta) &= \mathbb{E}\left[m_{0x}(d_x - \theta_x)^+ + m_{1x}(\theta_x - d_x)^+ \,\middle|\, \theta_x \sim \mathcal{N}(\mu, 1/\beta) \right] \\ &\quad - \left[m_{0x}(d_x - \mu)^+ + m_{1x}(\mu - d_x)^+ \right] \\ &= m_{0x} \cdot \Delta(d_x - \mu, \beta) + m_{1x} \cdot \Delta(\mu - d_x, \beta). \end{aligned} \tag{A.15}$$

Hence for any fixed $\beta > 0$, $H_x(\mu,\beta) \to 0$ as $\mu \to +\infty$ and $\mu \to -\infty$.

Payoff Condition 3

Notice that

$$H_x(\cdot,\beta) \leq (m_{0x} + m_{1x})/\sqrt{2\pi\beta} = \widetilde{H}_x(\cdot,\beta), \tag{A.16}$$

hence the first inequality in Payoff Condition 3 holds. Now in the sub-problem with single alternative x, $\beta_{nx} = \beta_{0x} + n \cdot \beta_x^\varepsilon$, hence $PS(x;n) = \{(\mu,\beta) \in \Lambda : \beta \geq n \cdot \beta_x^\varepsilon\}$. It

follows that

$$\lim_{n\to\infty}\left[\sup_{(\mu,\beta)\in PS(x;n)}\widetilde{H}_x(\mu,\beta)\right]=\lim_{n\to\infty}\left[\frac{m_{0x}+m_{1x}}{\sqrt{2\pi n\beta_x^{\varepsilon}}}\right]=0.$$

Thus the second equation in Payoff Condition 3 holds and

$$N_x=\min\left\{n\geq 0:\frac{m_{0x}+m_{1x}}{\sqrt{2\pi n\beta_x^{\varepsilon}}}\leq c_x\right\}=\lceil\frac{(m_{0x}+m_{1x})^2}{2\pi c_x^2\beta_x^{\varepsilon}}\rceil. \tag{A.17}$$

Special Condition 2

Fix $\beta>0$. Let $\rho=\mu-d_x$.

Case I $c_x\sqrt{\beta+\beta_x^{\varepsilon}}/(m_{0x}+m_{1x})<1$.

We know that for any $\alpha\in(0,1)$, $I_\alpha=[-z_\alpha,z_\alpha]:=[-\Phi^{-1}(1-\alpha/2),\Phi^{-1}(1-\alpha/2)]$ is the $100(1-\alpha)\%$ confidence interval for the standard normal distribution. By Proposition 3 and (A.15), (A.16),

$$\begin{aligned}
&\mathbb{E}[V_x(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})]\leq\mathbb{E}[H_x(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})]\\
&=(1-\alpha)\mathbb{E}[H_x(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})\mid Z\in I_\alpha]+\alpha\mathbb{E}[H_x(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})\mid Z\in I_\alpha^c]\\
&\leq(1-\alpha)\mathbb{E}[H_x(\mu+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})\mid Z\in I_\alpha]+\alpha\widetilde{H}_x(\cdot,\beta+\beta_x^{\varepsilon})\\
&=(1-\alpha)\mathbb{E}[m_{0x}\Delta(-\rho-\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})+m_{1x}\Delta(\rho+\tilde{\sigma}_x(\beta)Z,\beta+\beta_x^{\varepsilon})\mid Z\in I_\alpha]\\
&\quad+\alpha\widetilde{H}_x(\cdot,\beta+\beta_x^{\varepsilon}).
\end{aligned} \tag{A.18}$$

where $\widetilde{H}_x(\cdot,\beta)=(m_{0x}+m_{1x})/\sqrt{2\pi\beta}$.

Pick $\alpha=c_x\sqrt{\beta+\beta_x^{\varepsilon}}/(m_{0x}+m_{1x})$. Then $\alpha\widetilde{H}_x(\cdot,\beta+\beta_x^{\varepsilon})=c_x/\sqrt{2\pi}$.

When $\rho\geq\tilde{\sigma}_x(\beta)z_\alpha$, i.e., $\mu\geq d_x+\tilde{\sigma}_x(\beta)z_\alpha$, since $\Delta(\cdot,\beta+\beta_x^{\varepsilon})$ is increasing at

$(-\infty, 0]$ and decreasing at $[0, +\infty)$, we know

$$\begin{aligned}
&\mathbb{E}\left[m_{0x}\Delta(-\rho - \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^{\varepsilon}) + m_{1x}\Delta(\rho + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^{\varepsilon}) \mid Z \in I_\alpha\right] \\
&\leq m_{0x}\Delta(-\rho + \tilde{\sigma}_x(\beta)z_\alpha, \beta + \beta_x^{\varepsilon}) + m_{1x}\Delta(\rho - \tilde{\sigma}_x(\beta)z_\alpha, \beta + \beta_x^{\varepsilon}),
\end{aligned}$$

which goes to 0 as $\rho \to +\infty$, i.e., $\mu \to +\infty$.

Moreover, (A.8) shows that $\mathscr{R}_x(\mu, \beta) \to -c_x$ as $\rho \to +\infty$, i.e., $\mu \to +\infty$.

(2.16) and (A.18) then yield

$$\begin{aligned}
\limsup_{\mu \to +\infty} L_x(\mu, \beta, V_x) &\leq \lim_{\mu \to +\infty} \mathscr{R}_x(\mu, \beta) + \limsup_{\mu \to +\infty} \mathbb{E}\left[V_x(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^{\varepsilon})\right] \\
&\leq -c_x + c_x/\sqrt{2\pi} < 0.
\end{aligned}$$

It follows that we can find $\overline{\mu}_x(\beta) \geq d_x + \tilde{\sigma}_x(\beta)z_\alpha$ such that for all $\mu \geq \overline{\mu}_x(\beta)$, $L_x(\mu, \beta, V_x) < 0$ and hence $V_x(\mu, \beta) = 0$.

Similarly, we can find some $\underline{\mu}_x(\beta) \leq d_x - \tilde{\sigma}_x(\beta)z_\alpha$ such that $V_x(\mu, \beta) = 0$ for all $\mu \leq \underline{\mu}_x(\beta)$.

Case II $c_x\sqrt{\beta + \beta_x^{\varepsilon}}/(m_{0x} + m_{1x}) \geq 1$. Then

$$\mathbb{E}\left[V(\mu + \tilde{\sigma}_x(\beta)Z, \beta + \beta_x^{\varepsilon})\right] \leq \widetilde{H}_x(\cdot, \beta + \beta_x^{\varepsilon}) = \frac{m_{0x} + m_{1x}}{\sqrt{2\pi(\beta + \beta_x^{\varepsilon})}} \leq c_x/\sqrt{2\pi}.$$

It follows that

$$\limsup_{\mu \to +\infty} L_x(\mu, \beta, V_x) \leq -c_x + c_x/\sqrt{2\pi} < 0.$$

Hence we can also find $\overline{\mu}_x(\beta)$ and $\underline{\mu}_x(\beta)$ satisfying Special Condition 2.

APPENDIX B

APPENDIX OF CHAPTER 3

B.1 Mathematical Proofs

B.1.1 Proof of Lemma 1

Proof. From the definition of the function $h(\cdot,\cdot)$ (just after (3.14)), for any vectors a, b and b' with $b(i) \leq b'(i)$ for all i, we have $h(a,b) \leq h(a,b')$. From (3.14), we have $V_n(\vec{x},A,\beta) = h(\mu_n(A), \widetilde{\sigma}_n(\vec{x},A,\beta))$, where for all $x' \in A$, the element of $\widetilde{\sigma}_n(\vec{x},A,\beta)$ corresponding to x' is $\left[\Sigma_n\left(x',x^{(1)}\right) - \Sigma_n\left(x',x^{(2)}\right)\right]/B$, where B is the denominator (in the lower equation for pairs) in (3.11). Hence we only need to show that B is a decreasing function of $\rho\left(x^{(1)},x^{(2)}\right)$. The result follows immediately by observing $\Lambda\left(x^{(1)},x^{(2)}\right) = \rho\left(x^{(1)},x^{(2)}\right)\left[\Lambda(x^{(1)},x^{(1)})\Lambda(x^{(2)},x^{(2)})\right]^{1/2}$. □

B.1.2 Preliminary Results for the Convergence Proofs

We first state and prove several lemmas needed to prove the convergence results stated in Section 3.5. These lemmas all assume Assumptions 1 and 2. Condition 4 is assumed only in the proof of Theorem 1.

Lemma 8. *There exist random variables $\mu_\infty \in \mathbb{R}^k$ and $\Sigma_\infty \in \Sigma_+^k$ (the space of $k \times k$ positive semi-definite matrices), such that μ_n converges to μ_∞, and Σ_n converges to Σ_∞ almost surely.*

Proof. Let (μ_n, Σ_n) and $M_n = \left(\mu_n, \Sigma_n + \mu_n \mu_n^T\right)$. We can write the components of M_n as the conditional expectation of an integrable random variable with respect to $\mathscr{X}_n$, $\mathscr{Y}_n$ by $\mu_n = \mathbb{E}_n \theta, \Sigma_n + \mu_n \mu_n^T = \mathbb{E}_n \theta \theta^T$. This implies that M_n is a uniformly integrable martingale and hence converges almost surely (Doob's second martingale convergence theorem, e.g. see Oksendal 2003, App. C). Because (μ_n, Σ_n) is a continuous transformation of M_n, it also converges almost surely to some random variable $(\mu_\infty, \Sigma_\infty)$. □

Lemma 9. $\Sigma_n(x', x) = \Sigma_0(x', x) - \Sigma_0(x, \mathscr{X}_n)[S_n]^{-1}\Sigma_0(\mathscr{X}_n, x')$.

Proof. Let $i_{x'}$ be the index of x' in $\mathscr{X}_{n,x}$. (If x' appears more than once, let it be the index of one occurance.) Let $e_{x'}$ be a column vector with length $|\mathscr{X}_n| + 1$ that has value 1 at entry $i_{x'}$ and 0 elsewhere. Let e_x be defined similarly. Using (3.5), (3.7) and the symmetry of $[S_n]^{-1}$, we then have

$$\begin{aligned}
\Sigma_n(x', x) &= e_{x'}^T \Sigma_n(\mathscr{X}_{n,x}, \mathscr{X}_{n,x}) e_x = e_{x'}^T \left(I_{|\mathscr{X}_n|+1} - K_n(x)\left[I_{|\mathscr{X}_n|}, \vec{0}\right]\right) \Sigma_0(\mathscr{X}_{n,x}, \mathscr{X}_{n,x}) e_x \\
&= e_{x'}^T \Sigma_0(\mathscr{X}_{n,x}, \mathscr{X}_{n,x}) e_x - e_{x'}^T K_n(x)\left[I_{|\mathscr{X}_n|}, \vec{0}\right] \Sigma_0(\mathscr{X}_{n,x}, \mathscr{X}_{n,x}) e_x \\
&= \Sigma_0(x', x) - e_{x'}^T \Sigma_0(\mathscr{X}_{n,x}, \mathscr{X}_{n,x})\left[I_{|\mathscr{X}_n|}, \vec{0}\right]^T [S_n]^{-1}\left[I_{|\mathscr{X}_n|}, \vec{0}\right] \Sigma_0(\mathscr{X}_{n,x}, \mathscr{X}_{n,x}) e_x \\
&= \Sigma_0(x', x) - \Sigma_0\left(x', \mathscr{X}_n\right)[S_n]^{-1}\Sigma_0(\mathscr{X}_n, x) \\
&= \Sigma_0(x', x) - \Sigma_0(x, \mathscr{X}_n)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n, x'\right).
\end{aligned}$$

□

Lemma 10. $\Sigma_{n+1}(x, x) \leq \Sigma_n(x, x)$ *for all* x.

Proof. Using standard results from Bayesian linear regression (e.g., Gelman et al. 2004, Sec. 14.6) and the Sherman-Morrison-Woodbury formula (e.g., Rasmussen and Williams 2006, App. A.3), the posterior variance Σ_{n+1} of θ can be computed recursively by

$$\Sigma_{n+1} = \Sigma_n - \Sigma_n X_{n+1}\left[X_{n+1}^T(\Lambda + \Sigma_n) X_{n+1}\right]^{-1} X_{n+1}^T \Sigma_n,$$

where

$$X_{n+1} = \begin{cases} e_x, & \text{if } \vec{x}_{n+1} = x, \\ \left[e_{x^{(1)}}, e_{x^{(2)}}\right], & \text{if } \vec{x}_{n+1} = \left(x^{(1)}, x^{(2)}\right), \end{cases}$$

and e_x is a $k \times 1$ vector with a value of 1 at the entry for x and 0 elsewhere.

It is clear that $\Lambda + \Sigma_n$ and $X_{n+1}^T (\Lambda + \Sigma_n) X_{n+1}$ are positive definite. Hence for any x,

$$\Sigma_{n+1}(x,x) = \Sigma_n(x,x) - e_x^T \Sigma_n X_{n+1} \left[X_{n+1}^T (\Lambda + \Sigma_n) X_{n+1}\right]^{-1} X_{n+1}^T \Sigma_n e_x \leq \Sigma_n(x,x).$$

□

Lemma 11. *For all x and $x^{(1)} \neq x^{(2)}$, $P\left(x^{(1)}, x^{(2)}\right) = \Lambda\left(x^{(1)}, x^{(1)}\right) + \Lambda\left(x^{(2)}, x^{(2)}\right) - 2\Lambda\left(x^{(1)}, x^{(2)}\right) > 0$ and*

$$\nu_n^{\mathrm{KG}_\beta}(x) \leq \frac{1}{c(x)} \sqrt{\frac{2 \max_{x'} \Sigma_0(x',x') \Sigma_n(x,x)}{\beta \pi \Lambda(x,x)}},$$

$$\nu_n^{\mathrm{KG}_\beta}\left(x^{(1)}, x^{(2)}\right) \leq \frac{1}{c\left(x^{(1)}, x^{(2)}\right)} \sqrt{\frac{2 \max_{x'} \Sigma_0(x',x')}{\beta \pi P\left(x^{(1)}, x^{(2)}\right)}} \left[\sqrt{\Sigma_n\left(x^{(1)}, x^{(1)}\right)} + \sqrt{\Sigma_n\left(x^{(2)}, x^{(2)}\right)}\right].$$

Proof. First, we have

$$\begin{aligned} V_n(\vec{x}, A_n(\vec{x}), \beta) &= \mathbb{E}_n\left[\max \mu_{n+1}(A_n(\vec{x})) \mid \vec{x}_{n+1} = \vec{x}, \beta_{n+1} = \beta\right] - \max \mu_n(A_n(\vec{x})) \\ &= \mathbb{E}\left[\max\left\{\mu_n(A_n(\vec{x})) + \widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta) Z\right\}\right] - \max \mu_n(A_n(\vec{x})) \\ &\leq \max \mu_n(A_n(\vec{x})) + \mathbb{E}\left[\max\left\{\widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta) Z\right\}\right] - \max \mu_n(A_n(\vec{x})) \\ &= \mathbb{E}\left[\max\left\{\widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta) Z\right\}\right] \leq \mathbb{E}\left[\max\left\{|\widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)| \cdot |Z|\right\}\right] \\ &= \mathbb{E}|Z| \cdot \max\left\{|\widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)|\right\} = \sqrt{2/\pi} \cdot \max\left\{|\widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)|\right\} \\ &= \sqrt{2/\pi} \cdot \max_{j=1,2,\ldots,|A_n(\vec{x})|} |e_j^T \widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)|, \end{aligned}$$

where e_j is a $|A_n(\vec{x})| \times 1$ vector with 1 at entry j and 0 elsewhere.

We now derive an upper bound on $e_j^T \widetilde{\sigma}_n(\vec{x}, A_n(\vec{x}), \beta)$. First, $P\left(x^{(1)}, x^{(2)}\right) = [e_{x^{(1)}} - e_{x^{(2)}}]^T \Lambda [e_{x^{(1)}} - e_{x^{(2)}}] > 0$ because Λ is positive definite by Assumption 2, where $e_{x^{(j)}}$

is a vector with 1 at entry $x^{(j)}$ and 0 elsewhere. Similarly, we have $\Sigma_n\left(x^{(1)},x^{(1)}\right)+\Sigma_n\left(x^{(2)},x^{(2)}\right)-2\Sigma_n\left(x^{(1)},x^{(2)}\right)\geq 0$ because Σ_n is positive semi-definite. Now applying (3.11) and Lemma 10, for $\vec{x}=x$ we have

$$|e_j^T\widetilde{\sigma}_n(x,A_n(x),\beta)| = \frac{|e_j^T\Sigma_n(A_n(x),x)|}{\sqrt{\beta^{-1}\Lambda(x,x)+\Sigma_n(x,x)}} = \frac{|\Sigma_n\left(A_n^{(j)}(x),x\right)|}{\sqrt{\beta^{-1}\Lambda(x,x)+\Sigma_n(x,x)}}$$
$$\leq \sqrt{\frac{\Sigma_n\left(A_n^{(j)}(x),A_n^{(j)}(x)\right)\Sigma_n(x,x)}{\beta^{-1}\Lambda(x,x)}} \leq \sqrt{\frac{\Sigma_0\left(A_n^{(j)}(x),A_n^{(j)}(x)\right)\Sigma_n(x,x)}{\beta^{-1}\Lambda(x,x)}},$$

where $A_n^{(j)}(\vec{x})$ is the jth component of $A_n(\vec{x})$. Similarly for $\vec{x}=\left(x^{(1)},x^{(2)}\right)$ we have

$$|e_j^T\widetilde{\sigma}_n(\vec{x},A_n(\vec{x}),\beta)| = \frac{|e_j^T\Sigma_n\left(A_n(\vec{x}),x^{(1)}\right)-e_j^T\Sigma_n\left(A_n(\vec{x}),x^{(2)}\right)|}{\sqrt{\beta^{-1}P\left(x^{(1)},x^{(2)}\right)+\Sigma_n\left(x^{(1)},x^{(1)}\right)+\Sigma_n\left(x^{(2)},x^{(2)}\right)-2\Sigma_n\left(x^{(1)},x^{(2)}\right)}}$$
$$\leq \frac{|\Sigma_n\left(A_n^{(j)}(\vec{x}),x^{(1)}\right)|+|\Sigma_n\left(A_n^{(j)}(\vec{x}),x^{(2)}\right)|}{\sqrt{\beta^{-1}P\left(x^{(1)},x^{(2)}\right)}}$$
$$\leq \sqrt{\frac{\Sigma_n\left(A_n^{(j)}(\vec{x}),A_n^{(j)}(\vec{x})\right)}{\beta^{-1}P\left(x^{(1)},x^{(2)}\right)}}\left[\sqrt{\Sigma_n\left(x^{(1)},x^{(1)}\right)}+\sqrt{\Sigma_n\left(x^{(2)},x^{(2)}\right)}\right]$$
$$\leq \sqrt{\frac{\Sigma_0\left(A_n^{(j)}(\vec{x}),A_n^{(j)}(\vec{x})\right)}{\beta^{-1}P\left(x^{(1)},x^{(2)}\right)}}\left[\sqrt{\Sigma_n\left(x^{(1)},x^{(1)}\right)}+\sqrt{\Sigma_n\left(x^{(2)},x^{(2)}\right)}\right].$$

The claimed bounds in the lemma for $\nu_n^{\mathrm{KG}_\beta}(x)$ and for $\nu_n^{\mathrm{KG}_\beta}\left(x^{(1)},x^{(2)}\right)$ follow directly.

□

Lemma 12. *Under the allocation rule $x_1=x_2=\cdots=x_n=x$, $\Sigma_n(x,x)$ decreases to 0 as $n\to+\infty$. Under the allocation rule $\vec{x}_1=\vec{x}_2=\cdots=\vec{x}_n=\left(x^{(1)},x^{(2)}\right)$, $\Sigma_n\left(x^{(1)},x^{(1)}\right)$ and $\Sigma_n\left(x^{(2)},x^{(2)}\right)$ decrease to 0 as $n\to+\infty$.*

Proof. Lemma 10 shows that $\Sigma_n(x,x)$ is a decreasing sequence bounded below by zero, for all x. It suffices to show that the limit is 0 under these two cases.

First consider the case when $x_1 = x_2 = \cdots = x_n = x_0$. Note $\Sigma_0(\mathscr{X}_n, \mathscr{X}_n) = \Sigma_0(x_0, x_0)ee^T$, where e is an $n \times 1$ vector with n entries of 1. By Lemma 9 and the Sherman-Morrison-Woodbury formula, for any x and x',

$$\begin{aligned}
\Sigma_n(x,x') &= \Sigma_0(x,x') - \Sigma_0(x,\mathscr{X}_n)[S_n]^{-1}\Sigma_0(\mathscr{X}_n,x') \\
&= \Sigma_0(x,x') - \Sigma_0(x,x_0)\Sigma_0(x',x_0)e^T\left[\Sigma_0(x_0,x_0)ee^T + \Lambda(x_0,x_0)I_n\right]^{-1}e \\
&= \Sigma_0(x,x') - \frac{\Sigma_0(x,x_0)\Sigma_0(x',x_0)}{\Lambda(x_0,x_0)}e^T\left[I_n - \frac{\Sigma_0(x_0,x_0)}{n\Sigma_0(x_0,x_0)+\Lambda(x_0,x_0)}ee^T\right]e \\
&= \Sigma_0(x,x') - \frac{n\Sigma_0(x,x_0)\Sigma_0(x',x_0)}{n\Sigma_0(x_0,x_0)+\Lambda(x_0,x_0)}.
\end{aligned}$$

Specifically,

$$\Sigma_n(x_0,x_0) = \Sigma_0(x_0,x_0)\left[1 - \frac{n\Sigma_0(x_0,x_0)}{n\Sigma_0(x_0,x_0)+\Lambda(x_0,x_0)}\right] \to 0 \text{ as } n \to +\infty.$$

Next consider the case when $\vec{x}_1 = \vec{x}_2 = \cdots = \vec{x}_n = \left(x_0^{(1)}, x_0^{(2)}\right)$. Let

$$D_1 = \begin{bmatrix} \Sigma_0\left(x_0^{(1)},x_0^{(1)}\right) & \Sigma_0\left(x_0^{(1)},x_0^{(2)}\right) \\ \Sigma_0\left(x_0^{(1)},x_0^{(2)}\right) & \Sigma_0\left(x_0^{(2)},x_0^{(2)}\right) \end{bmatrix}, \qquad D_2 = \begin{bmatrix} \Lambda\left(x_0^{(1)},x_0^{(1)}\right) & \Lambda\left(x_0^{(1)},x_0^{(2)}\right) \\ \Lambda\left(x_0^{(1)},x_0^{(2)}\right) & \Lambda\left(x_0^{(2)},x_0^{(2)}\right) \end{bmatrix}.$$

Let $U = [I_2, I_2, \ldots, I_2]^T$ be a $2n \times 2$ matrix with n I_2-blocks. Let $u = \left[\Sigma_0\left(x,x_0^{(1)}\right), \Sigma_0\left(x,x_0^{(2)}\right)\right]^T$ and $v = \left[\Sigma_0\left(x',x_0^{(1)}\right), \Sigma_0\left(x',x_0^{(2)}\right)\right]^T$ be two 2×1 vectors. Then $\Sigma_0(\mathscr{X}_n, \mathscr{X}_n) = UD_1U^T$, and Γ_n is a block diagonal matrix with n blocks, with each block equal to D_2. Similar to the above argument we have

$$\begin{aligned}
\Sigma_n(x,x') &= \Sigma_0(x,x') - \Sigma_0(x,\mathscr{X}_n)[S_n]^{-1}\Sigma_0(\mathscr{X}_n,x') \\
&= \Sigma_0(x,x') - \Sigma_0(x,\mathscr{X}_n)\left[UD_1U^T + \Gamma_n\right]^{-1}\Sigma_0(\mathscr{X}_n,x') \\
&= \Sigma_0(x,x') - \Sigma_0(x,\mathscr{X}_n)\left[\Gamma_n^{-1} - \Gamma_n^{-1}U\left(D_1^{-1} + U^T\Gamma_n^{-1}U\right)^{-1}U^T\Gamma_n^{-1}\right]\Sigma_0(\mathscr{X}_n,x') \\
&= \Sigma_0(x,x') - n\left[u^TD_2^{-1}v - nu^TD_2^{-1}\left[D_1^{-1} + nD_2^{-1}\right]^{-1}D_2^{-1}v\right] \\
&= \Sigma_0(x,x') - nu^T(nD_1 + D_2)^{-1}v,
\end{aligned}$$

where the last line follows from the previous line by the following computation, which uses the matrix identity $A^{-1}B^{-1} = (BA)^{-1}$ and the Sherman-Morrison-Woodbury formula:

$$\begin{aligned}
&\left[I - nD_2^{-1}(D_1^{-1} + nD_2^{-1})^{-1}\right] D_2^{-1} = \left[I - n(D_1^{-1}D_2 + nI)^{-1}\right] D_2^{-1} \\
&= \left[I - \left(I + n^{-1}D_1^{-1}D_2\right)^{-1}\right] D_2^{-1} = \left[I - (I - D_1^{-1}(nI + D_2D_1^{-1})^{-1}D_2\right] D_2^{-1} \\
&= D_1^{-1}(nI + D_2D_1^{-1})^{-1} = (nD_1 + D_2)^{-1}.
\end{aligned}$$

Simple algebra then yields $\lim_{n\to+\infty} nu^T (nD_1 + D_2)^{-1} v = (d_2 + d_3 - d_4)/d_1$, where

$$\begin{aligned}
d_1 &= \Sigma_0\left(x_0^{(1)}, x_0^{(1)}\right)\Sigma_0\left(x_0^{(2)}, x_0^{(2)}\right) - \left[\Sigma_0\left(x_0^{(1)}, x_0^{(2)}\right)\right]^2, \\
d_2 &= \Sigma_0\left(x_0^{(1)}, x_0^{(1)}\right)\Sigma_0\left(x, x_0^{(2)}\right)\Sigma_0\left(x', x_0^{(2)}\right), \\
d_3 &= \Sigma_0\left(x_0^{(2)}, x_0^{(2)}\right)\Sigma_0\left(x, x_0^{(1)}\right)\Sigma_0\left(x', x_0^{(1)}\right), \\
d_4 &= \Sigma_0\left(x_0^{(1)}, x_0^{(2)}\right)\left[\Sigma_0\left(x, x_0^{(1)}\right)\Sigma_0\left(x', x_0^{(2)}\right) + \Sigma_0\left(x, x_0^{(2)}\right)\Sigma_0\left(x', x_0^{(1)}\right)\right].
\end{aligned}$$

Under Assumption 2, we always have $d_1 > 0$ because Σ_0 is positive definite. Specifically, when $x = x' = x_0^{(i)}$ $(i = 1,2)$, $[d_2 + d_3 - d_4]/d_1 = \Sigma_0\left(x_0^{(i)}, x_0^{(i)}\right)$. Hence $\Sigma_n\left(x_0^{(i)}, x_0^{(i)}\right) \to 0$ as $n \to +\infty$ for $i = 1,2$. □

Lemma 13. *If alternative x is sampled infinitely often, then $\Sigma_n(x,x) \to 0$ and $\nu_n^{\mathrm{KG}_\beta}(x) \to 0$ as $n \to \infty$. If alternative $x' \neq x$ is also sampled infinitely often, then $\Sigma_n(x',x') \to 0$ and $\nu_n^{\mathrm{KG}_\beta}(x,x') \to 0$ as $n \to \infty$.*

Proof. There are k possible decisions in Ξ that involve sampling alternative x, namely, x and (x,x') for $x' \neq x$. Because x is sampled infinitely many times, at least one of these k decisions is chosen infinitely often. Let $\vec{x}$ be one such decision and $\{q_n\}_{n=1}^{\infty}$ be a strictly increasing subsequence of $\mathbb{Z}^+$ such that $\vec{x}_{q_n} = \vec{x}$ for $n = 1,2,\ldots$ Because the ordering of the decision-observation pairs can be changed without altering $\Sigma_n(x,x)$,

and because taking additional observations can only decrease $\Sigma_n(x,x)$ by Lemma 10, we know that an upper bound on $\Sigma_{q_n}(x,x)$ is given by the posterior variance of θ_x at time n under an allocation rule, call it π, that chooses $x_1 = x_2 = \cdots = x_n = \vec{x}$. Call this posterior variance $\Sigma_n^\pi(x,x)$, so we have $\Sigma_{q_n}(x,x) \leq \Sigma_n^\pi(x,x)$. Lemma 12 shows $\lim_{n\to\infty} \Sigma_n^\pi(x,x) = 0$. Hence $\lim_{n\to\infty} \Sigma_{q_n}(x,x) = 0$. Because $\{\Sigma_n(x,x)\}_n$ is a non-negative decreasing sequence, $\lim_{n\to\infty} \Sigma_n(x,x)$ exists and equals 0, due to the uniqueness of the limit. Combining this with Lemma 11 and the non-negativity of the KG_β factors, we have $\lim_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(x) = 0$.

If $x' \neq x$ is also sampled infinitely often, similarly we have $\lim_{n\to\infty} \Sigma_n(x',x') = 0$, and thus $\lim_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(x,x') = 0$ by Lemma 11.

□

Lemma 14. *If* $\liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) = 0$ *for all* $\vec{x} \in \Xi$, *then* $\lim_{n\to\infty} \Sigma_n(x,x) = 0$ *for all* x.

Proof. Consider an arbitrary sample path on which the μ_n converges to μ_∞. Lemma 8 shows that the set of such sample paths is almost sure. We will show that the claim holds on this sample path.

Lemma 10 shows that $\{\Sigma_n(x,x)\}_n$ is a non-negative decreasing sequence and hence $\lim_{n\to\infty} \Sigma_n(x,x)$ exists and is non-negative for any x. We prove the contrapositive of the statement of the lemma. That is, we suppose that $\max_x [\lim_{n\to\infty} \Sigma_n(x,x)] > 0$ and show that $\liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) > 0$ for some $\vec{x} \in \Xi$.

We choose two alternatives on which to focus in our analysis. First, because at least one decision $\vec{x}' \in \Xi$ is chosen by the algorithm infinitely often, Lemma 13 shows that there exists an alternative x' with $\lim_{n\to\infty} \Sigma_n(x',x') = 0$. Second, by our choice of sample path, $\lim_{n\to\infty} \mu_n = \mu_\infty$. Let $x^* = \operatorname{argmax} \mu_\infty$, breaking ties arbitrarily. Then

$\mu_\infty(x^*) \geq \mu_\infty(x)$ for all x. It follows that there exists N large enough and a sequence $\{\varepsilon_n\}$ decreasing to 0 such that $\mu_n(x^*) \geq \mu_n(x) - \varepsilon_n$ for all x for $n \geq N$. If $\lim_{n\to\infty} \Sigma_n(x^*,x^*) > 0$, let $x^{(1)} = x^*$ and $x^{(2)} = x'$; otherwise pick $x^{(1)}$ with $\lim_{n\to\infty} \Sigma_n\left(x^{(1)},x^{(1)}\right) > 0$ and let $x^{(2)} = x^*$. Let $\vec{x} = \left(x^{(1)},x^{(2)}\right)$.

For each $n \geq N$, let

$$a_n^1 = \mu_n\left(x^{(1)}\right), \quad a_n^2 = \mu_n\left(x^{(2)}\right),$$
$$b_n^1 = \frac{\Sigma_n\left(x^{(1)},x^{(1)}\right) - \Sigma_n\left(x^{(1)},x^{(2)}\right)}{\sqrt{\beta^{-1}P + Q_n}}, \quad b_n^2 = \frac{\Sigma_n\left(x^{(2)},x^{(1)}\right) - \Sigma_n\left(x^{(2)},x^{(2)}\right)}{\sqrt{\beta^{-1}P + Q_n}},$$

where P and Q_n are given in (3.12). Then we have the following:

$$\begin{aligned} V_n(\vec{x},A_n(\vec{x}),\beta) &= \mathbb{E}_n\left[\max \mu_{n+1}(A_n(\vec{x})) \mid \vec{x}_{n+1} = \vec{x}, \beta_{n+1} = \beta\right] - \max \mu_n(A_n(\vec{x})) \\ &\geq \mathbb{E}_n\left[\max\left\{\mu_{n+1}\left(x^{(1)}\right), \mu_{n+1}\left(x^{(2)}\right)\right\} \mid \vec{x}_{n+1} = \vec{x}, \beta_{n+1} = \beta\right] \quad \text{(B.1)} \\ &\quad - \max\left\{\mu_n\left(x^{(1)}\right), \mu_n\left(x^{(2)}\right)\right\} - \varepsilon_n \\ &= \mathbb{E}\left[\max\left\{a_n^1 + b_n^1 Z, a_n^2 + b_n^2 Z\right\}\right] - \max\{a_n^1, a_n^2\} - \varepsilon_n \\ &= |b_n^1 - b_n^2| f\left(-\frac{|a_n^1 - a_n^2|}{|b_n^1 - b_n^2|}\right) - \varepsilon_n, \quad \text{(B.2)} \end{aligned}$$

where $f(-s) = \varphi(s) - s\Phi(-s)$ is as defined in Section 3.3.2, and (B.2) is understood to be 0 when $|b_n^1 - b_n^2| = 0$. In this sequence of expressions, the first line applies (3.13); the second line uses $\max \mu_n(A_n(\vec{x})) \leq \mu_n(x^*) + \varepsilon_n = \max\{\mu_n(x^{(1)}, \mu_n(x^{(2)})\} + \varepsilon_n$ together with the fact that $A_n(\vec{x})$ contains $x^{(1)}$ and $x^{(2)}$; the third line uses (3.10) and (3.11); and the last line follows from computations involving the normal distribution, which may be found in equation (14) of Frazier et al. (2009). We will take the limit of (B.2) as n goes to ∞.

By our choice of sample path, μ_n converges to μ_∞, so $\lim_{n\to\infty} |a_n^1 - a_n^2| = |\mu_\infty(x^{(1)}) -$

$\mu_\infty(x^{(2)})| := \gamma_1 < \infty$. We now show that $\lim_{n\to\infty} |b_n^1 - b_n^2|$ is strictly positive. First,

$$|b_n^1 - b_n^2| = \frac{|\Sigma_n\left(x^{(1)},x^{(1)}\right) - 2\Sigma_n\left(x^{(1)},x^{(2)}\right) + \Sigma_n\left(x^{(2)},x^{(2)}\right)|}{\sqrt{\beta^{-1}P + Q_n}} = \frac{|Q_n|}{\sqrt{\beta^{-1}P + Q_n}}.$$

Then, $\left\{\Sigma_n\left(x^{(1)},x^{(1)}\right)\right\}_n$ is bounded above by $\Sigma_0(x^{(1)},x^{(1)})$ by Lemma 10, and $\lim_{n\to\infty}\Sigma_n\left(x^{(2)},x^{(2)}\right) = 0$, so

$$\lim_{n\to\infty} |\Sigma_n\left(x^{(1)},x^{(2)}\right)| \le \lim_{n\to\infty} \sqrt{\Sigma_n\left(x^{(1)},x^{(1)}\right)\Sigma_n\left(x^{(2)},x^{(2)}\right)} = 0.$$

Hence $\lim_{n\to\infty}\Sigma_n\left(x^{(1)},x^{(2)}\right) = 0$. It follows that

$$\lim_{n\to\infty} Q_n = \lim_{n\to\infty}\Sigma_n\left(x^{(1)},x^{(1)}\right) - 2\Sigma_n\left(x^{(1)},x^{(2)}\right) + \Sigma_n\left(x^{(2)},x^{(2)}\right) = \lim_{n\to\infty}\Sigma_n(x^{(1)},x^{(1)}),$$

which is strictly positive by the construction of $x^{(1)}$. Thus,

$$\lim_{n\to\infty}|b_n^1 - b_n^2| = \liminf_{n\to\infty}\frac{|Q_n|}{\sqrt{\beta^{-1}P+Q_n}} = \frac{|\lim_{n\to\infty}\Sigma_n(x^{(1)},x^{(1)})|}{\sqrt{\beta^{-1}P + \lim_{n\to\infty}\Sigma_n(x^{(1)},x^{(1)})}} := \gamma_2 > 0.$$

Recall (B.2). The function $s \mapsto f(-s)$ is continuous, so (B.2) is continuous in $(|a_n^1 - a_n^2|, |b_n^1 - b_n^2|)$ on $[0,\infty)\times(0,\infty)$, and the limit of (B.2) as $n\to\infty$ is $\gamma_2 f(-\gamma_1/\gamma_2)$. Since $V_n(\vec{x}, A_n(\vec{x}), \beta)$ is bounded below by (B.2),

$$\liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) = \frac{\liminf_{n\to\infty} V_n(\vec{x}, A_n(\vec{x}), \beta)}{\beta c(\vec{x})} \ge \frac{1}{\beta c(\vec{x})}\gamma_2 f\left(-\frac{\gamma_1}{\gamma_2}\right) > 0,$$

where we have used that $\gamma_1 < \infty$ and $\gamma_2 > 0$, and $f(-s)$ is strictly positive for $s < \infty$.

□

B.1.3 Proof of Theorem 1 (Convergence Proof)

Proof. We first show, by contradiction, that $\liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) = 0$ almost surely for all $\vec{x} \in \Xi$. Consider an arbitrary sample path of the KG_β^2 algorithm from the almost sure set on which the claim of Lemma 14 holds. Let

$$\chi_0 := \left\{\vec{x}\in\Xi : \lim_{n\to\infty}\nu_n^{\mathrm{KG}_\beta}(\vec{x}) \text{ exists and is } 0\right\} \text{ and } \chi_1 := \left\{\vec{x}\in\Xi : \liminf_{n\to\infty}\nu_n^{\mathrm{KG}_\beta}(\vec{x}) = 0\right\},$$

Suppose for contradiction that $\chi_1 \neq \Xi$, i.e., that

$$\Xi \setminus \chi_1 = \left\{ \vec{x} \in \Xi : \liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) > 0 \right\}$$

is not empty.

Pick $\vec{x} \in \Xi \setminus \chi_1$. By Condition 4, there exists a subsequence of $\mathbb{Z}^+$, denoted by $\{n_i\}_{i=1}^{\infty}$, such that $\vec{x}_{n_i} \in \Xi_{n_i}$ for all i. Also, $\liminf_{i\to\infty} \nu_{n_i}^{\mathrm{KG}_\beta}(\vec{x}) \geq \liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) > 0$. Thus there exists some $\varepsilon > 0$ and a subsequence of $\{n_i\}_{i=1}^{\infty}$, denoted $\{n_j\}_{j=1}^{\infty}$, such that $\nu_{n_j}^{\mathrm{KG}_\beta}(\vec{x}) \geq \varepsilon$ for all j. Then $\nu_{n_j}^{\mathrm{KG}_\beta}(\vec{x}_{n_j}) \geq \nu_{n_j}^{\mathrm{KG}_\beta}(\vec{x}) \geq \varepsilon$ for all j.

For each $\vec{x}' \in \Xi \setminus \chi_0$, the contrapositive of Lemma 13 implies there exists a finite number $N(\vec{x}')$ such that the KG_β^2 algorithm does not choose $\vec{x}'$ for $n > N(\vec{x}')$. Let $N := \max_{\vec{x}' \in \Xi\setminus\chi_0} N(\vec{x}')$. Then $\vec{x}_n \in \chi_0$ for all $n > N$.

For each $\vec{x}' \in \chi_0$, $\lim_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}') = 0$. Hence there exists a finite number $N_0(\vec{x}')$ such that $\nu_n^{\mathrm{KG}_\beta}(\vec{x}') < \varepsilon$ for all $n > N_0(\vec{x}')$. Let $N_0 := \max_{\vec{x}' \in \chi_0} N_0(\vec{x}')$. Then for all $n > N_0$, $\nu_n^{\mathrm{KG}_\beta}(\vec{x}') < \varepsilon$ for any $\vec{x}' \in \chi_0$.

It follows that $\nu_n^{\mathrm{KG}_\beta}(\vec{x}_n) < \varepsilon$ for all $n > \max\{N_0, N\}$, which contradicts $\nu_{n_j}^{\mathrm{KG}_\beta}(\vec{x}_{n_j}) \geq \varepsilon$ for all j. We thus conclude that $\chi_1 = \Xi$ on this sample path, i.e. $\liminf_{n\to\infty} \nu_n^{\mathrm{KG}_\beta}(\vec{x}) = 0$ for all $\vec{x} \in \Xi$. Since the chosen sample path was arbtitrary, this holds almost surely.

Since we chose a sample path on which Lemma 14 holds, $\lim_{n\to\infty} \Sigma_n(x,x) = 0$ on this sample path. Moreover, as the set of sample paths on which Lemma 14 holds is almost sure, $\lim_{n\to\infty} \Sigma_n(x,x) = 0$ almost surely.

To show that $\lim_n \mu_n(x) = \theta(x)$ almost surely for each x, we first show this limit holds in L^2. For each x, $E\left[(\mu_n(x) - \theta(x))^2\right] = E\left[E_n\left[(\mu_n(x) - \theta(x))^2\right]\right] = E\left[\Sigma_n(x,x)\right]$. Taking the limit as $n \to \infty$ and using $0 \leq \Sigma_n(x,x) \leq \Sigma_0(x,x)$ with the dominated convergence theorem implies $\lim_{n\to\infty} E\left[(\mu_n(x) - \theta(x))^2\right] = E\left[\lim_{n\to\infty} \Sigma_n(x,x)\right] = 0$. Then,

since $\mu_n(x)$ converges to $\theta(x)$ in L^2, and Lemma 8 implies $\lim_{n\to\infty}\mu_n(x)$ exists almost surely, this almost sure limit equals $\theta(x)$.

We now show that $\lim_{n\to\infty}\operatorname{argmax}_x \mu_n(x) = \operatorname{argmax}_x \theta(x)$ almost surely. First, $x^* \in \operatorname{argmax}_x \theta(x)$ is almost surely unique as a realization of a multivariate normal random variable, and so $\varepsilon = \theta(x^*) - \max_{x\neq x^*}\theta(x)$ is almost surely strictly positive. Fix a sample path on which $\lim_{n\to\infty}\mu_n(x) = \theta(x)$ for each x (which occurs almost surely). There exists $N < \infty$ such that $|\mu_n - \theta(x)| < \frac{\varepsilon}{2}$ for all $n > N$. Then, for all $n > N$ and all $x \neq x^*$, $\mu_n(x^*) > \theta(x^*) - \frac{\varepsilon}{2} > \theta(x) + \frac{\varepsilon}{2} > \mu_n(x)$, implying x^* is the unique element in $\operatorname{argmax}_x \mu_n(x)$. This shows that $\lim_{n\to\infty}\operatorname{argmax}_x \mu_n(x) = \operatorname{argmax}_x \theta(x)$ almost surely.

□

B.2 MLE for Unknown Parameters in Section 3.6.2

This section derives the MLE used in Section 3.6.2to estimate the parameters η, σ_0^2, $\vec{\alpha} = \{\alpha_i\}_{1\leq i\leq d}$, σ_ε^2 and ρ, which determine Λ, μ_0 and Σ_0 through the model defined in Section 3.6.1. The derivation is related to results of Huang et al. (2006) and Rasmussen and Williams (2006, Sections 2 and 5), but it goes beyond this previous work in considering the sampling correlation ρ.

Continue to let $\mathscr{X}_n$ and $\mathscr{Y}_n$ represent all design points and outputs that have been observed through stage n, including the initialization phase. Let $m = |\mathscr{X}_n|$ be the total number of observations. Set $g = \sigma_0^2/\sigma^2$, $\sigma^2 = \sigma_0^2 + \sigma_\varepsilon^2$, and let δ_{ij} be 1 if $\mathscr{X}_n^{(i)}$ and $\mathscr{X}_n^{(j)}$ are sampled using CRN, and 0 otherwise. Then $\mathscr{Y}_n \sim \mathscr{N}\left(\eta\vec{1}, \sigma^2 R\right)$ for a correlation

matrix R defined by

$$R(i,j) = \begin{cases} 1, & \text{if } i = j, \\ g \exp\left\{ -\sum_{l=1}^{d} \alpha_l \left[\zeta_l\left(\mathscr{X}_n^{(i)}\right) - \zeta_l\left(\mathscr{X}_n^{(j)}\right) \right]^2 \right\} + (1-g)\rho\,\delta_{ij}, & \text{if } i \neq j. \end{cases}$$

The MLE is then $\operatorname{argmax}_{\eta,\sigma_0^2,\vec{\alpha},\sigma_\varepsilon^2,\rho} \log p\left(\mathscr{Y}_n \mid \eta, \sigma_0^2, \vec{\alpha}, \sigma_\varepsilon^2, \rho\right)$. We reparameterize this problem by replacing $(\sigma_0^2, \sigma_\varepsilon^2)$ with (g, σ^2), which uniquely determine each other, to obtain an equivalent formulation of the MLE,

$$\operatorname*{argmax}_{\eta,\sigma^2,g,\vec{\alpha},\rho} \log p\left(\mathscr{Y}_n \mid \eta, \sigma_0^2, \vec{\alpha}, \sigma_\varepsilon^2, \rho\right) = \operatorname*{argmax}_{\eta,\sigma^2,g,\vec{\alpha},\rho} \log p\left(\mathscr{Y}_n \mid \eta, \sigma_0^2, R\right),$$

where we have noted that the parameters $g, \vec{\alpha}, \rho$ only influence the log-likelihood through the correlation matrix R, which is determined by them.

We solve this optimization problem in two steps, first optimizing over σ^2 and η with the other parameters fixed, which can be done analytically, and then numerically optimizing the resulting value over the set of R matrices that can be achieved with the remaining parameters g, $\vec{\alpha}$,ρ. We first describe optimization over σ^2 and η in the following lemma.

Lemma 15. *The maximum log-likelihood over η and σ^2 with R fixed is*

$$\log p\left(\mathscr{Y}_n \mid \widehat{\eta}, \widehat{\sigma}^2, R\right) = \max_{\eta,\sigma^2} \log p\left(\mathscr{Y}_n \mid \eta, \sigma^2, R\right) = -\frac{1}{2}\left(m \log \widehat{\sigma}^2 + \log|R|\right) - \frac{m}{2}(1 + \log 2\pi),$$

where $\vec{1}$ denotes a length-m column vector of ones, $|R|$ is the determinant of R, and

$$\widehat{\sigma}^2 = \frac{1}{n}\left(\mathscr{Y}_n - \widehat{\eta}\vec{1}\right)^T R^{-1}\left(\mathscr{Y}_n - \widehat{\eta}\vec{1}\right) \qquad \widehat{\eta} = \left(\vec{1}^T R^{-1} \vec{1}\right)^{-1} \vec{1}^T R^{-1} \mathscr{Y}_n. \tag{B.3}$$

Proof. We first rewrite the log-likelihood as

$$\begin{aligned} \log p\left(\mathscr{Y}_n \mid \eta, \sigma^2, R\right) &= -\frac{1}{2}\left(\mathscr{Y}_n - \eta\vec{1}\right)^T \left(\sigma^2 R\right)^{-1}\left(\mathscr{Y}_n - \eta\vec{1}\right) - \frac{1}{2}\log|\sigma^2 R| - \frac{m}{2}\log 2\pi \\ &= -\frac{1}{2\sigma^2}\left(\mathscr{Y}_n - \eta\vec{1}\right)^T R^{-1}\left(\mathscr{Y}_n - \eta\vec{1}\right) - \frac{m}{2}\log\sigma^2 - \frac{1}{2}\log|R| - \frac{m}{2}\log 2\pi \end{aligned}$$

Observe that $\widehat{\eta} = \text{argmin}_\eta \left[\left(\mathscr{Y}_n - \eta \vec{1} \right)^T R^{-1} \left(\mathscr{Y}_n - \eta \vec{1} \right) \right] = \left(\vec{1}^T R^{-1} \vec{1} \right)^{-1} \vec{1}^T R^{-1} \mathscr{Y}_n$ is the generalized least squares estimate of η. Let $C := \left(\mathscr{Y}_n - \widehat{\eta} \vec{1} \right)^T R^{-1} \left(\mathscr{Y}_n - \widehat{\eta} \vec{1} \right)$. We then consider a function $H : \mathbb{R}^+ \mapsto \mathbb{R}$ with $H(s) = C/s + m \log s$. Since $H'(s) = -C/s^2 + m/s$, we know C/m is the global minimum of H. It follows that $\widehat{\sigma}^2 = C/m$ is the MLE of σ^2. We thus conclude that $\log p\left(\mathscr{Y}_n \mid \widehat{\eta}, \widehat{\sigma}^2, R \right) = -\frac{1}{2} \left(m \log \widehat{\sigma}^2 + \log |R| \right) - \frac{m}{2}(1 + \log 2\pi)$ is the maximum log marginal likelihood of $\mathscr{Y}_n$ given matrix R.

□

To complete the calculation of the MLE, we maximize the expression for $\log p\left(\mathscr{Y}_n \mid \widehat{\eta}, \widehat{\sigma}^2, R \right)$ from Lemma 15 over matrices R that can be obtained by varying the remaining parameters g, ρ and $\vec{\alpha}$. Denote such maximizers by $\widehat{g}$, $\widehat{\rho}$ and $\widehat{\vec{\alpha}}$. To find them, we examine the partial derivatives of $\log p\left(\mathscr{Y}_n \mid \widehat{\eta}, \widehat{\sigma}^2, R \right)$ with respect to g, ρ and α_l $(l = 1, \ldots, d)$. Let t denote any of these parameters. Rasmussen and Williams (2006, Sec. 5) show $\frac{\partial R^{-1}}{\partial t} = -R^{-1} \frac{\partial R}{\partial t} R^{-1}$ and $\frac{\partial \log |R|}{\partial t} = \text{tr}\left(R^{-1} \frac{\partial R}{\partial t} \right)$. Thus, we can write

$$\frac{\partial}{\partial t} \log p\left(\mathscr{Y}_n \mid R \right) = -\frac{1}{2} \left[\left(\frac{n}{\widehat{\sigma}^2} \frac{\partial \widehat{\sigma}^2}{\partial t} \right) + \text{tr}\left(R^{-1} \frac{\partial R}{\partial t} \right) \right],$$

where

$$\frac{\partial \widehat{\sigma}^2}{\partial t} = -\frac{1}{n} \left(2 \frac{\partial \widehat{\eta}}{\partial t} \left(\mathscr{Y}_n - \widehat{\eta} \vec{1} \right)^T R^{-1} \vec{1} + \left(\mathscr{Y}_n - \widehat{\eta} \vec{1} \right)^T R^{-1} \frac{\partial R}{\partial t} R^{-1} \left(\mathscr{Y}_n - \widehat{\eta} \vec{1} \right) \right),$$

$$\frac{\partial \widehat{\eta}}{\partial t} = \left(\vec{1}^T R^{-1} \vec{1} \right)^{-2} \vec{1}^T R^{-1} \frac{\partial R}{\partial t} R^{-1} \vec{1} \vec{1}^T R^{-1} \mathscr{Y}_n - \left(\vec{1}^T R^{-1} \vec{1} \right)^{-1} \vec{1}^T R^{-1} \frac{\partial R}{\partial t} R^{-1} \mathscr{Y}_n.$$

Each entry of the matrix $\frac{\partial R}{\partial t}$ is given by $\frac{\partial R}{\partial t}(i,i) = 0$ for $i = 1,2,\ldots,m$ and, for $i \neq j$,

$$\frac{\partial R}{\partial g}(i,j) = \exp\left\{-\sum_{l=1}^{d} \alpha_l \left[\zeta_l\left(\mathscr{X}_n^{(i)}\right) - \zeta_l\left(\mathscr{X}_n^{(j)}\right)\right]^2\right\} - \rho\,\delta_{ij},$$

$$\frac{\partial R}{\partial \alpha_l}(i,j) = -g\left[\zeta_l\left(\mathscr{X}_n^{(i)}\right) - \zeta_l\left(\mathscr{X}_n^{(j)}\right)\right]^2 \exp\left\{-\sum_{l=1}^{d} \alpha_l \left[\zeta_l\left(\mathscr{X}_n^{(i)}\right) - \zeta_l\left(\mathscr{X}_n^{(j)}\right)\right]^2\right\},$$

$$l = 1,2,\ldots,d,$$

$$\frac{\partial R}{\partial \rho}(i,j) = (1-g)\delta_{ij}.$$

By applying a Cholesky decomposition to the positive definite matrix R, one can avoid a direct inversion of R in the computations above by solving triangular linear systems. Letting G be the Cholesky factor, the log determinant of R can be calculated efficiently by $\log|R| = 2\sum_{1=1}^{m} \log G_{ii}$.

With these expressions, we can then use gradient based maximization methods to find $\widehat{g}$, $\widehat{\rho}$ and $\widehat{\vec{\alpha}}$. As previously discussed, MLEs $\widehat{\eta}$ and $\widehat{\sigma}^2$ are given by (B.3), and MLEs $\widehat{\sigma}_0^2$, $\widehat{\sigma}_\varepsilon^2$ follow from inverting the definitions of g and σ^2 and applying the inverted expressions to $\widehat{g}$ and $\widehat{\sigma}^2$.

B.3 Gradients Results

This section provides details to support the computation of gradients of the posterior means and predictive covariances with respect to sampling decisions (singletons or pairs), under the assumption that the alternatives sampled are embedded in $\mathbb{R}^d$. These were used in Section 3.6.3 to compute the gradient of the VOI and KG factor with respect to the location of the sampling decision. We also demonstrate simplifications of those results for the special case of a GP prior distribution with Gaussian kernel and constant mean, and a sampling covariance that satisfies a compound sphericity assumption

(as in Section 3.6.1 and Section 3.6.2).

We continue the notational convention of Section 3.6.3, in which derivatives taken with respect to x (in the case of singletons) and $\vec{x}$ (in the case of pairs), actually indicate derivatives taken with respect to these alternatives' grid coordinates: $(\zeta_i(x) : i = 1,\ldots,d)$ for singletons; and $(\zeta_i(x^{(1)}), \zeta_i(x^{(2)}) : i = 1,\ldots,d)$ for pairs.

The expressions provided in B.3.1 and B.3.2 are for general priors and sampling models, and have within them terms such as $\nabla_x[\mu_0(x)]$, $\nabla_x[\Sigma_0(x,x)]$, and $\nabla_x[\Lambda(x,x)]$, whose values depend on the specific prior and form of sampling correlation assumed. Specific values for these quantities for the prior and sampling correlation used in Section 3.6.2 - 3.6.3 are provided in Appendix B.3.3.

B.3.1 Gradients of $\mu_n(x')$ and $\widetilde{\sigma}_n(x,x',\beta)$ when Sampling a Singleton

In this section, we provide expressions for $\nabla_x[\mu_n(x')]$ and $\nabla_x[\widetilde{\sigma}_n(x,x',\beta)]$ for an arbitrary alternative x'. These expressions can be substituted in (3.19) to obtain an expression for the gradient of the VOI $V_n(x,A_n(x),\beta)$ when sampling a singleton $\vec{x} = x$, that holds when $A_n(x)$ is as described in Section 3.6.3.

To support this computation, let $J_n(x') := \nabla_{x'}[\Sigma_0(x',\mathscr{X}_n)]$ be a $d \times |\mathscr{X}_n|$ matrix, the ith column of which is $\nabla_{x'}[\Sigma_0(x',\mathscr{X}_n(i))]$, where $\mathscr{X}_n(i)$ is the ith entry of $\mathscr{X}_n$. Recall $\widetilde{\mathscr{Y}_n}$, S_n and $K_n(\vec{x})$ from (3.5).

We first provide an expression for $\nabla_x[\mu_n(x')]$.

Lemma 16. $\nabla_x[\mu_n(x')] = \nabla_x[\mu_0(x)] + J_n(x)[S_n]^{-1}\widetilde{\mathscr{Y}_n}$ *if* $x = x'$, *and is* $\vec{0}$ *if* $x \neq x'$.

Proof. If $x \neq x'$, then $\mu_n(x')$ does not depend on x, so $\nabla_x[\mu_n(x')] = 0$. Now consider $x = x'$. Note that x is the last element of $\mathscr{X}_{n,x}$. Let e_x be a column vector $[0,0,\ldots,0,1]^T$ with length $|\mathscr{X}_n|+1$. Then

$$\begin{aligned}\mu_n(x) &= e_x^T \mu_n(\mathscr{X}_{n,x}) = e_x^T \left[\mu_0(\mathscr{X}_{n,x}) + K_n(x)\widetilde{\mathscr{Y}_n}\right] \\ &= e_x^T \mu_0(\mathscr{X}_{n,x}) + e_x^T \Sigma_0(\mathscr{X}_{n,x}, \mathscr{X}_{n,x}) \left[I_{|\mathscr{X}_n|}, \vec{0}\right]^T [S_n]^{-1}\widetilde{\mathscr{Y}_n} \\ &= \mu_0(x) + \Sigma_0(x, \mathscr{X}_n)[S_n]^{-1}\widetilde{\mathscr{Y}_n},\end{aligned}$$

where we use (3.5) and (3.6) in the last line. Because $[S_n]^{-1}\widetilde{\mathscr{Y}_n}$ does not depend on x, the gradient is $\nabla_x[\mu_n(x)] = \nabla_x(\mu_{0x}) + \nabla_x[\Sigma_0(x,\mathscr{X}_n)][S_n]^{-1}\widetilde{\mathscr{Y}_n} = \nabla_x[\mu_0(x)] + J_n(x)[S_n]^{-1}\widetilde{\mathscr{Y}_n}$.

□

We now provide an expression for $\nabla_x[\widetilde{\sigma}_n(x,x',\beta)]$.

Lemma 17.

$$\nabla_x\left[\widetilde{\sigma}_n\left(x,x',\beta\right)\right] = \frac{B\nabla_x[\Sigma_n(x',x)] - \Sigma_n(x',x)\nabla_x(B)}{B^2}$$

where $B := \sqrt{\Lambda(x,x)/\beta + \Sigma_n(x,x)}$, *and*

$$\nabla_x\left[\Sigma_n\left(x',x\right)\right] = \begin{cases} \nabla_x[\Sigma_0(x',x)] - J_n(x)[S_n]^{-1}\Sigma_0(\mathscr{X}_n,x'), & \text{if } x' \neq x, \\ \nabla_x[\Sigma_0(x,x)] - 2J_n(x)[S_n]^{-1}\Sigma_0(\mathscr{X}_n,x), & \text{if } x' = x, \end{cases}$$

$$\nabla_x(B) = \frac{1}{2B}\left\{\nabla_x[\Lambda(x,x)/\beta + \Sigma_0(x,x)] - 2J_n(x)[S_n]^{-1}\Sigma_0(\mathscr{X}_n,x)\right\}.$$

Proof. Recall that $\widetilde{\sigma}_{nx'}(X,\beta) = \Sigma_n(x',x)/B$, so $\nabla_x[\widetilde{\sigma}_n(x,x',\beta)]$ is as claimed. Next, recall from Lemma 9 that $\Sigma_n(x',x) = \Sigma_0(x',x) - \Sigma_0(x,\mathscr{X}_n)[S_n]^{-1}\Sigma_0(\mathscr{X}_n,x')$. Thus if $x' \neq x$, then

$$\begin{aligned}\nabla_x\left[\Sigma_n(x',x)\right] &= \nabla_x\left[\Sigma_0(x',x)\right] - \nabla_x[\Sigma_0(x,\mathscr{X}_n)][S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x'\right) \\ &= \nabla_x\left[\Sigma_0(x',x)\right] - J_n(x)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x'\right).\end{aligned}$$

If $x' = x$, then using standard matrix differentiation, we can compute the gradient as $\nabla_x [\Sigma_n(x,x)] = \nabla_x [\Sigma_0(x,x)] - 2J_n(x)[S_n]^{-1}\Sigma_0(\mathscr{X}_n, x)$. The claimed formula for $\nabla_x(B)$ follows from simple algebra. □

B.3.2 Gradients of $\mu_n(x')$ and $\widetilde{\sigma}_n(\vec{x}, x', \beta)$ when Sampling a Pair

In this section, we describe computation of $\nabla_{\vec{x}}[\mu_n(x')]$ and $\nabla_{\vec{x}}[\widetilde{\sigma}_n(\vec{x}, x', \beta)]$ for an arbitrary alternative x'. These expressions can be substituted in (3.20) to obtain an expression for the gradient of the VOI $V_n(x, A_n(x), \beta)$ when sampling a pair $\vec{x}$, that holds when $A_n(x)$ is as described in Section 3.6.3.

The gradient $\nabla_{x^{(i)}}[\mu_n(x')]$ for $i = 1,2$ is given in Lemma 16 where we replace x by $x^{(i)}$. The derivation is similar and is hence omitted. The derivation of $\nabla_{x^{(i)}}[\widetilde{\sigma}_n(\vec{x}, x', \beta)]$ when sampling pairs differs from that of the gradient when sampling a singleton, so details follow.

Lemma 18. *For $i = 1,2$,*

$$\begin{aligned}\nabla_{x^{(i)}}\left[\widetilde{\sigma}_n\left(\vec{x}, x', \beta\right)\right] = \frac{1}{B^2}\Big\{B\nabla_{x^{(i)}}\left[\Sigma_n\left(x', x^{(1)}\right) - \Sigma_n\left(x', x^{(2)}\right)\right] \\ - \left[\Sigma_n\left(x', x^{(1)}\right) - \Sigma_n\left(x', x^{(2)}\right)\right]\nabla_{x^{(i)}}(B)\Big\},\end{aligned} \tag{B.4}$$

where

$$\begin{aligned}B := \Big\{\beta^{-1}\left[\Lambda\left(x^{(1)}, x^{(1)}\right) + \Lambda\left(x^{(2)}, x^{(2)}\right) - 2\Lambda\left(x^{(1)}, x^{(2)}\right)\right] \\ + \Sigma_n\left(x^{(1)}, x^{(1)}\right) + \Sigma_n\left(x^{(2)}, x^{(2)}\right) - 2\Sigma_n\left(x^{(1)}, x^{(2)}\right)\Big\}^{\frac{1}{2}},\end{aligned} \tag{B.5}$$

$$\nabla_{x^{(i)}}\left[\Sigma_n\left(x',x^{(1)}\right)-\Sigma_n\left(x',x^{(2)}\right)\right] \tag{B.6}$$

$$=\begin{cases}\nabla_{x^{(1)}}\left[\Sigma_0\left(x',x^{(1)}\right)\right]-J_n\left(x^{(1)}\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x'\right), & \text{if } i=1, x'\neq x^{(1)}\\ \nabla_{x^{(1)}}\left[\Sigma_0\left(x^{(1)},x^{(1)}\right)-\Sigma_0\left(x^{(1)},x^{(2)}\right)\right] & \\ \qquad -J_n\left(x^{(1)}\right)[S_n]^{-1}\left[2\Sigma_0\left(\mathscr{X}_n,x^{(1)}\right)-\Sigma_0\left(\mathscr{X}_n,x^{(2)}\right)\right], & \text{if } i=1, x'=x^{(1)}\\ -\nabla_{x^{(2)}}\left[\Sigma_0\left(x',x^{(2)}\right)\right]+J_n\left(x^{(2)}\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x'\right), & \text{if } i=2, x'\neq x^{(2)}\\ \nabla_{x^{(2)}}\left[\Sigma_0\left(x^{(1)},x^{(2)}\right)-\Sigma_0\left(x^{(2)},x^{(2)}\right)\right] & \\ \qquad +J_n\left(x^{(2)}\right)[S_n]^{-1}\left[2\Sigma_0\left(\mathscr{X}_n,x^{(2)}\right)-\Sigma_0\left(\mathscr{X}_n,x^{(1)}\right)\right], & \text{if } i=2, x'=x^{(2)}\end{cases}$$

and

$$\nabla_{x^{(i)}}(B)=\frac{1}{B}\Bigg\{\nabla_{x^{(i)}}\left[\frac{1}{2}\left[\beta^{-1}\Lambda(x^{(i)},x^{(i)})+\Sigma_0(x^{(i)},x^{(i)})\right]-\left[\beta^{-1}\Lambda(x^{(1)},x^{(2)})+\Sigma_0(x^{(1)},x^{(2)})\right]\right]$$
$$+J_n\left(x^{(i)}\right)[S_n]^{-1}\left[\Sigma_0\left(\mathscr{X}_n,x^{3-i}\right)-\Sigma_0\left(\mathscr{X}_n,x^{i}\right)\right]\Bigg\}.$$

Proof. First, recall that $\widetilde{\sigma}_n(\vec{x},x',\beta)=\frac{1}{B}\left[\Sigma_n\left(x',x^{(1)}\right)-\Sigma_n\left(x',x^{(2)}\right)\right]$, hence (B.4) follows. Using (3.5) and (3.6), similar to the proof of Lemma 17, we have for $i=1,2$ that $\Sigma_n\left(x',x^{(i)}\right)=\Sigma_0\left(x',x^{(i)}\right)-\Sigma_0\left(x^{(i)},\mathscr{X}_n\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x'\right)$.

We show the first two cases ($i=1$) of (B.6). The other two cases ($i=2$) follow similarly, and are ommitted. In the first case ($x'\neq x^{(1)}$) we have $\nabla_{x^{(1)}}\left[\Sigma_n\left(x',x^{(1)}\right)-\Sigma_n\left(x',x^{(2)}\right)\right]=\nabla_{x^{(1)}}\left[\Sigma_n\left(x',x^{(1)}\right)\right]=\nabla_{x^{(1)}}\left[\Sigma_0\left(x',x^{(1)}\right)\right]-J_n\left(x^{(1)}\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x'\right)$. In the second case ($x'=x^{(1)}$) then from the observation that $\Sigma_n\left(x^{(1)},x^{(1)}\right)-\Sigma_n\left(x^{(1)},x^{(2)}\right)=\Sigma_0\left(x^{(1)},x^{(1)}\right)-\Sigma_0\left(x^{(1)},x^{(2)}\right)-\Sigma_0\left(x^{(1)},\mathscr{X}_n\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x^{(1)}\right)+\Sigma_0\left(x^{(1)},\mathscr{X}_n\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x^{(2)}\right)$, it follows from

standard matrix differentiation and the definition of $J_n\left(x^{(1)}\right)$ that

$$\begin{aligned}\nabla_{x^{(1)}}\left[\Sigma_n\left(x^{(1)},x^{(1)}\right)-\Sigma_n\left(x^{(1)},x^{(2)}\right)\right] =&\nabla_{x^{(1)}}\left[\Sigma_0\left(x^{(1)},x^{(1)}\right)-\Sigma_0\left(x^{(1)},x^{(2)}\right)\right]\\ &-2J_n\left(x^{(1)}\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x^{(1)}\right)\\ &+J_n\left(x^{(1)}\right)[S_n]^{-1}\Sigma_0\left(\mathscr{X}_n,x^{(2)}\right).\end{aligned}$$

It remains to compute $\nabla_{x^{(i)}}(B)$. Notice that for $i=1$,

$$\begin{aligned}&\nabla_{x^{(1)}}\left[\Sigma_n\left(x^{(1)},x^{(1)}\right)+\Sigma_n\left(x^{(2)},x^{(2)}\right)-2\Sigma_n\left(x^{(1)},x^{(2)}\right)\right]\\ =&\nabla_{x^{(1)}}\left[\Sigma_n\left(x^{(1)},x^{(1)}\right)-\Sigma_n\left(x^{(1)},x^{(2)}\right)\right]-\nabla_{x^{(1)}}\left[\Sigma_n\left(x^{(2)},x^{(1)}\right)-\Sigma_n\left(x^{(2)},x^{(2)}\right)\right]\\ =&\nabla_{x^{(1)}}\left[\Sigma_0\left(x^{(1)},x^{(1)}\right)-2\Sigma_0\left(x^{(1)},x^{(2)}\right)\right]+2J_n(x^{(1)})[S_n]^{-1}\left[\Sigma_0\left(\mathscr{X}_n,x^{(2)}\right)-\Sigma_0\left(\mathscr{X}_n,x^{(1)}\right)\right],\end{aligned}$$

where the last equation follows from (B.6). $\nabla_{x^{(1)}}(B)$ then follows from the definition of B. The formula for $\nabla_{x^{(2)}}(B)$ is similar.

□

B.3.3 Simplification under Compound Sphericity, Constant Prior Mean, and Gaussian Kernel

The gradients of the VOI and KG factors in Appendices B.3.1-B.3.2 involve the gradients of the sampling covariance matrix, and of the mean and covariance for the unknown mean θ. That is, they include the terms $\nabla_x[\Lambda(x,x')]$, $\nabla_x[\mu_0(x)]$ and $\nabla_x[\Sigma_0(x,x')]$ for arbitrary x,x'. These values depend on the prior distribution and the assumed form of the sampling correlation.

In this section, we provide specific values for these quantities that result from adopting the modeling choices from Section 3.6.2 - 3.6.3: a GP prior with a Gaussian kernel

and constant mean, and compound sphericity. These choices substantially simplify the expressions from Appendices B.3.1-B.3.2, as many terms become 0.

First, under compound sphericity, $\nabla_x [\Lambda(x,x')] = \vec{0}$ for arbitrary x and x'. Second, under constant prior mean, $\nabla_x [\mu_0(x)] = \nabla_x [\eta] = \vec{0}$. Third, we compute $\nabla_x [\Sigma_0(x,x')]$ for arbitrary x and x'. Denote by $\circ$ the Hadamard (componentwise) product of two vectors u and v of the same length, so that $(u \circ v)(i) = u(i)v(i)$. Then $\nabla_x [\Sigma_0 (x,x')] = 2\Sigma_0 (x,x') \alpha \circ [\zeta (x) - \zeta (x')]$. In particular, $\nabla_x [\Sigma_0(x,x)] = \nabla_x \left[\sigma_0^2\right] = \vec{0}$.

APPENDIX C

APPENDIX OF CHAPTER 4

C.1 Proof of Lemma 2

Proof. By (4.3) and (4.5),

$$a_{n+1}(x) = \int \mu_{n+1}(x+\delta, \omega)\, p(\delta, \omega)\, \mathrm{d}\delta\, \mathrm{d}\omega.$$

Since the posterior mean $\mu_{n+1}(\cdot,\cdot)$ is a linear function of the observations up to time n and y_{n+1}, we can write $a_{n+1}(x) \mid D_n, \theta_{n+1}, \omega_{n+1}$ as

$$s_n(x, \theta_{n+1}, \omega_{n+1}) + t_n(x, \theta_{n+1}, \omega_{n+1}) \cdot y_{n+1},$$

where s_n and t_n are real-valued, deterministic functions of D_n. Now since y_{n+1} conditioned on $D_n, \theta_{n+1}, \omega_{n+1}$ is normally distributed, we know that $a_{n+1}(x)$ is also normally distributed conditioned on $D_n, \theta_{n+1}, \omega_{n+1}$.

By the tower property,

$$a_n(x) = \mathbb{E}_n[g(x)] = \mathbb{E}_n[\mathbb{E}_{n+1}[g(x)]].$$

Also since Σ_{n+1} does not depend on y_{n+1} (it is fully determined by D_n, θ_{n+1} and ω_{n+1}), we know that b_n is well defined. By (4.4) and the conditional variance formula,

$$\begin{aligned}
& b_n(x, \theta_{n+1}, \omega_{n+1}) \\
&= \mathrm{Var}_n[g(x)] - \mathrm{Var}_{n+1}[g(x)] \\
&= \mathrm{Var}_n[g(x)] - \mathbb{E}_n[\mathrm{Var}_{n+1}[g(x)] \mid \theta_{n+1}, \omega_{n+1}] \\
&= \mathrm{Var}_n[\mathbb{E}_{n+1}[g(x)] \mid \theta_{n+1}, \omega_{n+1}].
\end{aligned}$$

Thus (4.8) follows. □

C.2 Convergence Properties

Assume $\Theta \times \Omega$ is compact, and $U(\cdot,\cdot)$ is a continuous function on $\Theta \times \Omega$.

For any set $A \subseteq \Theta \times \Omega$, define

$$M_n(A) = \sum_{m \leq n} \mathbf{1}_{\{(\theta_m, \omega_m) \in A\}}.$$

Proposition 2. *For any non-empty $A \subseteq \Theta \times \Omega$, $M_n(A) > 0$ for some n.*

Proof. For contradiction we assume that there exists a non-empty $A \subseteq \Theta \times \Omega$ such that $M_n(A) = 0$ for all n. Let (θ^*, ω^*) be a cluster point of the sequence $\{(\theta_n, \omega_n)\}_n$ (it exists since $\Theta \times \Omega$ is compact). Let A' be a closed, non-empty subset of A, and

$$\varepsilon = \liminf_{n \to \infty} \sup_{(\theta,\omega) \in A'} V_n(\theta, \omega).$$

Lemma 19. *Under certain regularity assumptions on the prior, $\varepsilon > 0$.*

Proof. Use that we never measure in A' implying variance is bounded below, and used that the posterior mean cannot be too large or small, based on $\sup U$ and $\inf U$. □

Thus, there exist $\theta', \omega' \in A' \subseteq A$ such that

$$\liminf_{n \to \infty} V_n\left(\theta', \omega'\right) \geq \varepsilon.$$

Lemma 20. *There exists $\delta > 0$ such that*

$$\limsup_{n \to \infty} V_n(\theta, \omega) \leq \varepsilon/2$$

for all $(\theta, \omega) \in B_\delta(\theta^*, \omega^*)$.

Proof. Use that we measure infinitely often near (θ^*, ω^*).

$$\begin{aligned}
V_n(\theta,\omega) &= \mathbb{E}\left[\max_x a_n(x) + b_n(x,\theta,\omega)\,Z\right] - \max_x a_n(x) \\
&\le \max_x a_n(x) + \mathbb{E}\left[\max_x b_n(x,\theta,\omega)\,Z\right] - \max_x a_n(x) \\
&= \mathbb{E}\left[\max_x b_n(x,\theta,\omega)\,Z\right] \\
&= \mathbb{E}\left[\max_x b_n(x,\theta,\omega)\,Z\mathbf{1}_{Z>0} + \max_x b_n(x,\theta,\omega)\,Z\mathbf{1}_{Z<0}\right] \\
&= \max_x b_n(x,\theta,\omega)\,\mathbb{E}\left[Z\mathbf{1}_{Z>0}\right] + \min_x b_n(x,\theta,\omega)\,\mathbb{E}\left[Z\mathbf{1}_{Z<0}\right] \\
&= \left[\max_x b_n(x,\theta,\omega) - \min_x b_n(x,\theta,\omega)\right]/\sqrt{2\pi} \\
&\le \sqrt{\frac{2}{\pi}}\max_x |b_n(x,\theta,\omega)|.
\end{aligned}$$

By (4.19),

$$b_n(x,\theta,\omega) = \frac{\iint \Sigma_n(x+\delta',\omega',\theta,\omega)\,p(\delta',\omega')\,\mathrm{d}\delta'\mathrm{d}\omega'}{\sqrt{\Sigma_n(\theta,\omega,\theta,\omega)}}.$$

$$\begin{aligned}
&|\Sigma_n\left(x+\delta',\omega',\theta,\omega\right)| \\
&\le \sqrt{\Sigma_n(x+\delta',\omega',x+\delta',\omega')\,\Sigma_n(\theta,\omega,\theta,\omega)}
\end{aligned}$$

□

Thus, there exists $N < \infty$ such that for all $n \ge N$, $V_n(\theta',\omega') \ge \varepsilon > V_n(\theta,\omega)$, for any $(\theta,\omega) \in B_\delta(\theta^*,\omega^*)$. It follows from (4.6) that for all $n \ge N$, $(\theta_n,\omega_n) \notin B_\delta(\theta^*,\omega^*)$, which contradicts the fact that (θ^*,ω^*) is a cluster point of the sequence $\{(\theta_n,\omega_n)\}_n$.

The contradiction shows $M_n(A) > 0$ for some n.

□

Theorem 2. *Every point (θ,ω) is a cluster point of the sequence $\{(\theta_n,\omega_n)\}_n$, or is measured.*

Proof. Let $\delta = \inf_n ||(\theta,\omega) - (\theta_n,\omega_n)||$. For contradiction assume $\delta > 0$. Let $A = B_{\delta/2}(\theta,\omega)$. By Proposition 2, there exists some n such that $(\theta_n,\omega_n) \in A$, which means that $||(\theta,\omega) - (\theta_n,\omega_n)|| \leq \delta/2$. This contradicts the definition of δ. □

Corollary 1. $\lim_{n\to\infty} \mu_n(\theta,\omega) = U(\theta,\omega)$ *for all* θ,ω.

BIBLIOGRAPHY

Abramowitz, M, I A Stegun. 1964. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. Dover Publications.

Amazon.com. 2012. . http://aws.amazon.com/ec2/pricing/.

Andradóttir, S. 1998. Simulation optimization. *Handbook of simulation: Principles, methodology, advances, applications, and practice*. Wiley-Interscience, New York, 307–333.

Andradóttir, S. 2006. An overview of simulation optimization via random search. S G Henderson, B L Nelson, eds., *Handbooks in Operations Research and Management Science: Simulation*. Amsterdam, 617–631.

Andradóttir, S, D Goldsman, S H Kim. 2005. Finding the best in the presence of a stochastic constraint. M E Kuhl, N M Steiger, F B Armstrong, J A Joines, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 732–738.

Andradóttir, S, S H Kim. 2010. Fully sequential procedures for comparing constrained systems via simulation. *Naval Research Logistics (NRL)* **57**(5) 403–421.

Ankenman, B, B L Nelson, J Staum. 2010. Stochastic Kriging for Simulation Metamodeling. *Operations Research* **58**(2) 371–382.

Araman, V F, R Caldenty. 2009. Dynamic Pricing for Perishable Products with Demand Learning. *Operations Research* **57** 1169–1188.

Barton, R R. 2009. Simulation optimization using metamodels. M D Rossetti, R R Hill, B Johansson, A Dunkin, R G Ingalls, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 230–238.

Bartz-Beielstein, T, C W G Lasarczyk, M Preuss. 2005. Sequential parameter optimization. *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 1. IEEE, 773–780.

Batur, D, S H Kim. 2010. Finding feasible systems in the presence of constraints on multiple performance measures. *ACM Transactions on Modeling and Computer Simulation (TOMACS)* **20**(3) 13.

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

Bechhofer, R E, B W Turnbull. 1978. Two (k+1)-decision selection procedures for comparing k normal means with a specified standard. *Journal of the American Statistical Association* 385–392.

Bechhofer, Robert Eric, Jack Kiefer, Milton Sobel. 1968. *Sequential identification and ranking procedures*, vol. 3. University of Chicago Press Chicago.

Bellman, R. 1954. The Theory of Dynamic Programming. *Bull. Amer. Math. Soc.* **60** 503–516.

Bertsekas, D P. 2005. *Dynamic programming and optimal control, vol. I*. 3rd ed. Athena Scientific.

Bertsekas, D P. 2007. *Dynamic programming and optimal control, vol. II*. 3rd ed. Athena Scientific.

Bertsekas, D P, J N Tsitsiklis. 1996. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA.

Bofinger, E, G J Lewis. 1992. Two-stage procedures for multiple comparisons with a control. *American Journal of Mathematical and Management Sciences* **12** 253.

Booker, A J, J E Dennis, P D Frank, D B Serafini, V Torczon, M W Trosset. 1999. A rigorous framework for optimization of expensive functions by surrogates. *Structural and Multidisciplinary Optimization* **17**(1) 1–13.

Brochu, E, V M Cora, N de Freitas. 2009. A Tutorial on {B}ayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. Tech. Rep. TR-2009-23, Department of Computer Science, University of British Columbia.

Brown, D B, J E Smith, P Sun. 2010. Information Relaxations and Duality in Stochastic Dynamic Programs.

Brown, David B, James E Smith. 2011. Dynamic portfolio optimization with transaction costs: Heuristics and dual bounds. *Management Science* **57**(10) 1752–1770.

Buchholz, P, A Thummler. 2005. Enhancing evolutionary algorithms with statistical selection

procedures for simulation optimization. *Proceedings of the 2005 Winter Simulation Conference* 11—-pp.

Chang, P B, B J Williams, T J Santner, W I Notz, Bartel D.L. 1999. Robust Optimization of Total Joint Replacements Incorporating Environmental Variables. *Journal of Biomechanical Engineering* **121** 304–310.

Chen, C, L H Lee. 2010. *Stochastic simulation optimization: an optimal computing budget allocation*. World Scientific.

Chen, X, B E Ankenman, B L Nelson. 2012. The Effects of Common Random Numbers on Stochastic Kriging Metamodels. *ACM TOMACS* **22**(7) 1–20.

Chen, X, B E Ankenman, B L Nelson. 2013. Enhancing Stochastic Kriging Metamodels with Gradient Estimators. *Operations Reesarch* **61** in press.

Chick, S E. 2006. Bayesian ideas and discrete event simulation: why, what and how. L F Perrone, F P Wieland, J Liu, B G Lawson, D M Nicol, R M Fujimoto, eds., *Proceedings of the 37th conference on Winter simulation*. Winter Simulation Conference, Piscataway, NJ, 96–105.

Chick, S E, J Branke, C Schmidt. 2010. Sequential Sampling to Myopically Maximize the Expected Value of Information. *INFORMS J. on Computing* **22**(1) 71–80.

Chick, S E, P I Frazier. 2009. The Conjunction of the Knowledge Gradient and Economic Approach to Simulation Selection. *Proceedings of the 2009 Winter Simulation Conference Proceedings*. Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 528–539.

Chick, S E, P I Frazier. 2011. Sequential Sampling for Selection with {ESP}.

Chick, S E, P I Frazier. 2012. Sequential Sampling for Selection with Economics of Selection Procedures. *Management Science* **58**(3) 550–569.

Chick, S E, N Gans. 2009. Economic Analysis of Simulation Selection Problems. *Management Sci.* **55**(3) 421–437.

Chick, S E, K Inoue. 2001a. New Procedures to Select the Best Simulated System Using Common Random Numbers. *Management Science* **47**(8) 1133–1149.

Chick, S E, K Inoue. 2001b. New Two-Stage and Sequential Procedures for Selecting the Best Simulated System. *Operations Research* **49**(5) 732–743.

Clark, C E. 1961. The greatest of a finite set of random variables. *Operations Research* **9**(2) 145–162.

Clark, G M, W Yang. 1986. A Bonferroni selection procedure when using common random numbers with unknown variances. J Wilson J. Henriksen, S Roberts, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 313–315.

Cressie, N A C. 1993. *Statistics for Spatial Data, revised edition*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics, Wiley Interscience, New York.

Damerdji, H, M K Nakayama. 1996. Two-stage procedures for multiple comparisons with a control in steady-state simulations. J M Charnes, D J Morrice, D T Brunner, J J Swain, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 375.

DeGroot, M H. 1970. *Optimal {S}tatistical {D}ecisions*. McGraw Hill, New York.

Ding, X, M L Puterman, A Bisi. 2002. The Censored Newsvendor and the Optimal Acquisition of Information. *Operations Research* **50**(3) 517–527.

Dudewicz, E J, S R Dalal. 1983. Multiple comparisons with a control when variances are unknown and unequal. *American Journal of Mathematics and Management Sciences* **4** 275–295.

Dudewicz, E J, J S Ramberg. 1972. Multiple comparisons with a control: Unknown variances. *The Annual Technical Conference Transactions of the American Society of Quality Control*, vol. 26. 483–488.

Dunnett, C W. 1955. A multiple comparison procedure for comparing several treatments with a control. *Journal of the American Statistical Association* **50**(272) 1096–1121.

Dynkin, E B, A A Yushkevich. 1979. *Controlled Markov Processes*. Springer, New York.

Forrester, A, A Sobester, A Keane. 2008. *Engineering design via surrogate modelling: a practical guide*. Wiley, West Sussex, UK.

Frazier, P I. 2009. Knowledge-Gradient Methods for Statistical Learning. Ph.D. thesis, Princeton University.

Frazier, P I. 2012. Tutorial: Optimization via Simulation with Bayesian Statistics and Dynamic Programming. *Proceedings of the 2012 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 1–16.

Frazier, P I. 2013. . http://people.orie.cornell.edu/pfrazier/src.html.

Frazier, P I, James J Cochran, Louis A Cox, Pinar Keskinocak, Jeffrey P Kharoufeh, J Cole Smith. 2010. *Decision-Theoretic Foundations of Simulation Optimization*. John Wiley & Sons, Inc. doi:10.1002/9780470400531.eorms0780. URL `http://dx.doi.org/10.1002/9780470400531.eorms0780`.

Frazier, P I, W B Powell. 2008. The Knowledge-Gradient Stopping Rule for Ranking and Selection. S J Mason, R R Hill, L Mönch, O Rose, T Jefferson, J W Fowler, eds., *Proceedings of the 2008 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 305–312.

Frazier, P I, W B Powell. 2010. Paradoxes in Learning and the Marginal Value of Information. *Decision Analysis* **7**(4) 378–403.

Frazier, P I, W B Powell, S Dayanik. 2008. A Knowledge Gradient Policy for Sequential Information Collection. *SIAM Journal on Control and Optimization* **47**(5) 2410–2439.

Frazier, P I, W B Powell, S Dayanik. 2009. The Knowledge Gradient Policy for Correlated Normal Beliefs. *INFORMS Journal on Computing* **21**(4) 599–613.

Frazier, P I, J Xie, S E Chick. 2011. Value of Information Methods for Pairwise Sampling with Correlations. *Proceedings of the 2011 Winter Simulation Conference*. 3979–3991.

Fu, M. 1994. Optimization via simulation: A review. *Ann. Oper. Res.* **53**(1) 199–248.

Fu, M C. 2002. Optimization for simulation: Theory vs. practice. *INFORMS Journal on Computing* **14**(3) 192–215.

Fu, M C, J Q Hu, C H Chen, X Xiong. 2004. Optimal computing budget allocation under correlated sampling. R G Ingalls, M D Rossetti, J S Smith, B A Peters, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 595–603.

Gelman, A B, J B Carlin, H S Stern, D B Rubin. 2004. *Bayesian data analysis*. 2nd ed. CRC Press, Boca Raton, FL.

Ginsbourger, D, R Le Riche, L Carraro. 2008. A multi-points criterion for deterministic parallel global optimization based on Gaussian processes .

Gittins, J C, D M Jones. 1974. A dynamic allocation index for the sequential design of experiments. J Gani, ed., *Progress in Statistics*. North-Holland, Amsterdam, 241–266.

Gittins, John, Kevin Glazebrook, Richard Weber. 2011. *Multi-armed Bandit Allocation Indices*. 2nd ed. Wiley.

Glynn, P, S Juneja. 2004. A large deviations perspective on ordinal optimization. R G Ingalls, M D Rossetti, J S Smith, B A Peters, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 577–585.

Glynn, P, S Juneja. 2011. Ordinal optimization: a nonparametric framework. *Proceedings of the 2011 Winter Simulation Conference*. Winter Simulation Conference, 4062–4069.

Goldsman, D, B Nelson. 1994. Ranking, Selection and Multiple Comparisons in Computer Simulation. J D Tew, S Manivannan, D A Sadowski, A F Seila, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 192–199.

Golub, G H, C F Van Loan. 1996. *Matrix Computations*. John Hopkins University Press, Baltimore, MD.

Gupta, S S, K J Miescke. 1996. Bayesian look ahead one-stage sampling allocations for selection of the best population. *Journal of Statistical Planning and Inference* **54**(2) 229–244.

Haugh, Martin B, Leonid Kogan. 2004. Pricing American options: a duality approach. *Operations Research* **52**(2) 258–270.

Healey, C, S Andradóttir, S H Kim. 2012. Selection Procedures for Simulations with Multiple Constraints.

Henderson, S G. 2000. Mathematics for simulation. *Simulation Conference Proceedings, 2000. Winter*, vol. 1. IEEE, 137–146.

Henderson, Shane G, Barry L Nelson. 2006. *Handbooks in Operations Research and Management Science: Simulation: Simulation*, vol. 13. Elsevier.

Hochberg, Y, A C Tamhane. 1987. *Multiple comparison procedures*. Wiley New York.

Hong, L J, B L Nelson. 2006. Discrete optimization via simulation using COMPASS. *Operations Research* **54**(1) 115–129.

Hong, L J, B L Nelson, S G Henderson, J Xie. 2012 Accessed July 2013. . `http://simopt.org/wiki/index.php?title=Assemble_to_order`.

Hsu, J C. 1996. *Multiple Comparisons: theory and methods*. CRC Press, Boca Raton.

Hu, J, Y Wang, E Zhou, M C Fu, S I Marcus. 2012. A Survey of Some Model-Based Methods for Global Optimization. *Optimization, Control, and Applications of Stochatic Systems: in honor of Onésimo Hernández-Lerma*. Birkhäuser, 157–179.

Huang, D, T T Allen, W I Notz, N Zeng. 2006. Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models. *Journal of Global Optimization* **34**(3) 441–466.

Hunter, S R, N A Pujowidianto, Chun-Hung Chen, Loo Hay Lee, R Pasupathy, Chee Meng Yap. 2011. Optimal sampling laws for constrained simulation optimization on finite sets: The bivariate normal case. *Simulation Conference (WSC), Proceedings of the 2011 Winter*. IEEE, 4289–4297.

Hunter, Susan R, Raghu Pasupathy. 2012. Optimal sampling laws for stochastically constrained simulation optimization on finite sets. *INFORMS Journal on Computing* .

Hutter, F. 2009. Automated configuration of algorithms for solving hard computational problems. Ph.D. thesis, Citeseer.

Jones, D R. 2001. A Taxonomy of Global Optimization Methods Based on Response Surfaces. *Journal of Global Optimization* **21**(4) 345–383.

Jones, D R, M Schonlau, W J Welch. 1998. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization* **13**(4) 455–492.

Kiefer, J. 1953. Sequential minimax search for a maximum. *Proceedings of the American Mathematical Society* **4**(3) 502–506.

Kim, S H. 2005. Comparison with a standard via fully sequential procedures. *ACM TOMACS* **15**(2) 155–174.

Kim, S H, B L Nelson. 2006. Selecting the best system. *Handbooks in operations research and management science: simulation* **13** 501–534.

Krause, A, J Leskovec, C Guestrin, J VanBriesen, C Faloutsos. 2008. Efficient sensor placement optimization for securing large water distribution networks. *Journal of Water Resources Planning and Management* **134** 516.

Kushner, H J. 1964. A new method of locating the maximum of an arbitrary multi- peak curve in the presence of noise. *Journal of Basic Engineering* **86** 97–106.

Lizotte, D, T Wang, M Bowling, D Schuurmans. 2007. Automatic gait optimization with gaussian process regression. *Proc. of IJCAI*. 944–949.

Luo, Yao, Eunji Lim. 2013. Simulation-based optimization over discrete sets with noisy constraints. *IIE Transactions* **45**(7) 699–715.

Mahajan, A, D Teneketzis. 2008. Multi-armed bandit problems. *Foundations and Applications of Sensor Management* 121–151.

Marsden, A L, J A Feinstein, C A Taylor. 2008. A computational framework for derivative-free optimization of cardiovascular geometries. *Computer Methods in Applied Mechanics and Engineering* **197**(21–24) 1890–1905.

Marsden, A L, M Wang, J E Dennis, P Moin. 2004. Optimal aeroacoustic shape design using the surrogate management framework. *Optimization and Engineering* **5**(2) 235–262.

Maxwell, M S, M Restrepo, S G Henderson, H Topaloglu. 2010. Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing* **22**(2) 266–281.

Mockus, J. 1989. *Bayesian approach to global optimization: theory and applications*. Kluwer Academic, Dordrecht.

Mockus, J. 1994. Application of Bayesian approach to numerical methods of global and stochastic optimization. *Journal of Global Optimization* **4**(4) 347–365.

Nakayama, M K. 2000. Multiple comparisons with the best using common random numbers for steady-state simulations. *Journal of Statistical Planning and Inference* **85**(1-2) 37–48.

Nelson, B L, D Goldsman. 2001. Comparisons with a standard in simulation experiments. *Management Science* **47**(3) 449–463.

Nelson, B L, F J Matejcik. 1995. Using Common Random Numbers for Indifference-Zone Selection and Multiple Comparisons in Simulation. *Management Science* **41**(12) 1935–1945.

O'Hagan, A. 1991. Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference* **29**(3) 245–260.

Oksendal, B. 2003. *Stochastic Differential Equations: An Introduction with Applications*. Springer, Berlin.

Paulson, E. 1952. On the comparison of several experimental categories with a control. *The Annals of Mathematical Statistics* **23**(2) 239–246.

Paulson, E. 1962. A sequential procedure for comparing several experimental categories with a standard or control. *The Annals of mathematical statistics* 438–443.

Picheny, V, D Ginsbourger, O Roustant, R T Haftka, N H Kim, Others. 2010. Adaptive designs of experiments for accurate approximation of a target region. *Journal of Mechanical Design* **132** 71008.

Powell, W B. 2007. *Approximate Dynamic Programming: Solving the curses of dimensionality*. John Wiley and Sons, New York.

Raiffa, H, R Schlaifer. 1968. *Applied Statistical Decision Theory*. M.I.T. Press.

Rasmussen, C E, Z Ghahramani. 2003. Bayesian monte carlo. *Advances in neural information processing systems* **15** 489–496.

Rasmussen, C E, C K I Williams. 2006. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA. URL `http://www.gaussianprocess.org/gpml`.

Rogers, Leonard C G. 2002. Monte Carlo valuation of American options. *Mathematical Finance* **12**(3) 271–286.

Ryzhov, I, W B Powell, P I Frazier. 2012. The Knowledge Gradient Algorithm for a General Class of Online Learning Problems. *Operations Research* **60**(1).

Sankaran, S, C Audet, A L Marsden. 2010. A method for stochastic constrained optimization using derivative-free surrogate pattern search and collocation. *Journal of Computational Physics* **20** 4664–4682.

Sankaran, S, A L Marsden. 2010. The impact of uncertainty on shape optimization of idealized bypass graft models in unsteady flow. *Physics of Fluids* **22** 121902.

Sankaran, S, A L Marsden. 2011. A stochastic collocation method for uncertainty quantification in cardiovascular simulations. *Journal of Biomechanical Engineering* **133** 31001.

Santner, T J, W I Notz, Bartel D.L. 2001. Robust Design and Analysis of Total Joint Replacements: Finite Element Model Experiments with Environmental Variables. *Journal of Biomechanical Engineering* **123** 239–246.

Santner, Thomas J, Brian J Williams, William I Notz. 2003. *The design and analysis of computer experiments*. Springer.

Sasena, Michael James. 2002. Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations. Phd thesis, University of Michigan.

Schruben, Lee W, Barry H Margolin. 1978. Pseudorandom Number Assignment in Statistically Designed Simulation and Distribution Sampling Experiments. *JASA* **73**(363) 504–525.

Scott, Warren, Peter I Frazier, Warren B Powell. 2011. The Correlated Knowledge Gradient for Simulation Optimization of Continuous Parameters Using Gaussian Process Regression. *SIAM Journal on Optimization* **21** 996–1026.

Shi, L, S Ólafsson. 2000. Nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability* **2**(3) 271–291.

Siegmund, D. 1985. *Sequential analysis: tests and confidence intervals*. Springer Series in Statistics, Springer-Verlag, New York.

SimOpt.org. 2011. . http://simopt.org/.

Sloane, N. 2007. The on-line encyclopedia of integer sequences. *Towards Mechanized Mathematical Assistants* 130.

Szechtman, R, E Yücesan. 2008. A new perspective on feasibility determination. *Proceedings of the 40th Conference on Winter Simulation*. Winter Simulation Conference, 273–280.

Tew, Jeff, James R Wilson. 1992. Validation of Simulation Analysis Methods for the {S}chruben-{M}argolin Correlation-Induction Strategy. *Operations Research* **40**(1) 87–103.

van Beers, W C M, J P C Kleijnen. 2008. Customized sequential designs for random simulation experiments: Kriging metamodeling and bootstrapping. *EJOR* **186**(3) 1099–1113.

Varaiya, P P, J C Walrand, C Buyukkoc. 1985. Extensions of the multiarmed bandit problem: The discounted case. *IEEE TRANS. AUTOM. CONTROL.* **30**(5) 426–439.

Villemonteix, J, E Vazquez, E Walter. 2009. An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization* **44**(4) 509–534.

Wald, A, J Wolfowitz. 1948. Optimum Character of the Sequential Probability Ratio Test. *The Annals of Mathematical Statistics* **19**(3) 326–339.

Wang, H, R Pasupathy, B W Schmeiser. 2013. Integer-Ordered Simulation Optimization using R-SPLINE: Retrospective Search with Piecewise-Linear Interpolation and Neighborhood Enumeration. *ACM TOMACS* **23**(3) in press.

Wang, Y, M C Fu, S I Marcus. 2010. Model-based evolutionary optimization. B Johansson, S Jain, J Montoya-Torres, J Hugan, E Yucesan, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 1199–1210.

Whittle, P. 1980. Multi-armed bandits and the Gittins index. *Journal of the Royal Statistical Society. Series B (Methodological)* **42**(2) 143–149.

Xie, J, P I Frazier. 2013a. Sequential Bayes-Optimal Policies for Multiple Comparisons with a Known Standard. *Operations Research* **61**(5) 1174–1189. doi:10.1287/opre.2013.1207.

Xie, J, P I Frazier. 2013b. Upper bounds on the Bayes-optimal procedure for ranking and selection with independent normal priors. *Proceedings of the 2011 Winter Simulation Conference*. 877–887. doi:10.1109/WSC.2013.6721479.

Xie, J, P I Frazier, S E Chick. 2013. Bayesian Optimization via Simulation with Pairwise Sampling and Correlated Prior Beliefs. in review.

Xie, J, P I Frazier, S Sankaran, A Marsden, S Elmohamed. 2012. Optimization of Computationally Expensive Simulations with Gaussian Processes and Parameter Uncertainty: Application to Cardiovascular Surgery. *The 50th Annual Allerton Conference on Communication, Control and Computing*. IEEE, 406–413.

Xu, J, B L Nelson, J Hong. 2010. Industrial strength COMPASS: A comprehensive algorithm and software for optimization via simulation. *ACM TOMACS* **20**(1) 3.

Yang, W N, B L Nelson. 1991. Using Common Random Numbers and Control Variates in Multiple-Comparison Procedures. *Operations Research* **39**(4) 583–591.

Zhou, E, M C Fu, S I Marcus. 2008. A particle filtering framework for randomized optimization algorithms. S J Mason, R R Hill, L Mönch, O Rose, T Jefferson, J W Fowler, eds., *Proc. Winter Simulation Conference*. IEEE, Piscataway, NJ, 647–654.