The Knowledge-Gradient Algorithm for Sequencing Experiments in Drug Discovery

Diana M. Negoescu Peter I. Frazier Warren B. Powell

May 26, 2010

Abstract

We present a new technique for adaptively choosing the sequence of molecular compounds to test in drug discovery. Beginning with a base compound, we consider the problem of searching for a chemical derivative of this molecule that best treats a given disease. The problem of choosing the molecules to test to maximize the expected quality of the best compound discovered may be formulated mathematically as a ranking and selection problem, in which each molecule is an alternative. We apply a recently developed algorithm, known as the knowledge-gradient algorithm. This algorithm uses correlations in our Bayesian prior distribution between the performance of different alternatives (molecules) to dramatically reduce the number of molecular tests required, but has heavy computational requirements that limit the number of alternatives that can be considered to a few thousand. We develop computational improvements that allow the knowledge-gradient method to consider much larger sets of alternatives, and demonstrate the method on a problem with 87120 alternatives.

1 Introduction

In drug discovery, medical researchers often begin with a single molecule that shows some promise for treating a given disease, and then test many variations of that molecule to find one that produces the best results. These variations are obtained by substituting atoms or groups of atoms at certain locations (sites) on the original molecule with other atoms or groups of atoms (substituents). The number of variations increases exponentially in the number of sites and substituents, and therefore the number of candidate compounds is usually extremely large. Synthesizing and testing a compound may require several days’ work and a significant investment of lab materials, which strictly limits the number of tests that a research team can perform. A critical problem is therefore deciding which compounds should be tested in order to most accurately and quickly find compounds with good disease-treating ability.

The problem of deciding which compounds to evaluate can be modeled mathematically as a ranking and selection problem. In this problem, we have a budget of measurements that we allocate sequentially to alternatives (molecular compounds in our case) so that, when we finish our experiments, we have collected the information needed to maximize our ability to find the best alternative. In deciding which compound to measure, we may use the fact that compounds
with similar structures often have similar properties. While a vast literature exists for ranking and selection problems in which the alternatives are treated independently, the use of structure within the set of alternatives has received much less attention. A recent ranking and selection algorithm that uses correlated Bayesian beliefs to take advantage of structure within the set of alternatives is the knowledge-gradient algorithm for correlated beliefs (KGCB) (Frazier et al. (2009)). However, the standard implementation of this algorithm requires storing and manipulating a covariance matrix whose dimension is the number of alternatives. In drug discovery, where one commonly has tens or hundreds of thousands of alternatives, this strategy is computationally expensive at best and often computationally infeasible.

In this paper, we represent beliefs about molecules using a linear, additive model. We take advantage of the structure of this model to develop two computational improvements to the knowledge-gradient algorithm that substantially streamline the procedure. When the number of molecules considered is large, as it almost always is during drug discovery, our new procedures improve computation time by several orders of magnitude. These improvements allow the KGCB algorithm to be used in applied settings for which the standard implementation is simply too slow to be useful. After introducing these new computation-saving procedures, we evaluate the performance of the KGCB algorithm using a previously published data set.

While in this paper we restrict our discussion to drug discovery, it is worth noting that these improvements have applications not just in drug discovery, but in any problem in which beliefs obey a linear additive model. One example is choosing the shortest path through a network when the length along a path is exactly or approximately equal to the sum of the lengths along each link. Some of the computational improvements we develop can also be applied to problems whose beliefs are only approximated by a linear additive model, such as identifying ads that produce the most ad-clicks based on a linear model using features from each ad.

There are two main approaches to ranking and selection: the frequentist approach, which is based entirely on observed data, and the Bayesian approach, which uses subjective a priori beliefs on the values of the compounds. We briefly review the Bayesian approach here. For a more thorough review of both approaches, see Swisher et al. (2003) and Kim & Nelson (2006).

Within the Bayesian approach, there are two main directions of research. The first is the Optimal Computing Budget Allocation (OCBA) (see, e.g., Chen et al. (1996)), in which the probability of correct selection is maximized and the posterior variance is reduced under the assumption that sampling will not change the posterior means. The second direction consists of Value of Information Procedures (VIP) (see, e.g., Chick & Inoue (2001b)) and is similar to OCBA in that it maximizes the improvement in a single stage’s allocation, but unlike OCBA, it considers the change in posterior mean when estimating the improvement. One of the first contributions in VIP is Gupta & Miescke (1996), which uses a one-step analysis under an independent normal prior distribution. Under this analysis, one chooses the measurement decision that would be optimal if
only one additional sample were allowed. The independent normal case was then further analyzed by Frazier et al. (2008), and extended to the unknown variance case by Chick et al. (2010). Frazier et al. (2008) uses the term ‘knowledge-gradient’ to refer to this one-step approach because the value of a single sample is the difference in implementation value between two different quantities of knowledge – the knowledge that one has before the sample, and the knowledge that one has after it. The knowledge-gradient approach of Gupta & Miescke (1996) is extended to correlated normal beliefs in Frazier et al. (2009) and its KGCB algorithm. With correlated beliefs, one may model how structural relationships between alternatives (such as those between chemically related molecules) cause the values of alternatives to relate to one another. These relationships allow learning about multiple alternatives from just a single measurement, often resulting in a dramatic improvement in measurement efficiency.

While an abundant literature in ranking and selection treats beliefs that are a priori independent across alternatives, correlated a priori beliefs have received much less attention. To our knowledge, Frazier et al. (2009) is the first to consider them for ranking and selection. There are, however, Bayesian approaches to ranking and selection using common random numbers (CRN), and correlated sampling with CRN induces correlated posteriors, even if the prior is independent. CRN is useful for ranking and selection because sampling two systems with positively correlated noise allows estimating which system is better with more accuracy than is possible with independent noise (see, e.g., Ehrlichman & Henderson (2008), which also provides a more general technique than CRN for sampling with correlation). Chick & Inoue (2001a) proposes a VIP approach for using CRN, while Fu et al. (2007) proposes an OCBA-based approach. In the indifference-zone ranking and selection and multiple comparisons literature, several methods allow CRN (Yang & Nelson 1991, Nelson & Matejcik 1995, Kim & Nelson 2001), but these methods are non-Bayesian and hence do not use prior distributions, correlated or otherwise. It is important to note that in the drug discovery problem we consider here, the samples are independent and only the prior distribution is correlated, which differs from using an independent prior distribution with correlated sampling. If the prior is independent, then we must sample each alternative at least once in order to do well in all but certain special cases, whether or not samples are correlated. It is only with a correlated prior that one may sample fewer than the full set of alternatives and still reasonably hope to find an alternative whose value is close to best.

Although correlated priors have appeared infrequently in Bayesian ranking and selection, some frequentist methods use linear models to compare alternatives’ values without requiring that every alternative be sampled, achieving conceptually similar improvements to those with correlated priors. These methods even allow infinite numbers of alternatives, as would occur if we were tuning a set of continuous parameters to maximize the output of a simulation model. For the use of such techniques in multiple comparisons, see Chapter 7 of Hsu (1996), and for their use in multi-armed bandits, see Kleinberg (2005). The use of linear designs within response surface
methods (Myers & Montgomery 2002) can also be understood in this way.

In contrast with their rarity within Bayesian ranking and selection, correlated prior distributions have a rich history within Bayesian global optimization (BGO), where Gaussian process prior distributions model the similarity between values of a continuous function evaluated at nearby points. Then, as in Bayesian ranking and selection, these beliefs are used to decide at which points to evaluate this function, with the ultimate goal of finding the function’s global maximum. There are two predominant methods for choosing measurements in BGO: maximal probability of improvement (MPI, also called $P^*$) (Kushner 1964, Stuckman 1988, Perevozchikov 1991) and expected improvement (EI) (Mockus 1972, Mockus et al. 1978, Jones et al. 1998). These two methods are most frequently applied when measurements are free from noise, but MPI was extended to the noisy case in Calvin & Zilinskas (2005), and EI was extended to the noisy case in Huang et al. (2006). The expected improvement criterion is similar to the knowledge-gradient method, except that EI considers the improvement in the function’s value only at the point measured, while KG considers improvements across the entire domain. KG methods were compared to EI methods for a discretized BGO problem in Frazier et al. (2009). KG performed slightly better than EI in the noise-free case, and significantly better in the noisy case. See Sasena (2002) Section 3.2 and Frazier (2010) for reviews of the BGO literature.

Within the medical community, the process of using statistical and computational tools to support drug discovery is called virtual or in silico screening. Although the literature on this subject has enjoyed massive growth in the past few years, only a small portion of it treats sequential designs. Most of it instead focuses on non-sequential prediction techniques using data collected in a single batch (see, e.g., Burbidge et al. (2001), Jorissen & Gilson (2005), Davies et al. (2006)), despite the fact that sequential designs can be much more efficient (Warmuth et al. 2003). One exception is Bajorath (2002), an early review of virtual screening, which differentiates between focused (which we call non-sequential) and sequential screening.

Previous mathematical work on sequential experimental design for drug discovery has been focused within the machine learning community, where it has been formulated as an active learning problem. Most frequently, active learning concerns itself with choosing which examples should be labeled in order to best train a classifier. Warmuth et al. (2003), with more mathematical details provided in Warmuth et al. (2002), takes this approach, where compounds that perform well in a laboratory test are labeled as “active,” while those that perform poorly are labeled “inactive.” These labels are assumed to be available without noise. Taking an alternative approach, De Grave et al. (2008) considers a version of the drug discovery problem closer to the one we consider here, where one observes real-valued activity levels for compounds. Rather than searching for the single best compound, the goal considered is finding the $k$ compounds with the largest activity levels, and measurements are assumed to be free from noise. Several empirical strategies are evaluated, including versions of MPI and EI.
An important part of any Bayesian ranking and selection algorithm for drug discovery is the prior distribution on the values of the candidate compounds. Several such prior distributions have been developed in the medicinal chemistry literature, where the collection of statistical models that predict biological activity from chemical structure are known collectively as Quantitative Structure Activity Relationship (QSAR) models. In our case, the biological activity of interest is the compound’s ability to treat a disease, as measured in a laboratory test. Examples of such laboratory tests include those that test a compound’s ability to kill diseased cells, or to inhibit a protein interaction believed to be critical to the disease’s progression.

The first attempt to quantify relationships between biological activity and chemical structure dates to 1868, when Crum-Brown and Fraser published the first formulation of a quantitative relationship between ‘physiological activity’ and ‘chemical structure’ (Crum-Brown & Fraser (1868)). The modern origins of the field lie in the Free-Wilson model (Free & Wilson (1964)) and the Hansch Analysis (Hansch & Fujita (1964)). In both methods, multiple linear regression is used to correlate activity on a laboratory test with a collection of compound features. In the Free-Wilson model these features are indicator variables indicating the presence or absence of substituents at given sites, and in the Hansch Analysis these features are physiochemical properties such as molecular weight and lipophilicity. More recently, support vector machines have been used together with much larger collections of compound features (see, e.g., Warmuth et al. (2003), Jorissen & Gilson (2005)). Many other statistical methods have been applied to QSAR, including cluster analysis, pattern recognition, principle component analysis, discriminant analysis, partial least squares, neural networks, and evolutionary algorithms. See (Grover et al. (2000)) for a review.

In this paper, we consider the Free-Wilson model. The Free-Wilson model assumes that the base molecule and each substituent contributes additively to the overall value of the compound. This contribution is assumed to be independent of the presence or absence of other substituents. We also consider a generalization of Free-Wilson that allows for deviations from perfect linearity. The structure of the Free-Wilson model allows us to derive two highly efficient implementations of the KGCB algorithm that can be used for sets of candidate compounds much larger than can be handled by the original algorithm. While the original implementation manipulates a belief on the values of the compounds, our new implementations manipulate beliefs on the parameters of our linear model. Since the number of compounds is exponential in the number of substituents per site, maintaining a belief in this way reduces computation and memory requirements substantially, allowing the KGCB algorithm to be used on problems with even hundreds of thousands of candidate compounds.

The paper is organized as follows. In section 2, we describe the ranking and selection problem and the relationships between the structure and value of an alternative. In Section 3 we review the original implementation of the KGCB algorithm and in Sections 4 and 5 we describe two computational improvements to this implementation. In Section 6 we present an empirical study.
of the KGCB algorithm on a previously published data set.

2 Model

We suppose that we have $M$ alternatives and a budget of $N$ measurements, and we wish to sequentially decide which alternatives to measure so that when we exhaust our budget of measurements we have maximized our ability to find the best alternative. We assume that samples from testing alternative $i$ are normally distributed with unknown mean $\vartheta_i$ and known variance $\lambda_i$, and conditionally independent of all other samples, given the unknown mean and the decision to sample that alternative. We write $\vartheta$ to indicate the column vector $(\vartheta_1, ..., \vartheta_M)'$. For a summary of all the symbols we use, please refer to Table 1 at the end of this section. We further assume that our belief about $\vartheta$ is distributed according to a multivariate normal prior distribution with mean vector $\mu_0$ and positive semi-definite covariance matrix $\Sigma_0$,

$$\vartheta \sim \mathcal{N}(\mu_0, \Sigma_0).$$

We assume that we have a budget of $N$ sampling decisions, $x^0, x^1, ..., x^{N-1}$. The measurement decision $x^n$ selects an alternative to test at time $n$ from the set $\{1, ..., M\}$. The measurement error $\epsilon^{n+1}$ is assumed to be normally distributed $\epsilon^{n+1} \sim \mathcal{N}(0, \lambda_{x^n})$ and independent conditional on $x^n$. Therefore, the resulting sample observation is $\hat{y}^{n+1} = \vartheta_{x^n} + \epsilon^{n+1}$. Through experiments, we try to learn the value of $\vartheta$, which is assumed to be fixed throughout the duration of the trials.

We define a filtration $(\mathcal{F}^n)_{n=0}^N$ as the sequence of sigma-algebras generated by the samples observed by time $n$ and the identities of their originating alternatives. More explicitly, $\mathcal{F}^n$ is the sigma-algebra generated by $x^0, \hat{y}^1, x^1, \hat{y}^2, ..., x^{n-1}, \hat{y}^n$. We write $E_n$ and $\text{Var}_n$ to indicate $E[\cdot | \mathcal{F}^n]$ and $\text{Var}[\cdot | \mathcal{F}^n]$, the conditional expectation and variance, respectively, taken with respect to $\mathcal{F}^n$. Then define $\mu^n := E_n[\vartheta]$, and $\Sigma^n := \text{Cov}[\vartheta | \mathcal{F}^n]$. Conditionally on $\mathcal{F}^n$, our posterior belief distribution on $\vartheta$ is multivariate normal with mean vector $\mu^n$ and covariance matrix $\Sigma^n$.

We define $\Pi$ to be the set of all possible policies satisfying our sequential requirement, that is, $\Pi := \{(x^0, ..., x^{N-1}) : x^n \in \mathcal{F}^n\}$. We let $\pi$ be a generic policy in $\Pi$ and we write $E^\pi$ to indicate the expectation taken when the policy is $\pi$.

After exhausting the budget of $N$ measurements, we select the alternative with the highest posterior mean. Our goal is to choose a measurement policy maximizing expected reward, which can be written as,

$$\sup_{\pi \in \Pi} \mathbb{E}^\pi \left[ \max_i \mu_i^N \right].$$

We now describe how the prior mean vector $\mu_0$ and covariance matrix $\Sigma_0$ are chosen according to the Free-Wilson model. Although we apply this work to drug discovery, where we think of “compounds,” “sites,” and “substituents,” we describe the model in the generic language of
“alternatives,” “dimensions” and “attributes.” These generic terms facilitate the observation that the Free-Wilson model is simply a linear model whose explanatory variables are 0 or 1. To define them, we suppose the existence of several dimensions, and state that each attribute may be associated with only one dimension. Each alternative is obtained through a specific choice of which single attribute, if any, is present in each dimension. In the context of drug discovery, an alternative is a compound, a dimension is a site, and an attribute is a substituent, where we consider the same atom or chemical group substituted at two different sites to be two different attributes.

2.1 The Free-Wilson Model

The Free-Wilson model (Free & Wilson (1964)) assumes that each attribute contributes additively to the value of the alternative. Denote by \( a_i \) the contribution of attribute \( i \), and by \( s^x \) a vector of 0’s and 1’s, with a 1 for every attribute that is present in alternative \( x \). Thus, \( s^x_i = 1 \) means that the \( i^{th} \) attribute is present in alternative \( x \). We denote by \( \zeta \) the value of the base alternative, which is the alternative obtained from taking \( s^x_i = 0 \) over every attribute \( i \). Let \( L(i) \) denote the dimension associated with attribute \( i \), and let \( k \) denote the total number of attributes. We restrict the \( s^x \) to specify at most one attribute associated with each dimension. That is, we require of each \( x \) that \( \sum_i s^x_i 1_{\{L(i)=l\}} \leq 1 \) for each dimension \( l \). Furthermore, we allow any \( s^x \) meeting this specification.

To illustrate these notions for drug discovery, we constructed a small toy example in Figure 1. This compound (alternative) has two sites (dimensions) \( X \) and \( Y \), at which atoms or groups of atoms (attributes) can be attached, creating new compounds. Assume there are two substituents, \( A \) and \( B \), which can be placed at site \( X \) and two substituents, \( C \) and \( D \), which can be placed at site \( Y \). Then, the original compound without any substituents attached (having hydrogen atoms at sites \( X \) and \( Y \)) can be represented by the vector \( s^{00} = (0, 0, 0, 0) \), where the first two zeros correspond to site \( X \) and the last two zeros correspond to site \( Y \). The compound resulting from attaching nothing at site \( X \) (keeping the hydrogen atom) and substituent \( C \) at site \( Y \) is represented by the vector \( s^{0C} = (0, 0, 1, 0) \). The compound resulting from attaching substituent \( B \) at site \( X \) and substituent \( C \) at site \( Y \) is represented by the vector \( s^{BC} = (0, 1, 1, 0) \). For this example, \( k = 4 \) (\( A, B, C, D \)), \( L(1) = L(A) = 0 \) (attribute \( A \) is absent), \( L(2) = L(B) = X \), \( L(3) = L(C) = Y \), \( L(4) = L(D) = 0 \).

The Free-Wilson model assumes that each attribute contributes additively to the value of the alternative, which may be expressed as

\[
\vartheta_x = \sum_i a_i s^x_i + \zeta.
\]

To exemplify, going back to the example in Figure 1, the value of the alternative having attribute
Under this model, if we sample alternative $x$, having attributes given by $s_{1x}, \ldots, s_{kx}$, the sample value would be of the form

$$\hat{y}_x = a_1 s_{1x} + \ldots + a_k s_{kx} + \zeta + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, \lambda_x)$ is independent measurement noise.

We suppose that we have an independent normal prior distribution on $\zeta$ and $a_1, \ldots, a_k$. Under this prior distribution, the mean $\mu_i$ of our belief about the value of alternative $i$ is

$$\mu_i^0 = \mathbb{E}[\zeta] + \sum_m s^i_m \mathbb{E}[a_m],$$

and the covariance $\Sigma_{ij}^0$ between the values of alternatives $i$ and $j$ is

$$\Sigma_{ij}^0 = \text{Cov}(i, j) = \text{Cov} \left( \sum_m a_m s^i_m + \zeta, \sum_{m'} a_{m'} s^{j}_{m'} + \zeta \right)$$

$$= \text{Var}(\zeta) + \sum_{m,m'} s^i_m s^{j}_{m'} \text{Cov}(a_m, a_{m'}) = \text{Var}(\zeta) + \sum_{m \in \mathcal{L}_{ij}} \text{Var}(a_m),$$

where $\mathcal{L}_{ij} = \{ l \in \{1, \ldots, k\} | s^i_l = s^{j}_l = 1 \}$ is the set of attributes common to alternatives $i$ and $j$.

The Free-Wilson model assumes that the contribution of each substituent is linear and additive. In practice, it is easy to envision substituents that interact with each other. For example, there might be a set of synergistic substituents, for which the compound containing all of them has a much larger value than compounds containing only a subset of them. Such behaviors may be handled with models that are linear in the parameters, but which include deviation terms. One such model is presented in the next section.

### 2.2 The General Model

We now generalize the Free-Wilson model, which assumes a perfectly additive structure, to allow some deviation from perfect additivity. These deviations are specified by terms $b_x$, with one such term for each alternative $x$. The resulting model is still a linear model, but with more terms. Under this model, the value of alternative $x$ is

$$\vartheta_x = \sum_i a_i s^x_i + b_x + \zeta.$$
Under our prior distribution, the terms $b_1, ..., b_M$ are normally distributed with mean 0 and variance $\sigma^2_b$. Additionally, they are independent of each other as well as from the $a_i$ and $\zeta$ terms. Under this structure, the covariance $\Sigma_{ij}^0$ between the values of alternatives $i$ and $j$ is

$$
\Sigma_{ij}^0 = \text{Cov}(i, j) = \text{Cov}\left(\sum_m a_m s^i_m + \zeta + b_i, \sum_{m'} a_{m'} s^j_{m'} + \zeta + b_j\right)
$$

$$
= \text{Var}(\zeta) + \sum_{m,m'} s^i_m s^j_{m'} \text{Cov}(a_m, a_{m'}) + \sigma^2_b \mathbb{1}_{\{i=j\}}
$$

$$
= \text{Var}(\zeta) + \sum_{m \in \mathcal{L}_{ij}} \text{Var}(a_m) + \sigma^2_b \mathbb{1}_{\{i=j\}}.
$$

Thus, under this general model, the covariance matrix of the values of the alternatives under our prior distribution is obtained by taking the covariance matrix from the Free-Wilson model and adding $\sigma^2_b$ to the diagonal entries. Since $\mathbb{E}[b_x] = 0$ for all $x$, the mean $\mu^0$ of our belief about the values of the alternatives is the same as it is under the Free-Wilson model.

We see a spectrum of behaviors from this model over different values of $\sigma^2_b$. When $\sigma^2_b = 0$, the model is identical to the Free-Wilson model in Section 2.1. When $\sigma^2_b$ is non-zero but still significantly smaller that $\text{Var}(a_i)$, it models a situation in which the values of the alternatives are well-approximated by an additive structure, but also understands that there may be small deviations from it. When $\sigma^2_b$ is very large, and in particular much larger than the $\text{Var}(a_i)$ terms, the deviation from additivity is quite large, correlations between the alternatives are very weak, and our belief is similar to one that is independent across alternatives.

It is in this second regime, where $\sigma^2_b$ is non-zero but not too large, that this general model is likely to be most useful. It allows exploiting the additive structure of a problem to learn quickly, while simultaneously allowing flexibility in its understanding that the alternatives’ true values may not obey this structure perfectly.

In the third regime, where $\sigma^2_b$ is large, we essentially have an independent prior distribution, and learning the value of one alternative teaches us almost nothing about the value of the other alternatives. This makes it very difficult to learn in situations with large numbers of alternatives, because in order to come close to finding a good alternative, we need to make at least as many measurements as the number of $b$’s. In such situations, we must either find other non-additive structure in the problem, or resign ourselves to making a number of measurements that is larger than the number of alternatives.

Although the problem is inherently quite difficult when $\sigma^2_b$ is large, often requiring many measurements to find good alternatives, the KGCB policy that we present below may still work well relative to the difficulty of the problem. Indeed, the case of independent beliefs was considered in Frazier et al. (2008), and the KG policy was shown to work well compared to other state-of-the-art policies.
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>Number of alternatives (in drug discovery, total number of compounds)</td>
</tr>
<tr>
<td>$N$</td>
<td>Number of measurements in our budget</td>
</tr>
<tr>
<td>$\vartheta_i$</td>
<td>Unknown mean of alternative $i$</td>
</tr>
<tr>
<td>$\lambda_i$</td>
<td>Known variance of alternative $i$</td>
</tr>
<tr>
<td>$\vartheta$</td>
<td>Column vector $(\vartheta_1, ..., \vartheta_M)'$</td>
</tr>
<tr>
<td>$\mu^n, \Sigma^n$</td>
<td>Mean and covariance of posterior distribution on $\vartheta$ after $n$ measurements</td>
</tr>
<tr>
<td>$x^i$</td>
<td>Sampling decision at time $i$, $x^i \in {1, ..., M}$</td>
</tr>
<tr>
<td>$\epsilon_{n+1}$</td>
<td>Measurement error of alternative $x^n$, $\epsilon_{n+1} \sim \mathcal{N}(0, \lambda_{x^n})$</td>
</tr>
<tr>
<td>$\hat{y}_{n+1}$</td>
<td>Sampling observation from measuring alternative $x^n$, $\hat{y}<em>{n+1} = \vartheta</em>{x^n} + \epsilon_{n+1}$</td>
</tr>
<tr>
<td>$\Pi$</td>
<td>Set of all possible policies ${(x^0, ..., x^{N-1}) : x^n \in \mathcal{F}^n}$</td>
</tr>
<tr>
<td>$S^n$</td>
<td>Defined as the pair $(\mu^n, \Sigma^n)$</td>
</tr>
<tr>
<td>$e_x$</td>
<td>Column vector of 0’s with a 1 at position $x$</td>
</tr>
<tr>
<td>$\hat{\sigma}(\Sigma, x)$</td>
<td>Defined as $\frac{\Sigma_{xx}}{\sqrt{\lambda_{x^n} + \Sigma_{xx}}}$</td>
</tr>
<tr>
<td>$\mu_{KG,n}$</td>
<td>Knowledge gradient factor, defined as $\max_x \mathbb{E}_n [\max_i \mu_i^n</td>
</tr>
<tr>
<td>$a_i$</td>
<td>Contribution of attribute (in drug discovery, substituent) $i$</td>
</tr>
<tr>
<td>$s^x$</td>
<td>Row vector of 0’s and 1’s. A 1 indicates that the $i^{th}$ attribute is present in alternative $x$</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>Value of the base alternative (with zeros in all locations)</td>
</tr>
<tr>
<td>$L(i)$</td>
<td>Location of substituent $i$</td>
</tr>
<tr>
<td>$A(l)$</td>
<td>Vector of substituents that may be placed at location $l$</td>
</tr>
<tr>
<td>$k$</td>
<td>Total number of attributes</td>
</tr>
<tr>
<td>$b_x$</td>
<td>Deviation term associated with alternative $x$</td>
</tr>
<tr>
<td>$\sigma_b$</td>
<td>Standard deviation of the $b_x$ terms</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Vector of attribute values $(\zeta, a_1, a_2, ..., a_k)$</td>
</tr>
<tr>
<td>$X$</td>
<td>Matrix in which each row is an alternative</td>
</tr>
<tr>
<td>$\tilde{x}^n$</td>
<td>Column vector of 0’s and 1’s encoding alternative $x^n$</td>
</tr>
<tr>
<td>$\gamma^n$</td>
<td>Defined as $\lambda_{x^n} + (\tilde{x}^n)^T C^n \tilde{x}^n$</td>
</tr>
<tr>
<td>$\hat{\epsilon}_{n+1}$</td>
<td>Defined as $y_{n+1} - (\theta^n)^T \hat{x}^n$</td>
</tr>
<tr>
<td>$\theta, C$</td>
<td>Mean and covariance of posterior distribution on $\alpha$ ($\alpha \sim \mathcal{N}(\theta, C)$)</td>
</tr>
<tr>
<td>$\theta^n, C^n$</td>
<td>Mean and covariance of posterior distribution on $\alpha$ after $n$ measurements</td>
</tr>
</tbody>
</table>

Table 1: Table of notation
3 The KG Algorithm with Correlated Beliefs

The Knowledge-Gradient with Correlated Beliefs (KGCB) policy, as introduced in Frazier et al. (2009), measures the alternative that attains the maximum in

\[
\nu^{KG,n} = \max_x \mathbb{E}_n \left[ \max_i \mu_{i,n+1} | S^n = s, x^n = x \right] - \max_i \mu_{i,n},
\]

where \( S^n := (\mu^n, \Sigma^n) \) parameterizes the posterior belief distribution at measurement \( n \). The knowledge-gradient (KG) factor, \( \nu^{KG,n} \), represents the incremental value obtained from measuring a particular alternative \( x \).

After each alternative is measured, we obtain a posterior distribution on \( \vartheta \) that depends on which alternative was measured, \( x^n \), its sampled value, \( \hat{y}^{n+1} \), and our belief on \( \vartheta \) prior to sampling, which is parameterized by \( \mu^n \) and \( \Sigma^n \). This posterior distribution may be calculated using standard results for normal sampling with a multivariate normal prior distribution (see, e.g., Gelman et al. (2004)) as

\[
\mu^{n+1} = \mu^n + \frac{\hat{y}^{n+1} - \mu^n \Sigma_n e_x}{\lambda_x + \Sigma_n e_x e_x'}, \quad \Sigma^{n+1} = \Sigma^n - \frac{\Sigma_n e_x e_x' \Sigma_n}{\lambda_x + \Sigma_n e_x e_x'},
\]

where \( e_x \) is a column \( M \)-vector with a single 1 at index \( x \) and the rest 0s.

We now describe the time-\( n \) conditional distribution of \( \mu^{n+1} \), which allows us to compute (1). This distribution is multivariate normal, with mean given by the tower property as

\[
\mathbb{E}_n[\mu^{n+1}] = \mu^n \quad \text{and covariance given by the conditional variance formula as } \tilde{\sigma}(\Sigma^n, x^n) = \frac{\Sigma e_x e_x' \Sigma^n}{\lambda_x + \Sigma e_x e_x'},
\]

The details of this computation are given in Frazier et al. (2009).

Thus, the time-\( n \) conditional distribution of \( \mu^{n+1} \) is the same as that of \( \mu^{n+1} = \mu^n + \tilde{\sigma}(\Sigma^n, x^n)Z \), where \( Z \) is any independent one-dimensional standard normal random variable. This allows us to rewrite (1) as

\[
x^{KG}(s) = \arg \max_x \mathbb{E} \left[ \max_i (\mu_{i,n} + \tilde{\sigma}(\Sigma^n, x^n)Z) | S^n, x^n = x \right] - \max_i \mu_{i,n}
= \arg \max_x h(\mu^n, \tilde{\sigma}(\Sigma^n, x)).
\]

Here, \( h : \mathbb{R}^M \times \mathbb{R}^M \rightarrow \mathbb{R} \) is defined by \( h(p, q) = \mathbb{E}[\max_i p_i + q_i Z] - \max_i p_i \), where \( p \) and \( q \) are deterministic \( M \)-vectors, and again \( Z \) is any one-dimensional standard normal random variable.

Frazier et al. (2009) provides a method for computing \( h(p, q) \), which is summarized in Algorithm 3.2. This algorithm in turn uses Algorithm 3.1 in an inner loop. In Algorithm 3.2, the components of \( p \) and \( q \) are sorted, and then some are dropped, resulting in new vectors \( p' \) and \( q' \) of length \( M' \), and a sequence \( c' \). These quantities are then used to calculate \( h(p, q) \) via

\[
h(p, q) = \sum_{i=1}^{M'-1} (q'_{i+1} - q'_i) f(-|c'_i|),
\]
where the function \( f \) is given by \( f(z) = \varphi(z) + z\phi(z) \), and where \( \varphi \) is the normal cumulative distribution function and \( \phi \) is the normal density.

**Algorithm 3.1:** Compute sequence \( c \) and undominated set \( A \) (Inputs: \( p, q \))

\[
c_0 \leftarrow -\infty, c_1 \leftarrow +\infty, A \leftarrow \{1\}
\]

for \( i \leftarrow 1 \) to \( M - 1 \)
\[
c_{i+1} \leftarrow +\infty
\]

repeat
\[
j \leftarrow A[end(A)]
c_j \leftarrow (p_j - p_{i+1})/(q_{i+1} - q_j)
\]

\[
k = A[end(A) - 1]
\]

if \( c_j \leq c_k \),
then
\[
A \leftarrow A(1, ..., end(A) - 1)
\]

loopdone \( \leftarrow \) false
else
loopdone \( \leftarrow \) true
until loopdone
\[
A \leftarrow (A, i + 1)
\]

**Algorithm 3.2:** Compute helper function \( h \) (Inputs: \( p, q \))

sort \((p_i, q_i)_{i=1}^M\) such that \( p_i \) are in non-decreasing order and ties broken such that \( p_i \leq p_{i+1} \) if \( q_i = q_{i+1} \)
for \( i \leftarrow 1 \) to \( M - 1 \)
\[
\text{if } q_i = q_{i+1} \text{ then}
\]

Remove entry \( i \) from \((p_i, q_i)_{i=1}^M\)
Use algorithm 3.1 to compute \( c \) and \( A \) from \( p, q \)
\[
p \leftarrow p[A], q \leftarrow q[A], c \leftarrow (c[A], +\infty), M \leftarrow \text{length}(A)
\]

return \((\log)(\sum_{i=1}^{M-1} (q_{i+1} - q_i) f(\lvert c_i \rvert))\)

Using Algorithm 3.2, we can compute \( h(\mu^n, \tilde{\sigma}(\Sigma^n, x)) \) for any vectors \( \mu^n \) and \( \tilde{\sigma}(\Sigma^n, x) \). This then allows computing the KG factor via (1) for each alternative, the largest of which gives the measurement decision of the KGCB policy. This is summarized in Algorithm 3.3.

**Algorithm 3.3:** KGCB1 Algorithm (Inputs: \( \mu^n, \Sigma^n \))

for \( x \leftarrow 1 \) to \( \text{length}(\mu^n) \)
\[
p \leftarrow \mu^n
\]
\[
q \leftarrow \tilde{\sigma}(\Sigma^n, x)
\]

\[\nu \leftarrow h(p, q) \text{ % use Algorithm 3.2}\]

if \( x = 1 \) or \( \nu > \nu^* \)
then
\[
\nu^* \leftarrow \nu, x^* \leftarrow x
\]

Within Algorithm 3.3, Algorithm 3.2 executes \( M \) times. Within one execution of Algorithm 3.2, the sort has complexity \( O(M \log M) \) and Algorithm 3.1 has complexity \( O(M) \). Thus,
the most computationally demanding step within Algorithm 3.2 is the sort, and the overall complexity of the KGCB algorithm as computed by Algorithm 3.3 is $O(M^2 \log M)$.

In drug discovery, families of molecules often contain tens or hundreds of thousands of compounds, which makes this algorithm for computing KGCB computationally infeasible. Thus, even though we might very much wish to use the KGCB method to reduce the number of physical measurements that need to be taken, the computational requirements of actually computing the KGCB measurement decisions under the standard algorithm, Algorithm 3.3, preclude doing so in most cases. The next section describes a first improvement to the standard algorithm that dramatically reduces the computational requirements and allows computing KGCB for large numbers of linearly related alternatives such as those encountered in drug discovery.

4 The KGCB$_2$ Algorithm

In this section, we present the first of two computational improvements to the standard implementation of KGCB that exponentially reduce the computational and storage requirements of the algorithm. The essential idea behind this improvement is to maintain a belief on the attributes themselves instead of on the (much larger) set of alternatives. This greatly improves the efficiency of the KGCB algorithm.

We first describe this improvement in the context of the Free-Wilson model from Section 2.1 in Sections 4.1 and 4.2, and then we extend this improvement to the general model from Section 2.2 in Section 4.3.

4.1 Beliefs On Attributes

In this section we describe how one may maintain a belief on attributes rather than on alternatives in the Free-Wilson model of Section 2.1.

Let $\alpha$ be the vector of attribute values $\alpha = (\zeta, a_1, ..., a_k)$ containing the value of the base molecule and of each substituent, where chemically identical substituents at different locations are given different indices. We assume the linear additive model for modeling structure-value relationships from Section 2.1 and we let $X$ be a matrix comprised of rows representing the alternative. Each row of $X$ is a vector of 0’s and 1’s of the same length as $\alpha$, and each 1 indicates an attribute that is present in the alternative. The value of this attribute is the corresponding component in $\alpha$. In the context of drug discovery, this row contains a single 1 in the first entry to indicate that the base molecule (whose value is $\zeta$) is present, and then the subsequent entries contains a 1 for each substituent present. Thus, this row is a 1 followed by the vector $s^x$ (defined in Section 2.1) corresponding to the molecule $x$ being represented. With these definitions, the true value of the alternatives is $\vartheta = X\alpha$.

Any multivariate normal belief on $\alpha$ induces a multivariate normal belief on $\vartheta$. If we have a
multivariate normal belief on \( \alpha \) with the \( k+1 \)-dimensional mean vector \( \theta \) and the \( (k+1) \times (k+1) \) covariance matrix \( C \),

\[
\alpha \sim \mathcal{N}(\theta, C),
\]

we then have the mean of the values of the alternatives given by \( E[\vartheta] = X\theta \). The covariance between the values of the alternatives is given by

\[
\text{Cov}(\vartheta_i, \vartheta_j) = \text{Cov}(\sum_k X_i^k \alpha_k, \sum_k X_j^k \alpha_k) = \sum_{k,k'} X_i^k X_j^{k'} C_{k,k'}
\]

\[
= e_i^T X C X^T e_j,
\]

where \( e_i \) is, as before, a column vector of length the size of our alternative database, with a 1 on position \( i \) and zeros everywhere else. Thus, the belief induced on \( \vartheta \) by (2) is

\[
\vartheta \sim \mathcal{N}(X\theta, XCX^T).
\]

Having described how a generic multivariate normal prior distribution on \( \alpha \) induces a multivariate normal prior distribution on \( \vartheta \), we begin with a prior distribution on \( \alpha \) with mean vector \( \theta_0 \) and covariance matrix \( C_0 \). Thus, the parameters of the induced prior distribution on \( \vartheta \) are

\[
\mu_0 = X\theta_0 \quad \text{and} \quad \Sigma_0 = XC_0 X^T.
\]

There exists a recursive expression for \( \mu^n \) and \( \Sigma^n \) that is similar to the recursive expression for \( \mu^n \) and \( \Sigma^n \) in Section 3. Before providing this expression, we first introduce some additional notation. Let \( x^n = (\tilde{x}_0^n; \tilde{x}_1^n; ...; \tilde{x}_k^n)^T \) be a column vector of 0s and 1s describing the alternative \( x^n \in \{1, ..., M\} \) that was measured at iteration \( n \), where \( \tilde{x}_0^n = 1 \) represents the presence of the base alternative, and \( \tilde{x}_i^n = 1 \) for those attributes \( i \) present in alternative \( x^n \) and 0 otherwise. Additionally, define \( \tilde{\epsilon}^{n+1} = y^{n+1} - (\theta^n)^T \tilde{x}^n \) and \( \gamma^n = \lambda x^n + (\tilde{x}^n)^T C^n \tilde{x}^n \). Then, the following updating equations result from standard expressions for normal sampling of linear combinations of attributes (see, e.g., Powell (2007)),

\[
\begin{aligned}
\theta^{n+1} &= \theta^n + \frac{\tilde{\epsilon}^{n+1}}{\gamma^n} C^n \tilde{x}^n, \\
C^{n+1} &= C^n - \frac{1}{\gamma^n} \left( C^n \tilde{x}^n (\tilde{x}^n)^T C^n \right).
\end{aligned}
\]

When the number of substituents is large, maintaining \( \mu^n \) and \( \Sigma^n \) through this recursive expression is much more efficient than maintaining \( \mu^n \) and \( \Sigma^n \) through the recursive expression in Section 3. This is because the dimension of \( \Sigma^n \) is equal to the number of alternatives, which grows exponentially with the number of substituents per site. For a set of compounds with 5 sites and 9 possible substituents at each site (to make 10 possible choices at each site, including the possibility of attaching no substituent) the number of possible compounds is \( 10^5 \), compared to
only 45 total substituents. In this case, $\Sigma^n$ is $10^5 \times 10^5$ while $C^n$ is only $46 \times 46$ (we add 1 to 45 to account for the base molecule).

### 4.2 Implementation of the KGCB\textsubscript{2} Algorithm

Because $\theta^n$ and $C^n$ are much smaller and easier to maintain than $\mu^n$ and $\Sigma^n$, there is significant computational and storage savings to be achieved by calculating the KGCB policy from these inputs directly rather than from $\mu^n$ and $\Sigma^n$.

To accomplish this, we recall from Section 4.1 that $\mu^n$ and $\Sigma^n$ may be written in terms of $\theta^n$ and $C^n$ as $\mu^n = X\theta^n$ and $\Sigma^n = XC^nX^T$. We also recall that $\Sigma^n$ enters into the computation of the KG-factor for alternative $x$, $\nu^{KG}_x = h(\mu^n, \tilde{\sigma}(\Sigma^n, x))$, only through $\tilde{\sigma}(\Sigma^n, x)$. This quantity is given by $\tilde{\sigma}(\Sigma^n, x) = \Sigma^n e_x / \sqrt{\lambda_x + \Sigma^{xx}}$, which depends only upon row $x$ in $\Sigma^n$, and not the entire matrix. To facilitate describing these computations, we define $A_{x, \cdot}$ to be row $x$ from generic matrix $A$. Then, the required row of $\Sigma^n$ may be computed from $C^n$ as

$$\Sigma^n_{x, \cdot} = (XC^n X^T) e_x = (XC^n) X^T e_x.$$

By calculating row $x$ of $\Sigma^n$ from $C^n$, then calculating $\tilde{\sigma}(\Sigma^n, x)$ from this row vector, and then computing the KG-factor from $\tilde{\sigma}(\Sigma^n, x)$ and $\mu^n$, we obtain the KG-factor while completely avoiding any computations with matrices of size $M \times M$. This is summarized in Algorithm 4.1.

**Algorithm 4.1:** KGCB\textsubscript{2} Algorithm (Inputs: $\theta^n, C^n, X$)

```plaintext
\begin{align*}
\mu^n &\leftarrow X\theta^n \\
B &\leftarrow XC^n \\
\text{for } x &\leftarrow 1 \text{ to } M \\
\Sigma^n_{x, \cdot} &\leftarrow BX^T \cdot \\
a &\leftarrow \mu^n \\
b &\leftarrow \Sigma^n_{x, \cdot} / \sqrt{\lambda_x + \Sigma^{xx}} \\
\nu &\leftarrow h(a, b) \% \text{ use Algorithm 3.2} \\
\text{if } x &\leftarrow 1 \text{ or } \nu > \nu^* \\
\text{then} \\
\nu^* &\leftarrow \nu, x^* \leftarrow x
\end{align*}
```

Algorithm 4.1 is similar to Algorithm 3.3, with the main differences being the first operation, which retrieves the mean belief on alternatives from the mean of the belief on attributes, and the second and third operations (the latter being the first step inside the loop), which together retrieve the covariance of the belief on alternatives from the covariance of the belief on attributes. The $B$ matrix used in the second and third operations caches $XC^n$, which does not depend on the measurement $x$ being considered in the loop.

This algorithm significantly improves upon Algorithm 3.3 because it computes only $\Sigma^n_{x, \cdot}$, a column vector, instead of the full matrix $\Sigma^n$. This is particularly significant when there are many alternatives. If there were $10^6$ alternatives, we would create a vector of size $10^6 \times 1$ instead of a matrix of size $10^6 \times 10^6$. 

---

15
4.3 Extension to the General Model

This improvement, which has been described thus far in the context of the Free-Wilson model, can also be implemented for the general model.

To do so, we must maintain a belief about the \(b_x\) terms together with our belief about \(\zeta\) and the \(a_i\) terms. Although the number of \(b_x\) terms is equal to the number of compounds, which is generally very large, we need only maintain a mean vector and covariance matrix for only those \(b_x\) terms corresponding to alternatives that we have measured. If we have not measured a particular alternative \(x\) by time \(n\), then our posterior belief distribution on \(b_x\) will have the same marginal distribution that it had under the prior distribution, and will remain independent of \(\zeta\), the \(a_i\) terms, and all other deviation terms. Thus, by explicitly maintaining a belief about \(\zeta\), \(a_1, \ldots, a_k\), and only those \(b_x\) terms for compounds we have measured, we can reconstruct our belief about those deviation terms we are not explicitly tracking as needed.

Toward this end, let us define a vector \(\alpha_n\) that contains \(\alpha = (\zeta, a_1, \ldots, a_k)\), and the \(b_x\) for \(x\) ranging over the unique alternatives in \(x^0, \ldots, x^{n-1}\). This vector plays the role that \(\alpha\) plays in Section 4.1. Let \(\theta^n\) and \(C^n\) be the mean and variance of our time \(n\) posterior belief distribution on \(\alpha_n\). Note that \(\alpha_0 = \alpha\), \(\theta^n = (\mathbb{E}[\zeta], \mathbb{E}[a_1], \ldots, \mathbb{E}[a_k])\) and \(C^n\) is a diagonal matrix whose diagonal is \((\mathbb{E}[\zeta], \mathbb{E}[a_1], \ldots, \mathbb{E}[a_k])\).

Before providing recursive expressions for \(\theta^n\) and \(C^n\), \(C^{n-1}\), we first define two quantities, \(\hat{\theta}^{n-1}\) and \(\hat{C}^{n-1}\). If we have previously measured alternative \(x^n\), so \(x^n \in \{x^0, \ldots, x^{n-1}\}\), then let \(\hat{\theta}^{n-1} = \theta^{n-1}\) and \(\hat{C}^{n-1} = C^{n-1}\). If we have not previously measured \(x^n\), then let \(\hat{\theta}^{n-1}\) be the column vector obtained by appending a scalar 0 to \(\theta^{n-1}\), and let \(\hat{C}^{n-1}\) be the \((1+k+n) \times (1+k+n)\) matrix obtained from the \((k+n) \times (k+n)\) matrix \(C^{n-1}\) by adding one extra row and column after the last row and column of \(C^{n-1}\). This extra row and column is all 0s, except for the diagonal entry, which is \(a_j^2\). These quantities \(\hat{\theta}^{n-1}\) and \(\hat{C}^{n-1}\) are constructed so that our time \(n-1\) posterior belief distribution on \(\alpha_n\) is \(N(\hat{\theta}^{n-1}, \hat{C}^{n-1})\). Additionally, let \(\tilde{x}^n\) be a column vector of 0s and 1s, with a 1 at exactly those indices of \(\alpha_n+1\) for which the alternative \(x^n\) contains the corresponding base alternative, substituent, or deviation term. We also define \(\tilde{c}^{n+1} = y^{n+1} - (\hat{\theta}^n)^T \tilde{x}^n\) and \(\gamma^n = \lambda x^n + (\tilde{x}^n)^T \tilde{c}^{n} \tilde{x}^n\). With these definitions, we may update \(\theta^{n+1}\) and \(C^{n+1}\) recursively from \(\theta^n\) and \(C^n\) (via \(\hat{\theta}^n\) and \(\hat{C}^{n}\)) as

\[
\begin{align*}
\theta^{n+1} &= \hat{\theta}^n + \frac{\tilde{c}^{n+1}}{\gamma^n} \tilde{x}^n, \\
C^{n+1} &= \hat{C}^{n} - \frac{1}{\gamma^n} (\tilde{c}^{n} \tilde{x}^n (\tilde{x}^n)^T \hat{C}^{n}).
\end{align*}
\]

These updating equations allow us to maintain our belief about \(\alpha_n\) in a computationally efficient way, analogously to the way in which we were able to recursively maintain about \(\alpha\) in Section 4.1.

We now show how KG factors may be computed from a belief on \(\alpha_n\) parameterized by \(\theta^n\) and \(C^n\). Since the KG factor is given by \(\nu_x = h(\mu^n, \bar{\sigma}(\Sigma^n, x))\), it is enough to compute \(\mu^n\) and
\[ \tilde{\sigma}(\Sigma^n, x) \] efficiently (without computing the much larger matrix \( \Sigma^n \)), and then to use the standard implementation of \( h \). The first term, \( \mu^n \), does not depend on \( x \) and is given by

\[ \mu^n = X^n \theta^n, \]

where \( X^n \) is a \( M \times |\alpha_n| \) matrix of 0s and 1s whose each row corresponds to an alternative, and has a 1 for the base alternative, and each substituent and deviation term from \( \alpha_n \) contained within the alternative. To compute the second term, \( \tilde{\sigma}(\Sigma^n, x) \), fix \( x^n = x \) and the corresponding \( \alpha_{n+1} \) and \( \tilde{C}^n \) resulting from this choice of \( x^n \). Let \( \tilde{X}^n \) be a \( M \times |\alpha_{n+1}| \) matrix that is similar to \( X^n \), except that it maps alternatives to components of \( \alpha_{n+1} \) rather than \( \alpha_n \). That is, each row of \( \tilde{X}^n \) corresponds to an alternative, and has a 1 for the base alternative and each substituent and deviation term from \( \alpha_{n+1} \) contained within the alternative. Then, observe that the beliefs about those \( b_x \) not included in \( \alpha_{n+1} \) will not change as a result of measuring \( x^n \), and so \( \tilde{\sigma}(\Sigma^n, x^n) \), which is the standard deviation of the change in beliefs about the values of the alternatives, is not affected by these deviation terms not included in \( \alpha_{n+1} \). Thus, we can compute \( \tilde{\sigma}(\Sigma^n, x^n) \) by dropping these left-out deviation terms. In such a model in which these deviation terms outside \( \alpha_{n+1} \) have been left out of the model, the \( x^{th} \) column of \( \Sigma^n \) is

\[ \Sigma^n_{x, \cdot} = (\tilde{X}^n \tilde{C}^n)(\tilde{X}^n_{x, \cdot})^T, \]

and \( \tilde{\sigma}(\Sigma^n, x^n) \) may be computed from this vector via \( \tilde{\sigma}(\Sigma^n, x) = \Sigma^n_{x, x}/\sqrt{\lambda_x + \Sigma^n_{xx}} \). The resulting method of computing the KGCB policy is summarized below in Algorithm 4.2.

**Algorithm 4.2: General KGCB₂ Algorithm (Inputs : \( \theta^n, C^n, X^n \))**

\[
\begin{align*}
\mu^n &\leftarrow X^n \theta^n \\
\text{for } x \leftarrow 1 \text{ to } M \\
&\text{Compute } \tilde{C}^n \text{ from } C^n \text{ and } x. \\
&\Sigma^n_{x, \cdot} \leftarrow (\tilde{X}^n \tilde{C}^n)(\tilde{X}^n_{x, \cdot})^T \\
&a \leftarrow \mu^n \\
b &\leftarrow \Sigma^n_{x, x}/\sqrt{\lambda_x + \Sigma^n_{xx}} \\
\nu &\leftarrow h(a, b) \% \text{ use Algorithm 3.2} \\
\text{if } x = 1 \text{ or } \nu > \nu^* \text{ then} \\
&\nu^* \leftarrow \nu, x^* \leftarrow x
\end{align*}
\]

With these expressions, we may compute the KG factor for alternative \( x \) in the general model without explicitly computing a covariance matrix \( \Sigma^n \). The dimensionality of the objects \( \theta^n \) and \( C^n \) that we must retain in memory is the sum of \( 1 + k \) together with the number of unique alternatives measured. When the number of measurements that can be made is much smaller than the number of alternatives, as it is in drug discovery, this is a significant savings.
5 The KGCB\textsubscript{3} Algorithm for the Free-Wilson Model

We now return from the general model to consider only the Free-Wilson model of Section 2.1. When using this model, we may compute the KGCB policy with even greater efficiency than described thus far. This further improvement has at its heart Proposition 1 below.

This proposition states that the calculation of each KG factor, which would ordinarily require considering the entire family of molecules, may be decomposed into the sum of a set of much smaller and easier to compute KG factors. Each of these smaller KG factors is the KG factor that would result from a base molecule with substituents only at a single site. Since the number of compounds grows exponentially with the number of substituents per site, and the complexity of computing a KG factor scales as $O(s \log s)$ where $s$ is the number of substituents considered, this decomposition dramatically reduces the computational effort required to compute a KG factor. While the KGCB policy still needs to calculate the KG factor corresponding to each possible measurement to find the largest one, faster calculation of each KG factor makes the overall computation much faster as well.

Before stating the proposition, recall that $\theta_{n}^{i} = \mathbb{E}_{n}[a_{j}]$ and $\theta_{0}^{n} = \mathbb{E}_{n}[\zeta]$ are the means of the time-$n$ posterior belief distribution on the value of substituent $j$ and the base molecule respectively, and that $L(j)$ is the site at which substituent $j$ may be placed. Additionally, let $A_{\ell} \in \{i : L(i) = \ell\}$, so that $A_{\ell}$ is a vector containing those substituents that may be placed at site $\ell$.

**Proposition 1.** Under the Free-Wilson model,

$$\nu_{x}^{n,KG} = \sum_{\ell} h \left( (\theta_{i}^{n})_{i \in A(\ell) \cup \{-1\}}, (\tilde{\sigma}_{x}^{n})_{i \in A(\ell) \cup \{-1\}} \right),$$

where $(\tilde{\sigma}_{x}^{n})_{n}^{2} := \text{Var}_{n}[\theta_{i}^{n+1} \mid x^{n} = x]$ for $i > 0$ and $\theta_{n}^{-1} = \tilde{\sigma}_{x,-1}^{n} = 0$.

**Proof:** We write

$$\max_{x'} \mu_{x'}^{n+1} = \max_{x'} \mathbb{E}_{n+1} \left[ \sum_{j} a_{j} s_{j}^{x'} + \zeta \right] = \max_{x'} \sum_{i} \theta_{i}^{n+1} s_{i}^{x'} + \theta_{0}^{n+1}$$

$$= \max_{x'} \sum_{\ell} \left( \sum_{i} \theta_{i}^{n+1} \mathbf{1}_{L(i) = \ell} \right) + \theta_{0}^{n+1} = \max_{\ell} \sum_{i \in A(\ell) \cup \{-1\}} \theta_{i}^{n+1} + \theta_{0}^{n+1},$$

where $\theta_{-1}^{n+1}$ is defined to be 0, and where the crucial final step is due to the fact that the maximum over $x'$ is achieved by the $x'$ that places the substituent with the largest estimated value in each site (or no substituent in a site, corresponding to $i = -1$, if all the estimates at that site are negative). Substituting this expression for $\max_{x'} \mu_{x'}^{n+1}$ and noting that the tower property implies $\mathbb{E}_{n}[\theta_{i}^{n+1}] = \mathbb{E}_{n}[\mathbb{E}_{n+1}[\zeta]] = \mathbb{E}_{n}[\zeta] = \theta_{0}^{n}$, we obtain

$$\mathbb{E}_{n}[\max_{x'} \mu_{x'}^{n+1}] = \mathbb{E}_{n} \left[ \sum_{\ell} \max_{i \in A(\ell) \cup \{-1\}} \theta_{i}^{n+1} + \theta_{0}^{n+1} \right] = \mathbb{E}_{n} \left[ \sum_{\ell} \max_{i \in A(\ell) \cup \{-1\}} \theta_{i}^{n+1} \right] + \theta_{0}^{n}.$$
Then, noting that \( \max_{x'} \mu_{x'}^n = \sum_{\ell} \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^n + \theta_{0 \ell}^n \) by an argument similar to the one for \( \max_{x'} \mu_{x'}^{n+1} \), we have

\[
\nu_{x, KG}^n = \left( \sum_{\ell} \mathbb{E}_n \left[ \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^{n+1} | x^n = x \right] + \theta_{0 \ell}^n \right) - \left( \sum_{\ell} \mathbb{E}_n \left[ \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^n \right] \right)
\]

\[
= \sum_{\ell} \mathbb{E}_n \left[ \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^{n+1} | x^n = x \right] - \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^n
\]

Since the joint distribution of \( (\theta_{i \ell}^{n+1})_{i \in A(\ell)} \) conditioned on \( F_n \) and \( x^n = x \) is the same as that of the \( F_n \) conditional distribution of \( (\theta_{i \ell}^n + \tilde{\sigma}_{x \ell}^n Z)_{i \in A(\ell)} \), where \( Z \) is a one-dimensional standard normal random variable, and \( \theta_{i \ell}^{n+1} = \theta_{i \ell}^n + \tilde{\sigma}_{x \ell}^n Z \) almost surely, we substitute for \( \theta_{i \ell}^{n+1} \) and

\[
\nu_{x, KG}^n = \sum_{\ell} \mathbb{E}_n \left[ \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^n + \tilde{\sigma}_{x \ell}^n Z \right] - \max_{i \in A(\ell) \cup \{-1\}} \theta_{i \ell}^n
\]

\[
= \sum_{\ell} h \left( (\theta_{i \ell}^n)_{i \in A(\ell) \cup \{-1\}}, (\tilde{\sigma}_{x \ell}^n)_{i \in A(\ell) \cup \{-1\}} \right) . \quad \Box
\]

To actually use the decomposition property inherent in this proposition, we must first compute \( \tilde{\sigma}_{x \ell}^n \). We have \( (\tilde{\sigma}_{x \ell}^n)^2 = \text{Var}_n [\mathbb{E}_{n+1}[a_i | x^n = x]] \) which, using the conditional variance formula and the fact that \( \mathbb{E}_n[\text{Var}_{n+1}[a_i]] = \text{Var}_{n+1}[a_i] \), becomes

\[
(\tilde{\sigma}_{x \ell}^n)^2 = \text{Var}_n[a_i] - \text{Var}_{n+1}[a_i | x^n = x].
\]

We have \( \text{Var}_n[a_i] = C_{ii}^n \), and by (4),

\[
\text{Var}_{n+1}[a_i] = e_i^T C^{n+1} e_i = e_i^T \left[ C^n - \frac{1}{\gamma^n} C^n \tilde{x}^n (\tilde{x}^n)^T C^n \right] e_i = C_{ii}^n - \frac{e_i^T C^n \tilde{x}^n (\tilde{x}^n)^T C^n e_i}{\lambda_{x^n} + (\tilde{x}^n)^T C^n \tilde{x}^n}.
\]

Therefore,

\[
(\tilde{\sigma}_{x \ell}^n)^2 = \frac{e_i^T C^n \tilde{x}^n (\tilde{x}^n)^T C^n e_i}{\lambda_{x^n} + (\tilde{x}^n)^T C^n \tilde{x}^n}.
\]

which gives the vector \( \tilde{\sigma}_{x \ell}^n \), with \( i \) ranging over all the substituents (not just those corresponding to a particular site) as

\[
\tilde{\sigma}_{x \ell}^n = \frac{C^n \tilde{x}^n}{\sqrt{\lambda_{x^n} + (\tilde{x}^n)^T C^n \tilde{x}^n}}.
\]  \hspace{1cm} (7)

We summarize the resulting implementation below in Algorithm 5.1.

\begin{algorithm}
\textbf{Algorithm 5.1:} KGCB$_3$ Algorithm (Inputs: \( \theta^n, C^n, X \))

\begin{algorithmic}
\FOR {\( x \leftarrow 1 \) \TO \( M \)}
\STATE \( \tilde{\sigma}_{x \ell}^n = \frac{C^n \tilde{x}^n}{\sqrt{\lambda_{x^n} + (\tilde{x}^n)^T C^n \tilde{x}^n}} \)
\STATE \( \nu_x = 0 \)
\ENDFOR

\FOR {\( \ell \leftarrow 1 \) \TO \( L \)}
\STATE \( p \leftarrow (\theta_{i \ell}^n)_{i \in A(\ell) \cup \{-1\}} \)
\STATE \( q \leftarrow (\tilde{\sigma}_{x \ell}^n)_{i \in A(\ell) \cup \{-1\}} \)
\STATE \( \nu_x \leftarrow \nu_x + h(p, q) \% \) use Algorithm 3.2
\IF {\( x = 1 \) or \( \nu_x > \nu^* \)}
\STATE \( \nu^* \leftarrow \nu_x, x^* \leftarrow x \)
\ENDIF
\ENDFOR
\end{algorithmic}

\end{algorithm}
If there are \( l \) dimensions with \( M_l \) substituents that can be placed at each dimension, the number of total alternatives is \( M = (M_l)^l \). The computational complexity of Algorithm 4.1 is \( O(M^2 \ln M) \), which is equivalent to \( O(l(M_l)^{2l} \ln (M_l)) \). The computational complexity of Algorithm 5.1 on the other hand is \( O(M_l^l M_l \ln (M_l)) \) because the outer loop executes \( (M_l)^l \) times, while the inner loop executes \( l \) times and takes \( O((M_l) \ln (M_l)) \). Thus the efficient implementation has a computational complexity of \( O(l(M_l)^{l+1} \ln (M_l)) \), compared to \( O(l(M_l)^{2l} \ln (M_l)) \) for the previous implementation.

6 Empirical Study

We simulate the performance of the KGCB algorithm using data from a previously published QSAR study of narcotic analgetics, Katz et al. (1977), which uses the Free-Wilson model. The paper contains a set of 6,7-Benzomorphans, which have 5 sites at which substituents can be attached. The molecule is shown in Figure 2. At the site labeled \( R_1 \) there are 11 possible substituents (together with \( H \)), at site \( R_2 \) there are 8 possible substituents, at site \( R_3 \) there are 5 possible substituents, at site \( R_4 \) there are 6 possible substituents and at site \( R_5 \) there are 11 possible substituents. The collection of compounds resulting from choosing a substituent at each site contains \( 11 \times 8 \times 5 \times 6 \times 11 = 29040 \) compounds. Additionally, each compound can be charged positively, negatively or be neutral, which brings the number of compounds to 87120. Katz et al. (1977) provides experimentally observed activity values for 99 of these compounds.

![Figure 2: An example of substituent locations (marked with \( R_1, ..., R_5 \)). Source: Katz et al. (1977)](image)

In Section 6.1, we describe methods for choosing the prior distribution on the values of this collection of compounds. In Section 6.2, we present an empirical study of the performance of KGCB on this collection of compounds, comparing against two other policies. In Section 6.3, we discuss other policies against which we did not compare, but could be included in a more extensive empirical study.
6.1 Prior distribution on the substituent contributions

When choosing a prior distribution, one may be working with experienced practitioners who may be able to articulate their prior distributions on the values of compounds or substituents. In many cases, however, even if one is quite experienced in drug discovery, or is working with those who are quite experienced, it may be difficult to articulate prior beliefs. In such cases it is useful to have a method for setting the prior distribution from some other source, such as previous measurements of families of molecules that may be completely different from the one to which drug discovery effort is being applied. We now present one such method that may be used with either the Free-Wilson model or the general model.

We first discuss prior distributions for the Free-Wilson model. Our method of choosing a prior distribution supposes that there exists a large population of substituent values in nature, and that nature has drawn independently at random from this population the particular substituent values to create the family of compounds being investigated. We suppose that we may approximate the distribution of values within this population with a normal distribution whose mean and variance we can estimate. A method of estimation is discussed below.

Our method also supposes that we began with a non-informative belief on the value $\zeta$ of the base molecule, which was subsequently updated by a laboratory measurement. This measurement provides an informative belief on the value of the base molecule that we will use within our prior distribution. The mean of this belief is the value of this measurement, and its variance is the variance of the measurement noise. A measurement of the base molecule is generally available in practice, since a molecule is usually chosen to have its space of chemical derivatives searched because it performed well on an initial screening of many chemically dissimilar molecules.

Given these suppositions, our prior distribution is given by taking $\theta_0$ and $\sigma_0^2$ to be the value and measurement variance, respectively, of our previous measurement of the base molecule, and $\theta_i$ and $\sigma_i^2$ for $i > 0$ to be the estimated mean and variance, respectively, of the population of substituents. Our independence supposition causes the off-diagonal terms of $\sigma^2$ to be 0. These values for $\theta_0$ and $\sigma_0^2$ specify a prior distribution as discussed in Section 4.1, and the equivalent prior distribution on compound values may be reconstructed as discussed in that section. This provides a prior distribution appropriate for the Free-Wilson model.

We now discuss how this method of setting a prior distribution can be extended to the general model. We suppose, in a manner quite similar to our supposition about substituent values, that there exists a large population of deviation terms from which nature has drawn independently at random to create the family of compounds being considered. Let $\sigma_b^2$ be the estimated variance of this population, and, without loss of generality, we may assume that the mean of the population of deviation terms is 0 because we can take any non-zero mean and move it into the value of the base molecule. Then the prior distribution, in the format of Section 4.3, is given by taking
the same $\theta^0$ and $C^0$ as just described for the Free-Wilson model, and using this value of $\sigma^2_b$ in subsequent updates.

In order to use either of these methods for setting the prior distribution, one needs access to a measurement of the base molecule (with that measurement’s variance), estimates of the mean and variance of the population of substituents, and, when using the general model, an estimate of the variance of the population of deviation terms. These estimates may be obtained from measurements of a reasonably large family of compounds that may be quite dissimilar from the one being investigated. Using these measurements, one may use linear regression to estimate the substituent values and deviation terms present in this family of compounds. The substituent values and deviation terms that result will certainly be a very small subset of the entire population of substituent values and deviation terms present in nature, but if one is willing to suppose that they are representative of what one may encounter in nature, then the population means and variances of the observed values may be taken as estimators of the means and variances of the overall populations. Because one might make this assumption that the observed values are “representative,” it is better if one has observations from multiple families of molecules, and it is also better if the observed values are from families that are more similar to the one to which drug discovery effort is being applied.

We followed exactly this method of estimation using the data in Katz et al. (1977). This paper provides measured values for 99 of the possible 87120 compounds. The measurement technique used is quite accurate, and it is reasonable to take the measured values as the “true” values. By fitting a multiple linear regression to this data set, we obtain substituent values and deviation terms. From these values, we estimate the mean and variance of the population of substituents as $0.31$ and $0.47$ respectively, and the variance of the population of deviation terms as $\sigma^2_b = 0.15$.

### 6.2 Simulation Results

Before discussing the ability of KGCB to discover good molecules with relatively few measurements in simulation, we first describe the computational costs and memory requirements of the various implementations of the Free-Wilson-based KGCB implementation on the set of 87120 compounds. Algorithm 5.1 is able to compute the KGCB decision and update the belief in less than 3 seconds, which is approximately 100 times faster than Algorithm 4.1. We were unable to run the standard implementation on this data set of 87120 compounds because it stores and manipulates a covariance matrix on the compounds with dimension 87120x87120, which requires 14GB of memory (if only the upper triangular part of the symmetric matrix is stored, and 4 byte single-precision floating point format is used). This memory requirement could not be met by the hardware available to us. In contrast, Algorithms 5.1 and 4.1 manipulate a covariance matrix on the attributes with dimension 43x43, requiring only 3.6KB of storage (15KB if the full matrix is stored in double-precision).
In our simulations, we observed the number of measurements required by the KGCB algorithm to find good compounds among collections of compounds of various sizes. In these simulations, we compared KGCB against two policies: a pure exploration policy that chooses compounds to test uniformly at random, and a policy from the frequentist design of experiments (DOE) literature based on a linear model, called one-factor-at-a-time (Montgomery (2005)) (OFAAT). This policy begins by testing the base molecule, and then sequentially tests each compound created by adding one substituent to the base molecule.

The OFAAT policy neglects interaction between factors (Montgomery (2005)). In most applications this is detrimental to performance, but under the perfectly linear Free-Wilson model there are no interactions between terms, and so neglecting interactions provides an advantage to OFAAT. In the general model, where factors interact via the deviation terms \( b_x \), a factorial design would perhaps be more appropriate (Montgomery (2005)), but such designs are infeasible in problems with many dimensions like the drug discovery problem considered here.

In our simulations, we maintain beliefs for each competing policy (exploration and OFAAT) using the same updates as the KGCB algorithm, and use these to choose implementation decisions. This allows the other policies to take advantage of the linear structure of the problem in their implementation decisions, even if they do not in their measurement decisions. We emphasize that these belief updates use correlated beliefs, which give our pure exploration policy a substantial advantage over any policy that uses independent belief updates. With our version of pure exploration, even though it measures uniformly at random, it learns about the compounds through correlations in the same way that KGCB does.

Although we compare only against pure exploration and the DOE-based OFAAT policy, other policies from the literature that more completely exploit the structure of the problem might have performed better. While the measurement decisions of OFAAT were chosen with knowledge of the linear structure of the problem, they do not exploit the objective function (which rewards information about the best alternative rather than information about all alternatives), and so other DOE-based policies might perform better. Pure exploration takes advantage of the linear structure of the problem only when making its implementation decisions, and not when making its measurement decisions. Thus it should be viewed only as a benchmark representing non-adaptive learning. In Section 6.3, we discuss other policies from the literature that might be applied to this problem, and which could be compared against in a more extensive empirical study.

Section 6.2.1 presents numerical results from the Free-Wilson model described in Section 2.1, and Section 6.2.2 presents numerical results from the general model described in Section 2.2.

**6.2.1 Results using the Free-Wilson model**

We begin by describing results for the simple, linear-additive model described in Section 2.1. We consider the opportunity cost, defined as the difference between the true value of the compound
that is actually best, and the true value of the compound which is best according to the policy’s posterior belief distribution. Thus, once the opportunity cost is 0, the policy has found the best compound. Although we consider the opportunity cost, which is a linear function of biological activity, the value of a compound to a practitioner is often a non-linear function of activity, e.g., when a compound may only be used as a drug when its activity exceeds a threshold.

Figure 3 plots the average opportunity cost as a function of the number of measurements using two data sets (one with 2640 compounds and the other with 87120 compounds) and two values for the noise standard deviation (0.1 and 0.5). Using the measured values published in Katz et al. (1977), we generate a truth for these compounds by fitting a linear regression to the data, taking the true means $\theta$ to be the predicted values from the regression and the standard deviation of the noise in our measurements to be the standard deviation of the residuals from the regression. For the 2640 compounds data set, we average over 100 runs, and for the 87120 compounds data set we average over 10 runs. Averaging over more runs removes more of the variance introduced by Monte Carlo sampling of the truths, causing the curves for 2640 compounds to be smoother than those for 87120 compounds. As Figure 3 shows, KGCB outperforms the other policies, with an average opportunity cost that is always lower.

OFAAT performs worse than both KGCB and pure exploration. Although OFAAT is a first-order orthogonal design and hence optimal with respect to a minimal variance criterion (Myers & Montgomery (2002) section 7.3.1), we consider a different criterion: the ability to find the best compound. Also, OFAAT measures compounds containing single substituents, while KGCB and pure exploration measure compounds containing several. Although this provides OFAAT more information per measurement about each substituent measured, the way that KGCB and pure exploration learn a little bit about more substituents seems to be better in this application.

KGCB also performs better than pure exploration. Although pure exploration is able to use the full Bayesian machinery to guide its implementation decisions, thus taking advantage of the linear structure of the model, its measurement decisions are chosen purely at random. In contrast, KGCB is able to take information gleaned in the early measurements to guide its later measurements to more informative compounds.

To further compare the performance of KGCB and pure exploration, we randomly selected sets of compounds from the full set of 87120, and performed more extensive analysis on these smaller problems. Figure 4(a) shows the expected opportunity cost of KGCB and pure exploration on a single set of compounds. To generate this figure, we randomly selected a single set of 10000 compounds, ran 15 independent sample paths each for KGCB and pure exploration on these compounds, and estimated the expected opportunity cost as a function of the number of measurements. An estimate and its standard error is plotted for each policy, where each error bar is twice the standard error in each direction. The figure shows that the estimated expected opportunity cost for the KGCB policy is lower than that of the pure exploration policy.
(a) Average opportunity cost over 100 runs with 2640 compounds and noise standard deviation 0.1.  
(b) Average opportunity cost over 100 runs with 2640 compounds and noise standard deviation 0.5.  
(c) Average opportunity cost over 10 runs with 87120 compounds and noise standard deviation 0.1.  
(d) Average opportunity cost over 10 runs with 87120 compounds and noise standard deviation 0.5.  

Figure 3: Average opportunity cost under the Free-Wilson model.

Figure 4(b) shows the distribution of the difference in opportunity cost over multiple sets of compounds. To generate this figure, we randomly selected 75 sets containing 10000 compounds each. For each set of compounds, we ran KGCB and pure exploration independently and observed the difference in opportunity cost as a function of the number of measurements $n$. When this difference is positive, pure exploration has a higher opportunity cost, and KGCB performs better. We then plot, for each $n$, the mean and standard deviation of this population of 75 differences. This characterizes the overall distribution of the difference in performance between KGCB and pure exploration across multiple sets of compounds. Figure 4(b) shows that the mean difference is always positive after the first measurement, and it has a maximum near the tenth measurement. This suggests that the learning rate is faster for KGCB than for pure exploration, and the value of using the KGCB policy is maximized when the measurement budget is small.

For more numerical work using the Free-Wilson model, please see the appendix.
6.2.2 Results using the general structural model

We now present the results using the general model described in Section 2.2. To simulate a true set of compound values in this set of experiments, we take the fitted values of $a_i$ and $\zeta$ from Katz et al. (1977), and for each compound $x$, we generate $b_x$ independently from a normal distribution with mean 0 and variance $\sigma^2_{b}$ (where $\sigma^2_{b}$ was obtained from the fit as well, as described in Section 6.1). We then combine these values according to the model to obtain true values for the compounds.

We compared KGCB only to the pure exploration policy, since OFAAT is not designed to handle the deviation terms present in the general model, and it did not perform as well as pure exploration in the deviation-free Free-Wilson case. To test the performance of KGCB under this model, we randomly selected 10 different sets of 1000 compounds each and ran the KGCB algorithm using the prior distribution described in Section 6.1. For each sample path, we perform 150 measurements comparing KGCB to pure exploration and we plot the average opportunity cost over the 10 runs just as in our previous experiments. The results are shown in Figure 5.

As with the Free-Wilson model, the KGCB policy performs significantly better than pure exploration. Opportunity cost decreases more slowly in this test than it did under tests that assumed the Free-Wilson model. This is because this test adds deviation terms to each molecule, which makes it harder for any policy to find the best compound. Although this test was performed with 1000 compounds and the general model contains a different deviation term for each one, the KGCB policy finds a very good compound with many fewer than 1000 measurements. This is because $\sigma^2_{b}$ is smaller than $\text{Var}_0(a_l)$, which makes learning the $a_l$ terms more important than learning the $b_x$ deviation terms, and allows a policy to find a compound with small opportunity cost without learning all the deviation terms.
6.3 Other Related Measurement Policies

We have compared the KGCB policy against two other policies: pure exploration and OFAAT. In this section we discuss other policies that a more extensive empirical study might include and discuss their relationship to KGCB. We note that KGCB has been compared against many of these other policies in other problem settings, and has outperformed them there.

First, the Bayesian global optimization (BGO) literature contains several relevant policies. First, there are the expected improvement (EI, Mockus (1972)) and maximum probability of improvement (MPI, Kushner (1964)) policies, both of which were discussed in Section 1. In addition, there is the interval estimation (IE) policy Kaelbling (1993), which combines exploration with exploitation by measuring the compound for which a particular linear combination of the posterior mean and posterior standard deviation is largest. IE was applied to BGO in Cox & John (1997), where it was called SDO, or sequential design for optimization.

As described in Section 1, De Grave et al. (2008) also considers sequential designs for drug discovery, and actually applies all three of these policies to a drug discovery problem (IE is referred to as the optimistic strategy). However, that empirical study assumes that measurements are free from noise, and the versions of these BGO policies they employ require modification for the noisy case before they can be applied to the problem we consider here. In addition to these three BGO policies, De Grave et al. (2008) also considers three other policies: pure exploration; largest value, which measures the untested compound with the largest estimated value; and largest variance, which measures the compound whose value has the largest posterior variance.

Frazier et al. (2008) compares KG to IE, largest variance, largest value, and pure exploration on the ranking and selection problem with an independent normal prior distribution. In this problem, since the prior distribution is independent and measurements of one alternative do not affect the other alternatives, KG can also be understood as a generalization of EI to the noisy case. (KGCB does not reduce to EI when beliefs are correlated, even if measurements are noise-free.)
Frazier et al. (2009) compares KG to EI in a discretized version of the BGO problem, as well as a version of EI generalized to handle noise called sequential kriging optimization (Huang et al. 2006). In both of these studies, KG performed as well or better than these other methods, finding better alternatives with fewer measurements, but requiring more computation for each decision.

There are also policies from the DOE literature that might be applied to this problem. In the numerical study of the Free-Wilson model, we compared against OFAAT. One could compare against other first-order orthogonal designs, or to a response surface method (Myers & Montgomery 2002) using such designs. One could also compare against screening methods, like the sequential bifurcation method described in Kleijnjen (2008). With such screening methods, it may be possible to determine which of the substituents have the largest values with fewer measurements than there are substituents. The sequential bifurcation method cannot be directly applied, since it requires knowing the sign of the effect of each factor, but it may be possible to adapt this method, or other similar methods, to the problem of drug discovery.

7 Conclusions

Drug discovery is a long and expensive process for which sophisticated sequential procedures like KGCB are well-suited. They have the potential to reduce the costs involved and to increase the number of drug discovery efforts that find success within budget. Our new algorithms for calculating the KGCB policy allow efficient implementation on large datasets, and our empirical study demonstrates that KGCB performs much better than the pure exploration and one-factor-at-a-time policies. Extensions and improvements to this work that we feel would be of value include more extensive empirical comparisons between KGCB and other policies, as well as implementations of the KGCB algorithm based on other QSAR models.

Acknowledgments

We would like to thank Jeffrey Toretsky, Sivanesan Dakshanamurthy, and the many others at the Lombardi Comprehensive Cancer Center at Georgetown University, who provided their substantial expertise and taught us about the broader context that motivated this paper. We would also like to thank the Go4TheGoal foundation for introducing us to the group at Georgetown University, and for encouraging us to pursue this work. We would also like to thank Elyse Powell for first pointing out the application of optimal learning to the problem of drug discovery. Finally, we would like to thank the editors and anonymous referees whose suggestions improved this work.

References


Hsu, J. (1996), Multiple Comparisons: theory and methods, CRC Press, Boca Raton, FL.


Kleinberg, R. (2005), Online decision problems with large strategy sets, PhD thesis, MIT.


Powell, W. B. (2007), Approximate Dynamic Programming: Solving the curses of dimensionality, John Wiley and Sons, Hoboken, NJ.


8 Appendix: Additional Numerical Results

In this section we present further numerical experiments using the Free-Wilson model, which complement and extend the results presented in Section 6.2.1.

Since the experiments in Section 6.2.1 showed that pure exploration is more competitive than OFAAT, the experiments we present here compare KGCB only to pure exploration. In these further numerical experiments, we randomly selected sets of compounds from the full set of 87120.

Figure 6 shows a sample path of 200 measurements on a set of randomly selected 1000 compounds, giving the opportunity cost at every iteration on the top panel and the true value of the compound that is measured at each step on the bottom panel. While pure exploration fails to find the best compound in the first 200 measurements, KGCB manages to do so in about 15 measurements (on this sample path), which is a significant improvement if we think about the amount of time and money each measurement requires. As the bottom panel shows, part of KGCB’s success in this sample path was the fact that it measured two very good compounds early on, at steps 10 and 15, while pure exploration was measuring mediocre compounds for a long period of time, which prevented it from discovering the best compound quickly enough.

We also present the results from running the code on data sets of randomly selected 10000

![Figure 6: A sample path of 200 measurements on a data set of 1000 compounds showing opportunity cost (top) and the true values of the compounds tested at each time step (bottom).](image)
compounds. Figure 7 shows four such sample paths of the new implementation. As the figure shows, the performance of the KGCB algorithm is encouraging, as it is usually able to find the best compound by the 75th measurement, and gets very close to finding it by the 25th measurement. In most sample paths, KGCB does at least as well as pure exploration - the policy in which we choose compounds to measure completely at random, but update the belief in the same way as we do for the KGCB policy.

We are interested also in the typical performance of the KGCB policy. That is, if we are able to perform only one sample path (which is generally the case in practice), we are interested in knowing the probability that the KGCB policy performs better than an exploration policy.

To better get a sense of the distribution of the relative performance of KGCB versus pure exploration, we have made a box and whiskers plot of the differences between opportunity costs at every 10 measurements. As shown in Figure 8, except for the first measurement, all the other measurements have the lower quartile cutoff at or above 0, which suggests that the probability that KGCB is better than pure exploration, on any measurement of any sample path, is higher than 1/2.

Having tested the KGCB policy on data sets of 10000 compounds, we increased the size of the data sets further to 25000 compounds. Figure 9 shows four sample paths from running the KGCB and pure exploration policies on four randomly selected data sets of 25000 compounds. The rate of convergence for these plots is slower than for 10000 compounds, but the KGCB policy still manages to get reasonably close to the best compounds after about 50 measurements.

Figure 10 shows the mean opportunity cost for nine sample paths using nine different randomly chosen data sets of 25000 compounds. Although not as impressive as the mean plot for 10000 compounds, this plot also attests that there is value in using the KGCB policy as opposed to a pure exploration policy.
Figure 7: Four sample paths using data sets of 10000 compounds and a noise standard deviation of 0.38.

Figure 8: Distribution of difference between opportunity costs between pure exploration and KGCB using 75 sample paths of 10000 compounds each and a noise standard deviation of 0.38.
Figure 9: Four sample paths using data sets of 25000 compounds and a noise standard deviation of 0.38.

Figure 10: Average over nine runs of sample paths using data sets of 25000 compounds and a noise standard deviation of 0.38.