Bayesian Optimization Techniques for High-dimensional Transportation Problems

Working Paper

Timothy Tay*[1] and Carolina Osorio†[1]

[1]Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge MA 02139, USA

Abstract

Simulation-based urban traffic and mobility models are increasingly being used to evaluate changes to network designs and operations of large urban networks. However, these simulation models are computationally costly to evaluate. Hence, when used within simulation-based optimization frameworks, it is important to use an efficient optimization algorithm that can identify good solutions within a limited number of iterations. This paper proposes a Bayesian optimization technique suitable for high-dimensional problems. The technique combines ideas from Bayesian optimization, Gaussian processes and analytical transportation modeling. The proposed method incorporates problem-specific prior information in the covariance function of the Gaussian process. The problem-specific prior information comes in the form of an analytical transportation model. This helps to promote exploration of the feasible region with the aim of finding better solutions. This can be on top of using the analytical transportation model in the prior mean function of the Gaussian process. We illustrate how the proposed method works on a 1-D example, and test it by running a Bayesian optimization of the 100-D Griewank function. We then demonstrate the use of the proposed Bayesian optimization approach on a large-scale fixed time traffic signal control problem for Midtown Manhattan. The results show that problem-specific information in the form of analytical transportation models can be used for both exploration and exploitation to tackle high-dimensional transportation problems efficiently using Bayesian optimization.

1 Introduction

Simulation-based urban traffic and mobility models are tools commonly used by transportation agencies and operators (e.g. ridesharing operators) to evaluate changes to their network designs or operations (Stone 2021, TSS-Transport Simulation Systems 2019, 2009). For instance, transportation agencies may use simulators to evaluate traffic management strategies, such as congestion pricing and traffic signal control. Ridesharing operators may use simulators to evaluate new algorithms before releasing it to production (Greenhall 2016).

These simulators can embed detailed models of traveler behavior, such as mode choice, route choice and response to real-time traffic conditions. This allows the interactions between travelers to be modeled. At the same time, the simulators can keep track of many quantities of interest for every traveler in the network in intricate detail (Pell et al. 2017), such as travel time, fuel consumption, number of stops, etc. Furthermore, the resolution of the simulation models, as well as the ability to simulate large-scale networks, are constantly improving. This makes it all the more enticing for transportation agencies and operators to make use of simulators in their planning and operations. For an in-depth review of existing traffic simulation models, we refer the reader to Barceló et al. (2010).

*tayt@mit.edu
†osorioc.edu

However, the higher simulation model resolution and greater spatial coverage of networks being simulated also lead to increasing computational demand of simulators. When used within simulation-based optimization (SO) frameworks, the computational cost of evaluating the simulator becomes more apparent, since a simulation run is required each time the objective function is evaluated. With a stochastic simulator, multiple simulation runs may even be required to obtain an accurate estimate of the objective function value.

At the same time, transportation agencies and operators are interested in optimizing for entire urban networks. For some problems, such as urban traffic signal optimization (Osorio and Chong 2015) and origin-destination (OD) matrix calibration (Zhang et al. 2017, Lu et al. 2015), this might lead to optimization problems of increasing dimensions. The increasing problem dimensionality means that more simulation runs (i.e. objective function evaluations) might be required. Hence, this points to the need for more efficient optimization algorithms.

In this work, we consider high-dimensional simulation-based optimization problems that have continuous and general (i.e. non-convex) objective functions, with unknown analytical forms. The constraints are assumed to be analytical and differentiable. Such a problem can generally be formulated as:

$$\min_{\mathbf{x}\in\chi} f(\mathbf{x}, \mathbf{z}; \mathbf{p}) \equiv \mathbb{E}[F(\mathbf{x}, \mathbf{z}; \mathbf{p})], \tag{1}$$

where f is the objective function, F represents the stochastic output of a simulation run, $\mathbf{x}$ is the high-dimensional vector of decision variables, χ is the feasible region, $\mathbf{z}$ denotes the vector of endogenous simulation variables and $\mathbf{p}$ represents the vector of deterministic exogenous parameters. The exogenous parameters and endogenous variables are specific to the problem in question. Examples for a traffic signal control problem are provided in Section 5.2.

Given the computation cost of evaluating the simulator every time we want to obtain an estimate of the objective function, we also assume a limited computational budget (i.e. the total number of simulation runs is limited). Since we are working with high-dimensional problems, the limited computational budget means that we are working in the regime where we have more decision variables than observations. When working in this regime, it is crucial for the optimization algorithm to balance exploration and exploitation so as to find a good solution within the limited computational budget. In the context of optimization, exploration refers to the search in regions with few evaluated points, while exploitation refers to searching in regions with good estimated performance (Sun et al. 2014).

In the past, common approaches used to tackle transportation SO problems often consisted of general-purpose algorithms, including genetic algorithms (Jin et al. 2017, Sebastiani et al. 2016, Paz et al. 2015, Stevanovic et al. 2008, Teklu et al. 2007, Yun et al. 2006) and simultaneous perturbation stochastic approximation (SPSA) (Tympakianaki et al. 2018, Lu et al. 2015, Tympakianaki et al. 2015). While these general-purpose algorithms can easily be applied to different problem types, they are not designed to be used under a tight computational budget. Instead, they tend to be designed based on asymptotic properties. When used in high-dimensional SO, the large number of objective function evaluations (i.e. simulation runs) required means that it is rather computationally inefficient. This is especially true when working with a computationally demanding simulator.

A different approach to SO involves combining information from the simulator with problem-specific prior information in the form of analytical transportation models with the aim of identifying good solutions efficiently. Osorio and Bierlaire (2013) proposed a metamodel SO framework, which embedded an analytical queueing network model approximation of the objective function (Osorio and Bierlaire 2009) in the metamodel. This metamodel SO approach has been used to tackle various types of high-dimensional problems in the transportation field, including urban traffic signal optimization (Chong and Osorio 2018, Osorio and Chong 2015, Osorio and Nanduri 2015a,b), OD matrix calibration (Zhang et al. 2017), congestion pricing (Osorio and Atastoy 2017) and car-sharing network design (Zhou et al. 2018). However, the metamodel SO approach does not explicitly try to balance exploration and exploitation. For instance, a general-purpose sampling strategy (e.g. uniform random sampling) is often used for exploration. There is potential to improve the performance of SO algorithms by exploiting the structural information of analytical transportation models to design suitable exploration-exploitation techniques.

To address the issue of balancing exploration and exploitation in SO, we propose a method to incorporate problem-specific prior information in the Bayesian optimization (BO) framework using Gaussian processes (GP) (a brief explanation of GPs is provided in Section 3.1). On top of using the analytical transportation model in the prior mean function of the GP, we propose a new analytical transportation model-based covariance function. Using the analytical transportation model in the prior mean function of the GP allows the algorithm to exploit the problem-specific prior information to quickly identify good solutions. On the other hand, the proposed covariance function is designed to encourage the algorithm to explore in regions with different analytical transportation model value from points which have already been evaluated. By incorporating an analytical transportation model in the prior mean function and the covariance function of the GP, the proposed method achieves the following

aims:

- **Efficiency:** The proposed covariance function allows the Bayesian optimization algorithm to efficiently tackle high-dimensional optimization problems with a limited number of simulation runs, by exploiting the correlations between the objective function and analytical transportation model to encourage targeted exploration of the feasible region.
- **Balances exploration and exploitation:** The Bayesian optimization approach used in the proposed method provides a way to balance the exploration-exploitation trade-off compared to the existing metamodel SO approach.
- **Generality:** The method can be easily generalized to other classes of optimization problems with expensive-to-evaluate objective functions, as long as a suitable analytical transportation model for approximating the objective function is available.

In the following section, we provide a review of BO, along with previous efforts attempting to use BO to solve high-dimensional optimization problems. In Section 3, we briefly present how GPs and BO work, followed by the proposed method used to combine problem-specific prior information in GPs, for use in BO. We then illustrate, in Section 4, how the method works in the case of the 1-dimensional Griewank function, followed by a validation of the method using the 100-dimensional Griewank function. The proposed method is tested in a case study using a model of Midtown Manhattan for a traffic signal optimization problem. The results of the case study are presented and discussed in Section 5. Lastly, the conclusions are provided in Section 6.

2 Review of High-Dimensional Bayesian Optimization

Bayesian optimization was first popularized by the efficient global optimization algorithm proposed by Jones et al. (1998). In recent years, BO has become widely-used for tackling global optimization problems where the objective function is expensive to evaluate, and hence the number of objective function evaluations allowed is small. It provides a very efficient approach for the optimization of expensive-to-evaluate functions (Sasena et al. 2002, Jones et al. 1998), with the efficiency stemming "from the ability of Bayesian optimization to incorporate prior belief about the problem to help direct the sampling, and to trade off exploration and exploitation of the search space" (Brochu et al. 2010). In the field of transportation, BO has been employed to tackle problems such as OD matrix calibration (Schultz and Sokolov 2018) and toll optimization (Chen et al. 2014).

Most often, GP models are used as part of BO to approximate the unknown objective function. GPs provide an attractive way "to construct a Bayesian non-parametric regression model" (Shahriari et al. 2015), as analytical expressions are available for the posterior GP. This includes an analytical estimate of the variance of the posterior GP predictions, which is used in BO to balance the exploration-exploitation trade off. Other models have been used to approximate the objective function, such as the tree Parzen estimator (Bergstra et al. 2013, 2011) and random forests (Shahriari et al. 2015, Hutter 2009), but these models tend to be less suitable when working with a limited number of evaluated points. This is due to the data set (i.e. evaluated points) being split at every decision node of the trees. With a limited number of evaluated points, the trees have to be shallow to prevent overfitting. Furthermore, Mockus (1994) showed that the GP prior distribution satisfies the conditions necessary for BO to converge to the optimum. Hence, this makes GP models fitting for use in BO. A brief introduction to GPs is provided in Section 3.1.1. However, for a comprehensive guide on GPs, we refer the reader to Williams and Rasmussen (2006).

Historically, GPs are more frequently referred to as Kriging in geostatistics. This name has carried over to the metamodel-based optimization literature, where the Kriging metamodel is commonly used to approximate the objective function (see e.g. Kleijnen (2017)). While known by different names, the Kriging metamodels and GPs are mathematically equivalent. For instance, ordinary Kriging is equivalent to a GP with a constant prior mean and deterministic observations (Kleijnen 2017), while stochastic Kriging (Ankenman et al. 2010) is equivalent to a GP with a noise term and is used to model a stochastic function. The main difference between BO and Kriging metamodel-based optimization is that the latter directly optimizes the metamodel at each iteration (see e.g. Osorio and Bierlaire (2013)), while an acquisition function is derived from the GP and optimized in BO instead. The acquisition function helps to guide the search for the optimum, and is designed to systematically balance the exploration-exploitation trade-off. A more detailed explanation of the acquisition function is provided in Section 3.1.2.

Despite the successes of BO, it is widely acknowledged that BO is mostly limited to low-dimensional problems, typically less than 10 dimensions (Wang et al. 2016, Kandasamy et al. 2015). Scaling BO for use in higher-dimensional problems has been a major challenge in the field that remains unsolved. There are two key reasons for the poor performance at higher dimensions. First, there is the challenge of modeling the objective function at higher dimensions – the number of observations needed to get a good coverage of the feasible region increases exponentially with dimensions (Moriconi et al. 2020, Wang et al. 2016, Kandasamy et al. 2015). In the context of SO, this presents a significant practical challenge due to the high computational demand of obtaining observations through simulation. Second, there is the challenge of globally optimizing the acquisition function at every iteration (Rana et al. 2017, Kandasamy et al. 2015). The acquisition function often has many flat regions and can be highly multimodal, particularly in higher dimensions, making it tricky to find the optimum.

There are some other minor limitations of BO, which we do not directly address in this paper. In particular, these limitations are associated with the use of GP models for approximating the objective function. First, GPs are unable to model conditional variables, where a variable only becomes active when certain conditions are met for other decision variables. In such cases, tree-based models have been shown to be more suitable (Bergstra et al. 2013, 2011). Second, the traditional GP set-up also assumes stationary kernels (i.e. the covariance function has a fixed lengthscale). However, some objective functions may be non-stationary (Snoek et al. 2014). As a result, the traditional GP would poorly approximate the objective function, thus resulting in poor optimization performance.

There have been many attempts to tackle high-dimensional BO. Most of the existing work assumes that the objective function mostly depends on a lower-dimensional "active" subspace (Munteanu et al. 2019, Schultz and Sokolov 2018, Wang et al. 2016, Chen et al. 2012). For instance, Wang et al. (2016) tackled a problem with a billion dimensions, by projecting the higher-dimensional space to a lower-dimensional subspace through random embedding. Li et al. (2017) also proposed a dropout strategy to optimize only a subset of variables at each iteration. However, these methods assume that the objective function has a lower-dimensional "active" subspace, which may not hold true in general. Moriconi et al. (2020) showed that projecting the data onto a lower-dimensional subspace can lead to underfitting of the GP. Hence, they proposed the quantile GP for selecting on the best observation for each parameter sub-configuration when fitting the GP. However, the quantile GP method still assumes that the objective function is effectively low-dimensional. The elastic GP (Rana et al. 2017) tries to overcome the problem of flatness in the acquisition function by making the covariance function lengthscale large enough, so as to get some significant (i.e. nonzero) gradient values to aid the acquisition function optimization. The Add-GP-UCB model (Li et al. 2016) treats the acquisition function as an additive function of mutually exclusive lower-dimensional components, assuming that the acquisition function can be decomposed into an additive form. This allows the optimization of the acquisition function to be done in a lower-dimensional space.

A relevant branch of BO involves the use of multi-output GPs (Liu et al. 2018, Poloczek et al. 2017, Kandasamy et al. 2016, Swersky et al. 2013). It attempts to use correlation between the objective function and low-fidelity models, in order to inform the search for the optimum. The main aim of using multi-output GPs is to reduce the number of objective function evaluations, and hence computing time, needed to find the optimum. While multi-output GPs could potentially be used to provide better coverage of the feasible region through correlated observations of the low-fidelity models (see e.g. Swersky et al. (2013)), the large number of observations required for a high-dimensional problem could lead to another computational bottleneck – computing the posterior GP model involves a matrix inversion and typically scales as $O(t^3)$ (Williams and Rasmussen 2006, Chapter 8), where t is the number of observations.

In order to tackle high-dimensional SO problems with BO, we propose to take advantage of problem-specific prior information. This allows us to avoid the assumption that the objective function has a lower-dimensional "active" subspace. At the same time, our proposed method uses problem-specific prior information in a way that helps to overcome the lack of observations when modeling the objective function using a GP, while remaining computationally efficient. In the following section, we provide a brief explanation of how GPs and BO work, followed by a presentation of our proposed method.

3 Method

We first provide a brief explanation of how GPs and BO work in Section 3.1. Readers who are familiar with BO and GPs may wish to skip to Section 3.2. In Section 3.2, we propose methods to incorporate problem-specific prior information into GPs through the prior mean function and the covariance function.

3.1 Bayesian Optimization

Bayesian optimization consists of two main components – a model of the objective function that can be updated with the observations at every iteration, as well as the acquisition function which decides on the next point to evaluate. In Section 3.1.1, we provide a brief introduction to one of the most popular objective function models for use in BO – Gaussian processes. This is followed by an explanation of the role of the acquisition function in BO in Section 3.1.2, and a summary of the BO algorithm in Section 3.1.3.

3.1.1 Gaussian Process.

A GP represents a distribution over functions, and is specified by a prior mean function $m(\cdot)$ and a covariance function $k(\cdot,\cdot)$. Specifically, given a set of t points $\mathbf{x}_{1:t} = [\mathbf{x}_1, \ldots, \mathbf{x}_t]^T$, where $\mathbf{x}_i \in \mathbb{R}^D, i = 1, \ldots, n$, with objective function estimates $\mathbf{f}_{1:t}$, where $f_i = f(\mathbf{x}_i)$, the GP prior can be defined by the multivariate normal distribution:

$$\mathbf{f}_{1:t} \sim \mathcal{N}\left(\mathbf{m}_{1:t}, K\left(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}\right)\right), \tag{2}$$

where $\mathbf{m}_{1:t}$ is the vector of mean function values such that $m_i = m(\mathbf{x}_i)$, and $K(\mathbf{x}_{1:t}, \mathbf{x}_{1:t})$ is the covariance matrix, with the $(i,j)^{th}$ element being $k(\mathbf{x}_i, \mathbf{x}_j)$ (i.e. the covariance between $\mathbf{x}_i$ and $\mathbf{x}_j$).

The choice of prior mean function and covariance function is up to the user. Simply speaking, the mean function in GPs defines the mean value of the normal distributed objective function estimate at a given point in the feasible region, while the covariance function defines the covariance between two points which represents how correlated two points are believed to be. Ideally, the prior mean function and the covariance function should be chosen such that they best represent the objective function. For instance, if the objective function contains a periodic component, the covariance function should ideally contain a periodic component too (e.g. see Chapter 5.4.3 of Williams and Rasmussen (2006)). In the case where the form of the objective function is completely unknown, a zero function or constant is often used as the prior mean function, while a popular choice of covariance function is the squared exponential function:

$$k^{SE}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_0^2 \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\ell^2}\right), \tag{3}$$

where the hyperparameters σ_0^2 and ℓ are the amplitude and characteristic length-scale of the covariance respectively. The hyperparameter ℓ determines the Euclidean distance between two points required for the two points to effectively be uncorrelated. In Section 3.2, we show how problem-specific information, when available, can be incorporated in the prior mean function and the covariance function to aid the optimization of high-dimensional problems. For more information and examples of covariance functions, we refer the reader to Chapter 4 of Williams and Rasmussen (2006).

One of the factors contributing to the popularity of GPs is its analytical tractability. As seen from Eq. (2), the points $\mathbf{x}_{1:t}$ are jointly Gaussian in the GP prior. Similarly, a new point $\mathbf{x}_*$ would also be jointly Gaussian under the GP prior:

$$\begin{bmatrix} \mathbf{f}_{1:t} \\ f(\mathbf{x}_*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}_{1:t} \\ m(\mathbf{x}_*) \end{bmatrix}, \begin{bmatrix} K\left(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}\right) & \mathbf{k}(\mathbf{x}_{1:t}, \mathbf{x}_*) \\ \mathbf{k}(\mathbf{x}_*, \mathbf{x}_{1:t}) & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix}\right), \tag{4}$$

$$\text{where } \mathbf{k}(\mathbf{x}_*, \mathbf{x}_{1:t}) = [k(\mathbf{x}_*, \mathbf{x}_1), \ldots, k(\mathbf{x}_*, \mathbf{x}_t)], \tag{5}$$

$$\mathbf{k}(\mathbf{x}_{1:t}, \mathbf{x}_*) = [\mathbf{k}(\mathbf{x}_*, \mathbf{x}_{1:t})]^T. \tag{6}$$

If the points $\mathbf{x}_{1:t}$ and estimates $\mathbf{f}_{1:t}$ were treated as previous observations (i.e. training points), it is possible to condition on them to limit the distribution of possible functions as predicted by the GP (i.e. fitting the GP prior to the observations). This is known as the posterior or predictive distribution, and can be obtained analytically using the Sherman-Morrison-Woodbury formula (see e.g. Appendices A.2 and A.3 of Williams and Rasmussen (2006) for details on the derivation). In the case of noisy observations (assuming additive i.i.d. Gaussian noise with variance τ^2), the resulting posterior distribution given the t observations takes on a Gaussian distribution, with posterior mean function $\mu_t(\mathbf{x}_*)$ and predictive variance $\sigma_t^2(\mathbf{x}_*)$:

$$f(\mathbf{x}_*)|\mathbf{f}_{1:t}, \mathbf{x}_1, \ldots, \mathbf{x}_t \sim \mathcal{N}\left(\mu_t(\mathbf{x}_*), \sigma_t^2(\mathbf{x}_*)\right), \tag{7}$$

$$\mu_t(\mathbf{x}_*) = m(\mathbf{x}_*) + \mathbf{k}(\mathbf{x}_*, \mathbf{x}_{1:t})\left[K(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}) + \tau^2 I\right]^{-1}\left(\mathbf{f}_{1:t} - m(\mathbf{x}_*)\right), \tag{8}$$

$$\sigma_t^2(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}(\mathbf{x}_*, \mathbf{x}_{1:t})\left[K(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}) + \tau^2 I\right]^{-1}\mathbf{k}(\mathbf{x}_{1:t}, \mathbf{x}_*), \tag{9}$$

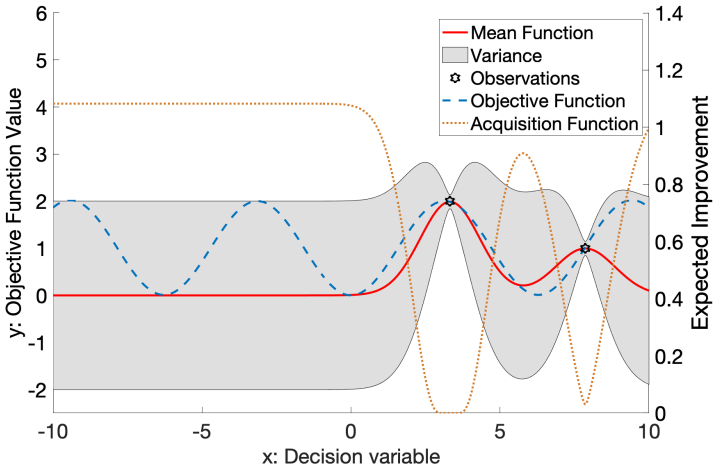


Figure 1: Illustration of a GP posterior and acquisition function for a 1-D optimization problem.

where I is the identity matrix. Eq. (8) indicates that the posterior mean function $\mu_t(\mathbf{x}_*)$ is fitted according to the difference between the observed objective function value and the prior mean function value (i.e. $\mathbf{f}_{1:t} - m(\mathbf{x}_*)$). Eq. (9) shows that the predictive variance $\sigma_t^2(\mathbf{x}_*)$ of the posterior GP is reduced in areas where there is strong covariance with existing observations (i.e. when the term $\mathbf{k}(\mathbf{x}_*, \mathbf{x}_{1:t}) \left[K(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}) + \tau^2 I\right]^{-1} \mathbf{k}(\mathbf{x}_{1:t}, \mathbf{x}_*)$ is large). Note that the computation of the matrix inversion $\left[K(\mathbf{x}_{1:t}, \mathbf{x}_{1:t}) + \tau^2 I\right]^{-1}$ in Eq. (8) and (9) has a computational complexity of $O(t^3)$, and may pose a challenge when there is a large number of observations (e.g. 10^5 observations). However, in the case of SO, the number of observations tends to be too small for this to be a problem.

A 1-D example of the GP posterior is illustrated in Figure 1. The two stars represent the observations which the GP was fitted to, giving rise to the posterior mean function and the predictive variance as depicted by the solid line and shaded region respectively. In this figure, the shaded region shows the values $\pm\sigma$ away from the posterior mean function. The objective function is represented by the dashed line. The global minimum lies at $x = 0$, with local minima at around $x = \pm 6.3$. The GP posterior can be interpreted as follows: at any point $\mathbf{x}_*$ in the feasible region, the GP posterior predicts that the corresponding objective function value is normally distributed with mean $\mu(\mathbf{x}_*)$ and variance $\sigma^2(\mathbf{x}_*)$[1], as given by Eq. (8) and (9). In other words, for given point x_* in Figure 1, the GP posterior predicts the objective function value to be normally distributed about $\mu(x_*)$, with the shaded region showing the values $\pm\sigma$ away from the mean. The observations provide information about the objective function at those points, thus reducing the predictive variance of the GP around those points. However, as the objective function being modeled is noisy, there is still a non-zero variance around the observations. In regions far from the observations, the GP posterior mean function tends towards the prior mean function, which is zero in this example.

3.1.2 Acquisition Function.

Given the posterior mean function and the predictive variance of the GP posterior, the acquisition function is used to identify the next point to evaluate. The role of the acquisition function is to balance exploitation and exploration when selecting the next evaluation point. In a minimization (resp. maximization) problem, selecting a point with a small (resp. large) posterior mean function value corresponds to exploitation, while selecting a point with a large predictive variance represents exploration. The acquisition function, therefore, is a function of both $\mu(\mathbf{x}_*)$ and $\sigma^2(\mathbf{x}_*)$. The next point to evaluate is then chosen by maximizing the acquisition function.

Several acquisition functions with analytical expressions have been proposed in the BO literature, including the probability of improvement (Kushner 1964), expected improvement (Jones et al. 1998, Mockus et al. 1978) and upper confidence bound (Srinivas et al. 2009). In this work, we work with the expected improvement (EI)

[1]Note that the subscript t are frequently omitted from $\mu(\mathbf{x}_*)$ and $\sigma^2(\mathbf{x}_*)$ for simplicity, and it can be assumed to include all available observations unless otherwise stated.

acquisition function, which is defined as follows (for a minimization problem):

$$EI(\mathbf{x}_*) = \mathbb{E}\left[\max\{0, f_{\min} - f(\mathbf{x}_*)\}\right] \tag{10}$$

$$= (f_{\min} - \mu(\mathbf{x}_*))\Phi(Z) + \sigma(\mathbf{x}_*)\phi(Z), \tag{11}$$

$$\text{where } Z = \frac{f_{\min} - \mu(\mathbf{x}_*)}{\sigma(\mathbf{x}_*)} \tag{12}$$

where $f_{\min}$ denotes the smallest objective function estimate of all the observations; $\phi(\cdot)$ and $\Phi(\cdot)$ in Eq. (11) denote the probability density function (pdf) and cumulative distribution function (cdf) of the standard normal distribution respectively. In Eq. (11), the first component can be interpreted as the exploitation component, as it is proportional to the possible improvement in objective function value $f_{\min} - \mu(\mathbf{x}_*)$. The second component can then be interpreted as the exploration component, as it is proportional to the predictive variance. To select the next point to evaluate $\mathbf{x}_{t+1}$, we solve the following maximization problem:

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}_*} EI(\mathbf{x}_*). \tag{13}$$

EI was chosen as the acquisition function as it is more commonly used than probability of improvements, and also it does not require an additional tuning parameter, unlike the upper confidence bound acquisition function (Snoek et al. 2012). When working with noisy observations, the EI acquisition function may not be properly defined since the true distribution of f is not known (see e.g. Section 5.1 of Frazier (2018)). There exist alternative acquisition functions that may be more suitable for noisy observations, such as the knowledge gradient function (Frazier et al. 2009, Wu and Frazier 2016), the entropy search function (Hennig and Schuler 2012) and the predictive entropy search function (Hernández-Lobato et al. 2014). However, these alternative acquisition functions do not have analytical closed forms. This makes them computationally costly to evaluate, compared to the EI function. Furthermore, the convergence guarantees of the acquisition functions hold only in asymptotic cases. Since we are considering short-term performance, the convergence guarantees are less applicable in our case.

The EI acquisition function is also depicted in the 1-D example in Figure 1 by the dotted line. As can be seen in the figure, the EI acquisition function has the largest values where there is a combination of a small posterior mean function value and a large predictive variance. In particular, the largest EI values lie in the region $x < 0$ in this iteration, where there are currently no observations, indicating that the algorithm should explore in this region. Furthermore, the EI value is zero around $x = 3.5$, since the GP posterior predicts that a minimum is very unlikely to lie in that area. The predictive variance around $x = 3.5$ is also close to zero.

The optimization of the EI acquisition function is commonly carried out using DIRECT (Jones et al. 1993), which is a deterministic, derivative-free optimizer. Given that we are considering stochastic problems, this means that DIRECT is not a suitable optimizer. Since the analytical expression of the gradient of EI is available (Frean and Boyle 2008, Section 3), we use a multistart gradient ascent approach to maximize the acquisition function (Hutter et al. 2011, Shahriari et al. 2015). More specifically, we use the *MultiStart* routine in Matlab, along with *fmincon* as the solver.

3.1.3 Algorithm.

The general BO algorithm is summarized in Algorithm 1[2]. The GP hyperparameters are first initialized. A set of random initial points in the feasible region are then sampled and evaluated, providing the initial observations for fitting the GP posterior. At every iteration, the GP hyperparameters are fitted to the observations through maximum likelihood estimation (for details, see Section 5.4.1 of Williams and Rasmussen (2006)). Then, the acquisition function, which is based on the GP posterior (see Eq. (10)-(12)), is maximized to select the next point for evaluation, as mentioned in Section 3.1.2. This is repeated until the optimization budget of T iterations is exceeded.

3.2 Gaussian Process with Problem-Specific Information

For some optimization problems, we may have access to problem-specific prior information or have some prior beliefs about the shape of the objective function. For instance, in transportation problems involving an urban road network, the underlying road network would be known beforehand. Hence, it may be possible to derive an

[2]In this work, the GP models were implemented using the GPML package for Matlab (Rasmussen and Nickisch 2018).

Algorithm 1: Bayesian optimization

1. **Initialization**
   (a) Choose a prior mean function and a covariance function
   (b) Initialize GP hyperparameters: $\alpha, \beta, \sigma_0^2, \ell, \ell^{f^A}$ (if applicable; see Section 3.2 for more details about the hyperparameters α, β and ℓ^{f^A})
   (c) Randomly sample t_0 points $\mathbf{x}_{1:t_0}$ from the feasible region and evaluate through simulation to obtain $\mathbf{f}_{1:t_0}$
2. **Optimization**
   for $t = t_0, \ldots, T-1$
   (a) Update GP hyperparameters through maximum likelihood estimation based on the data $\{\mathbf{x}_{1:t}, \mathbf{f}_{1;t}\}$ (see e.g. Section 5.4.1 of Williams and Rasmussen (2006))
   (b) Fit the GP to the data $\{\mathbf{x}_{1:t}, \mathbf{f}_{1;t}\}$ to obtain the posterior mean function $\mu_t(\mathbf{x}_*)$ (Eq. (8)) and the predictive variance $\sigma_t^2(\mathbf{x}_*)$ (Eq. (9))
   (c) Identify the next point to evaluate $\mathbf{x}_{t+1}$ by maximizing the acquisition function (Eq. (13))
   (d) Evaluate the point through simulation to obtain f_{t+1}

analytical transportation model $f^A(\cdot)$ that approximates or correlates with the objective function. In this section, we show how f^A can be incorporated in the GP prior mean function (Section 3.2.1) and the covariance function (Section 3.2.2) in order to enable efficient high-dimensional BO.

3.2.1 Prior Mean Function.

As mentioned in Section 3.1.1, a constant prior mean is typically chosen when there is no available prior information about the objective function:

$$m(\mathbf{x}) = \beta, \tag{14}$$

where β is a constant. In the case of a zero prior mean function, β is taken to be 0.

However, given an analytical transportation model $f^A(\cdot)$ that approximates the objective function, it would be natural to use it in the prior mean function:

$$m(\mathbf{x}) = \alpha f^A(\mathbf{x}), \tag{15}$$

where α is a scaling constant that can be fitted according to the data. Assuming f^A approximates the objective function well, it can inform the acquisition function about the locations of possible optima through the posterior mean function as shown in Eq. (8). This thus allows BO to exploit the problem-specific prior information to efficiently identify good solutions even at higher dimensions. This is illustrated using a 1-D example in Section 4.3.

3.2.2 Covariance Function.

The choice of the covariance function can give rise to more interesting GP posteriors. As shown by Eq. (8) and (9), the choice of the covariance function affects both the posterior mean function and the predictive variance. The covariance between two points, as defined by the covariance function, represents how correlated the two points are believed to be. Standard covariance functions, such as the squared exponential covariance function (Eq. (3)), work with the assumption that points close to one another have similar objective function values (i.e. the objective function is smooth). In the case of the squared exponential covariance function, the exponential term ensures that $k(\mathbf{x}_i, \mathbf{x}_j) \to \sigma_0^2$ when $\mathbf{x}_i$ and $\mathbf{x}_j$ are close to each other in Euclidean distance, and $k(\mathbf{x}_i, \mathbf{x}_j) \to 0$ when the Euclidean distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ becomes large. In fact, this is a necessary condition for BO to converge to the optimum (Mockus 1994).

However, other than assuming a smooth objective function, the standard covariance functions do not exploit any problem structure at all. Here, we propose a covariance function that is able to incorporate problem-specific prior information in the form of an analytical transportation model f^A to efficiently tackle high-dimensional BO problems:

$$k^{f^A}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_0^2 \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\ell^2}\right) \exp\left[-\frac{\left(f^A(\mathbf{x}_i) - f^A(\mathbf{x}_j)\right)^2}{2(\ell^{f^A})^2}\right], \tag{16}$$

where ℓ^{f^A} is the analytical transportation model length-scale. The analytical transportation model-based covariance function k^{f^A} differs from the standard squared exponential covariance function k^{SE} in Eq. (3) in the addition of a second exponential component. This additional exponential component uses the difference between the f^A values of the two points to further attenuate the covariance between the two points. Put differently, the difference in f^A values is used as an additional distance measure, which incorporates the problem-specific prior information, to determine how correlated the two points are. This means that for a given Euclidean distance, we are incorporating the prior belief that two points with similar f^A values have high covariance and vice versa. At the same time, the first exponential component in Eq. (16) ensures, that when the Euclidean distance between the two points is large, the covariance between the two points goes to zero, reflecting the uncertainty of making predictions in regions without any observations. The hyperparameter ℓ^{f^A} effectively determines how big of a difference in f^A values is required for the two points to become uncorrelated.

The squared exponential covariance function forms the basis for k^{f^A} in Eq. (16), allowing us to directly compare the performance of k^{f^A} and k^{SE} to determine the added value of using the analytical transportation model in the covariance function. However, we would like to emphasize that the analytical transportation model component can be added to any other covariance functions, such as the Matérn covariance functions. Hence, the proposed method of incorporating problem-specific information in the covariance function is highly generalizable.

The proposed covariance function k^{f^A} is able to encourage exploration of parts of the feasible region with different analytical transportation model values than previously evaluated points (i.e. observations). As seen in Eq. (9), the predictive variance is reduced in regions with non-zero covariance to previously evaluated points. Using k^{SE} as the covariance function means that only the regions with nearby observations in the Euclidean distance sense have reduced posterior variance. However, using k^{f^A} as the covariance function allows for points that are further away from observations to have greater covariance if they have similar f^A values as the previously evaluated points, thus resulting in a smaller predictive variance. Hence, with the predictive variance being attenuated based on the difference in f^A values compared to those of the previously evaluated points, there is a larger predictive variance in regions with f^A values that are different from those of the evaluated points. Since the acquisition function (Eq. (11)) assigns greater values to regions with high predictive variance, exploration in these regions are encouraged. The effect of using k^{f^A} as the covariance function is illustrated using a 1-D problem in Section 4.3.

The effect of using k^{f^A} is particularly significant in higher dimensional problems where there is a limited number of observations. The limited number of observations results in a poor coverage of the high-dimensional feasible region. This makes it difficult to model the objective function well if using a GP prior with no problem-specific prior information. However, by using k^{f^A} as the covariance function, both the posterior mean function and the predictive variance will have access to f^A (see Eq. (8) and (9)). This helps to tackle the two problems which plague conventional high-dimensional BO to some extent. First, while the limited number of observations provides a poor coverage of the high-dimensional feasible region, it provides a decent coverage of the 1-D space of f^A difference. In regions with no nearby observations by Euclidean distance, there could still be significant covariance between points with similar f^A values as the previously evaluated points. This helps to inform the GP on possible objective function values in the unexplored regions, based on similarities in f^A values. Second, when using k^{f^A} as the covariance function, the posterior mean function and the predictive variance are informed on possible objective function values in unexplored regions. This means the posterior mean function and the predictive variance are less likely to be constants in the unexplored regions. As a result, the acquisition function would have fewer flat regions, allowing it to be more easily optimized using a gradient ascent approach. Hence, this can allow BO to be effective in high-dimensional problems. Section 4.3 provides an illustration of how the use of k^{f^A} as the covariance function can help to tackle the problems of conventional high-dimensional BO.

3.2.3 Computational Trade-off

It should be mentioned that the use of an analytical transportation model in the GP involves a computational trade-off. The analytical transportation model approximation f^A requires some computational time to evaluate each time. Hence, it's use in the prior mean function (Eq. (15)) (resp. the covariance function (Eq. (16))) could lead to greater computational runtimes compared to using the constant prior mean function (Eq. (14)) (resp. the standard squared exponential covariance (Eq. (3))). The increase in computational runtime also depends on the analytical transportation model being used. Hence, the chosen model should ideally be quick to evaluate. In general, the increase in computational runtimes could be worthwhile if the use of f^A in the prior mean function and/or the covariance function can lead to more efficient optimization.

4 Validation and Illustration

In this section, we validate the proposed method of incorporating problem-specific prior information in the prior mean function and the covariance function of the GP works using the Griewank function as the objective function. We first define the Griewank function and introduce the optimization problem in Section 4.1. Then, we present the different benchmark methods in Section 4.2. In Section 4.3, we validate the proposed method using the 1-D Griewank function. Following that, we show that the proposed method is able to tackle high-dimensional BO using a 100-D Griewank function for illustration in Section 4.4. In Section 4.5, we illustrate the effects that biases in the analytical model can have on the optimization performance.

4.1 Griewank Function

The D-dimensional Griewank function (Griewank 1981) is defined as follows:

$$g(\mathbf{x}) = 1 + \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right), \tag{17}$$

where $\mathbf{x} = [x_1, \ldots, x_D]^T$. The Griewank function was chosen, as it is a continuous and non-convex function with multiple local minima – properties which are present in many objective functions of optimization problems in transportation. Furthermore, the Griewank function is easily generalizable to any number of dimensions, allowing us to test the proposed method on problems of any dimension.

To make it a stochastic problem, we add i.i.d. Gaussian noise ϵ with mean 0 and variance 0.01 to the Griewank function $g(\mathbf{x})$ (Eq. (18)). In addition, we set the feasible region as $[-10, 10]^D$ (Eq. (20)). The noisy D-dimensional Griewank function optimization problem that we consider in this section is summarized by Eq. (18)-(20):

$$\min_{\mathbf{x}} f(\mathbf{x}) = g(\mathbf{x}) + \epsilon, \tag{18}$$

$$\text{subject to } \epsilon \sim \mathcal{N}(0, 0.01), \tag{19}$$

$$x_i \in [-10, 10] \quad \forall i = 1, \ldots, D. \tag{20}$$

4.2 Benchmark Methods

In this problem, we assume there is prior knowledge that the objective function (Eq. (18)) has a quadratic component. Based on this, we choose a quadratic model as the analytical model:

$$f^A(\mathbf{x}) = \|\mathbf{x}\|_2^2. \tag{21}$$

Note that f^A in Eq. (21) is not an accurate model of the Griewank function, in the sense that it does not reflect the undulations and local minima. However, it still has significant correlation with the Griewank function, by following the same general trend. In fact, the minimum of f^A is perfectly aligned with the global minimum of the Griewank function at $\mathbf{x} = \mathbf{0}$.

To evaluate our proposed method, we make use of 4 different GP priors for the Griewank function optimization problems, as well as the case study in Section 5. The 4 different GP priors are summarized in Table 1, and they differ according to whether (i) the prior mean function and (ii) the covariance function are based on the analytical model f^A or not. The first column of Table 1 defines the names of the 4 GP priors used. The second column indicates whether the prior mean function is based on the analytical model as defined in Eq. (15). If not, the

constant prior mean in Eq. (14) is used. The third column indicates whether the covariance function is based on the analytical model as shown in Eq. (16). If it is not, the standard squared exponential covariance function given in Eq. (3) is used.

The Standard GP prior represents a general-purpose GP prior, which does not use any problem-specific information, and is taken as a benchmark for the other 3 proposed GP priors that make use of problem-specific information in the prior mean function and/or the covariance function. The comparison between Standard and Proposed-Covariance (resp. Proposed-Mean) allows us to evaluate the effect of using a problem-specific covariance function (resp. prior mean function). Comparing Proposed-Combined with Proposed-Covariance (resp. Proposed-Mean) serves to evaluate the added value of using the problem-specific information in the prior mean function (resp. covariance function).

4.3 1-D Griewank Function

Using the 1-D Griewank function, we first illustrate how the choice of GP prior affects the posterior when fitted to initial observations. For this example, 2 initial observations (Step 1c of Algorithm 1) were provided for fitting the GP posteriors, with the same pair of initial observations used across all 4 GP priors. For each observation, the objective function estimate was obtained by taking the mean of 4 simulations (i.e. 4 random draws of the noisy Griewank function in Eq. (18)). The fitting of the GP posterior was done by updating the GP hyperparameters through maximum likelihood estimation (see Step 2a of Algorithm 1), using the *minimize* function (which minimizes the negative log marginal likelihood) found in the GPML package for Matlab (Rasmussen and Nickisch 2018).

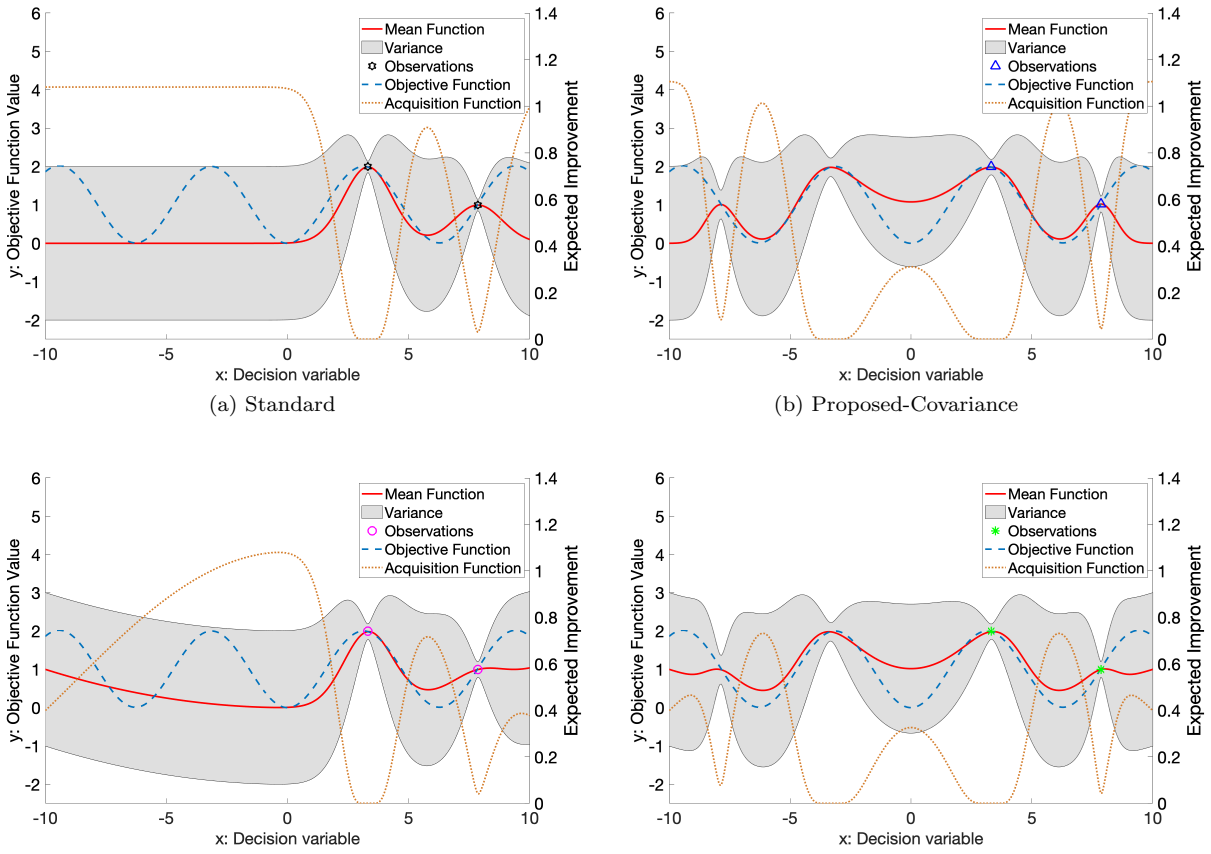
Figure 2 plots the 4 different GP posteriors after fitting to the initial observations, along with the objective function and their respective EI acquisition functions. In each plot, the x-axis represents the feasible region for this 1-D case, while the left and right y-axes represent the objective function value and EI value respectively. The initial observations are depicted by the two markers, with the marker styles and colors differing based on the GP prior. The marker styles and colors are consistent with those used in subsequent figures comparing the different GP priors. The posterior mean function is represented by the red solid line, while the shaded region represents the predictive variance and shows the values $\pm\sigma$ away from the posterior mean function. The objective function is depicted by the blue dashed line, with the global minimum occurring at $x = 0$ along with two other local minima in this 1-D case. The EI acquisition function for this iteration is shown by the orange dotted line.

The GP posterior for Standard is shown in Figure 2a. It can be observed that in the regions close to the observation, the posterior mean function models the objective function relatively well, while reducing the predictive variance around the observations. Correspondingly, the EI function is assigned a high value where there is a combination of a small posterior mean function value and large predictive variance, and vice versa. However, in the region with no observations (i.e. $x < 0$), we see that the posterior mean function simply tends towards the constant prior mean function, while the predictive variance goes towards σ_0^2. Since the posterior mean function and the predictive variance are essentially constant in the region $x < 0$, the EI function is also flat. As mentioned in Section 2, the acquisition function can have many flat regions and can be multimodal. This is especially true in higher dimensions. Hence, Figure 2a shows why it can be difficult to find the global maximum of the acquisition function for high-dimensional problems.

In the case of Proposed-Covariance (Figure 2b), the GP posterior mean function and the predictive variance are no longer constant in the region $x < 0$ despite the lack of observations, unlike Standard. In fact, the GP posterior predicts similar objective function values for points with similar f^A values, which can be seen by noting that the posterior mean function is almost symmetric about $x = 0$ and that f^A is a symmetric function about $x = 0$. Note that even though Proposed-Covariance does not use an informative prior mean function, its posterior mean function is affected by k^{f^A} (see Eq. (8)). Hence, this explains the symmetry in the posterior mean function.

Table 1: GP Priors

Name	**Prior Mean Function** Based on analytical model?	**Covariance Function** Based on analytical model?
Standard		
Proposed-Covariance		✓
Proposed-Mean	✓	
Proposed-Combined	✓	✓



(a) Standard

(b) Proposed-Covariance

(c) Proposed-Mean

(d) Proposed-Combined

Figure 2: GP posteriors fitted to 2 initial observations for the 1-D Griewank function.

In addition, the predictive variance around points with similar f^A values as the previously evaluated points were also reduced as expected. As a result of the behavior of the posterior mean function and the predictive variance in the region without observations ($x < 0$), the EI function is no longer flat, which is important when using a gradient-ascent approach when maximizing the acquisition function.

Proposed-Mean (Figure 2c) is similar to Standard, in that the posterior mean function tends towards the prior mean function in the region $x < 0$ which has no observations. The only difference is that the prior mean function in Proposed-Mean is $m(x) = x^2$, thus explaining why the posterior mean function increases quadratically as x becomes more negative in $x < 0$. At the same time, the predictive variance goes towards σ_0^2, since the covariance function in Proposed-Mean does not exploit any problem-specific information. As a result of the non-constant posterior mean function in the region $x < 0$, the EI function is also no longer flat, unlike the case for Standard.

The GP posterior of Proposed-Combined (Figure 2d) is similar to that of Proposed-Covariance, in that similar objective function values are predicted for points with similar f^A, due to the use of the problem-specific covariance function. The predictive variance is also reduced for points with similar f^A values as the previously evaluated points. The main difference between the GP posteriors of Proposed-Combined and Proposed-Covariance is that the posterior mean function of Proposed-Combined has an additional quadratic component, leading to larger μ values for points further away from $x = 0$ as compared to that of Proposed-Covariance. This difference is also reflected in the EI function, where the peaks at around $x = \pm 6$ and $x = \pm 10$ for Proposed-Combined are smaller than those in Proposed-Covariance. In fact, the EI peak near $x = \pm 10$ is smaller than the one near $x = \pm 6$ for Proposed-Combined, unlike Proposed-Covariance. This shows how f^A in the prior mean function can bias the posterior mean function, thus affecting the EI function. If f^A is a good approximation of the objective function, this could allow the optimization problem to solved more quickly.

The illustrations of the GP posteriors for the 4 different priors in Figure 2 shows how problem-specific information in the form of an analytical model can help when used in the prior mean function and/or the covariance

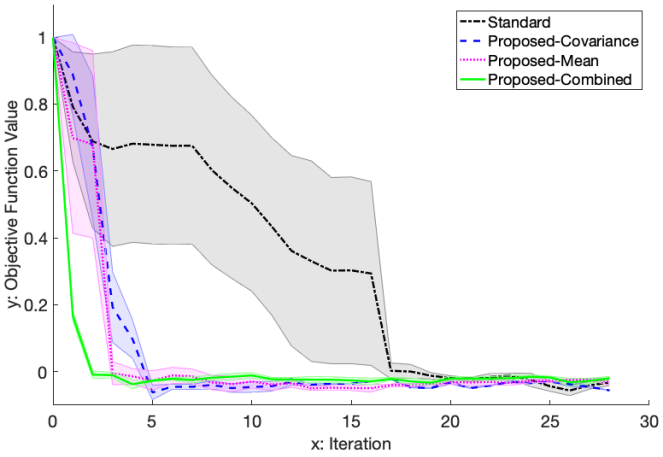


Figure 3: Optimization performance of the different GP priors for the 1-D Griewank function problem.

function, by making the EI function less flat. In particular, the use of k^{f^A} as the covariance function in Proposed-Covariance and Proposed-Combined allows the GP model to more efficiently model the objective function using fewer observations, by taking advantage of the correlation between f^A and the objective function. As a consequence, the k^{f^A} covariance function encourages the exploration of points with different f^A values from those of the previously evaluated points to further improve the model of the objective function. Hence, this is how the k^{f^A} covariance function can make high-dimensional BO more efficient.

Furthermore, it should also be noted that the quadratic component in the objective function is relatively weak compared to the sinusoidal component – the weight of the cosine term is much greater than the quadratic term (see Eq. (17)). The quadratic model f^A captures only the quadratic trend, but does not accurately model the sinusoidal component of the objective function. However, as seen in Figures 2b and 2d, the k^{f^A} covariance function still allows the correlation between f^A and the objective function to be exploited when modeling the objective function. In particular, it infers that the objective function is symmetric about $x = 0$ based on f^A. Hence, this suggests that models that capture a general trend in the objective function are sufficient. They do not need to accurately model the objective function. This opens the door for more models to be used as f^A in other problems.

Next, we look at the optimization performance using the 4 different GP priors. After obtaining the 2 initial observations, we continue with the Optimization step (Step 2) of Algorithm 1. At every iteration, a new point is identified by maximizing the EI acquisition function, and the objective function estimate of that point is taken as the mean of 4 simulations. In addition, the best solution at the end of each iteration is identified and simulated 2 more times, and the objective function estimate is updated as the mean of all the simulations for that solution up to that iteration. This is so as to account for the noise and obtain a more accurate objective function estimate of the current best solution. The algorithm stops when the computational budget is reached. For this example, the computational budget was taken to be 30 iterations in total, including the 2 initial observations (i.e. 28 optimization iterations).

Figure 3 shows the performance of BO with the 4 different GP priors. The x-axis shows the optimization iteration number, while the y-axis shows the objective function estimate of the best solution at a given iteration. The optimization algorithm was run 3 times for each GP prior. Each line in Figure 3 depicts the mean of the best objective function estimate of the 3 runs, while the shaded regions represent the values ± 1 standard error away from the mean. Standard corresponds to the black dash-dot line, Proposed-Covariance to the blue dashed line, Proposed-Mean to the magenta dotted line, and Proposed-Combined to the green solid line. In this plot, the sooner the curve reaches a y-value of 0, the more efficient the GP prior is for optimization.

From Figure 3, we observe that all 4 GP priors were able to find a solution with objective function value close to the global minimum (i.e. 0), as expected for a 1-D problem. Hence, this shows that the 4 GP priors works at low dimensions. Furthermore, Figure 3 also shows that the GP priors that make use of problem-specific information in the prior mean function and/or the covariance function (i.e. Proposed-Covariance, Proposed-Mean and Proposed-Combined) were able to identify a solution with objective function value close to the global minimum more quickly

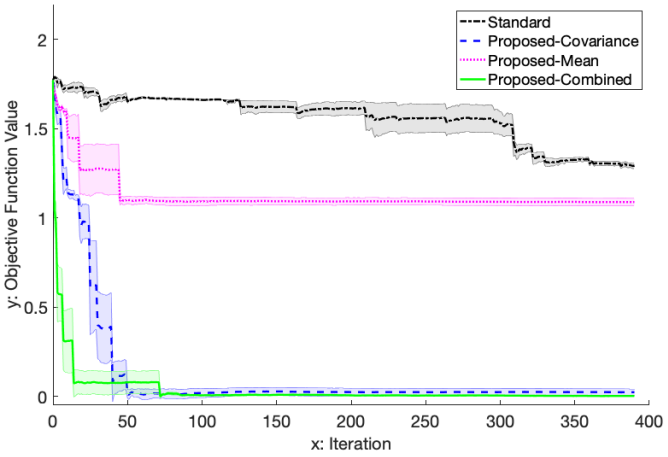


Figure 4: Optimization performance of the different GP priors for the 100-D Griewank function problem.

than Standard. Hence, even for a 1-D problem, having access to problem-specific information makes BO more efficient.

4.4 100-D Griewank Function

We next consider the 100-D noisy Griewank function as the objective function. This can be considered a high-dimensional problem in the context of BO, which has typically been limited to problems of less than 10 dimensions (Wang et al. 2016, Kandasamy et al. 2015).

For this 100-D problem, Algorithm 1 was implemented with 10 initial observations provided for fitting the GP posteriors, and a computational budget of 400 iterations (i.e. 390 optimization iterations). The set of 10 initial observations were sampled uniformly at random from the feasible region, with the same set of 10 observations used across the 4 different GP priors. Similar to the 1-D problem, for each observation, the objective function estimate was obtained by taking the mean of 4 simulations (i.e. 4 random draws of the noisy Griewank function in Eq. (18)). In addition, at every iteration, the best solution at the end of each optimization iteration is identified and simulated 2 more times, and the objective function estimate is updated as the mean of all the simulations for that solution up to that iteration. However, instead of updating the GP hyperparameters at every iteration (Step 2a of Algorithm 1), the hyperparameters were kept fixed throughout the optimization run as it was observed that this resulted in better optimization performance (see Appendix B for details on hyperparameter selection). The poorer optimization performance when the GP hyperparameters were updated was likely due to the maximum likelihood estimation having trouble finding a good set of hyperparameter values, resulting in large fluctuations in the estimated hyperparameter values at each iteration.

The BO performance of the 4 different GP priors are illustrated in Figure 4. Each line in Figure 4 depicts the mean of the best objective function estimate of the 3 runs, while the shaded regions represent the values ± 1 standard error away from the mean. As can be seen from Figure 4, the best solution identified by Standard had an objective function value of about 1.3, and it was unable to find a solution close to the minimum within the limited computational budget. This is expected given the high-dimensional nature of the problem, and no problem-specific prior information was provided in the prior for Standard. Proposed-Mean outperformed the solutions of Standard, but was unable to find a solution close to the global minimum. On the other hand, Proposed-Covariance and Proposed-Combined, which make use of the k^{f^A} as the covariance function, were able to consistently find a solution close to the minimum of 0 within 75 iterations. Hence, this shows the effectiveness of using the problem-specific information in the covariance function for BO in a high-dimensional setting. In addition, Proposed-Combined was able to find a solution close to the minimum faster than Proposed-Covariance, suggesting that the additional access to the problem-specific information in its prior mean function does make the optimization more efficient.

The effectiveness of the k^{f^A} covariance function in finding solutions with objective function values close to the global minimum of 0 can be attributed to its ability to get the BO algorithm to explore the f^A-space (i.e. evaluate points with different f^A values from previously evaluated points). By exploring the f^A-space, the the

algorithm would eventually be able to find a solution with objective function value close to the global minimum, especially if the minimum of f^A coincides with the global minimum of the objective function, and the f^A value at the global minimum is unique (i.e. $f^A = 0$ does not occur anywhere else in the feasible region). As illustrated by the 1-D Griewank function example, using the k^{f^A} covariance function (Figures 2b and 2d) results in the posterior GP predicting similar objective function values for points with similar f^A values. At the same time, the predictive variance is also reduced for points with similar f^A values as the previously evaluated points. As a result, points with different f^A values from the previously evaluated points have greater predictive variances, which naturally encourages exploration for these points.

The amount of exploration in the f^A-space done by the 4 different GP methods, along with the corresponding objective function values f, is shown in Figure 5. Each row of plots represent one of the 4 different GP priors. The left column of Figure 5 consists of 2-D histograms illustrating the distribution of f^A values (x-axis) and the corresponding f values (y-axis) of the evaluated points. Each histogram shows the distributions for the different GP priors. The histograms were plotted using data aggregated over 3 algorithm runs (i.e. $400 \times 3 = 1200$ observations). In each histogram plot, the color of the bin represents the number of observations that fall within that bin, with a lighter color meaning a greater number of observations. Note that the scale of the color bar differs for each GP prior.

The right column of Figure 5 plots the f^A values (x-axis) and the corresponding f values (y-axis) for every observation, aggregated over 3 algorithm runs. The contour lines further represent the 2-D empirical cumulative distribution function (ecdf) of f^A and f based on the observations, which is defined as

$$ecdf(f^A, f) = \frac{1}{N} \times |\{(f_i^A, f_i) : f_i^A \leq f^A, f_i \leq f\}|, \tag{22}$$

where N is the total number of observations (i.e. 1200), and $|\cdot|$ denotes the cardinality of a set. In words, the ecdf, as defined in Eq. (22), at a given point $(f^{A'}, f')$ is the fraction of observations that have f^A and f values less than or equal to $f^{A'}$ and f' respectively. In the plots, the contour lines denote the 2-D ecdf in steps of 0.1 (i.e. going from one contour line to the next in the increasing contour direction indicates a gain in ecdf of 0.1 and vice versa). In other words, the space between 2 adjacent contour lines contains 10% of the total number of observations. Hence, the 2-D ecdf contour lines help to better visualize the distribution of observations in the plots. The histogram (left column) and ecdf (right column) of Figure 5 present the same 2-D distribution of observations in different ways. However, the ecdf does it without having to discretize (i.e. bin) the f^A and f values. Hence, subsequent results (Figures 7 and 11) will be presented using the 2-D ecdf.

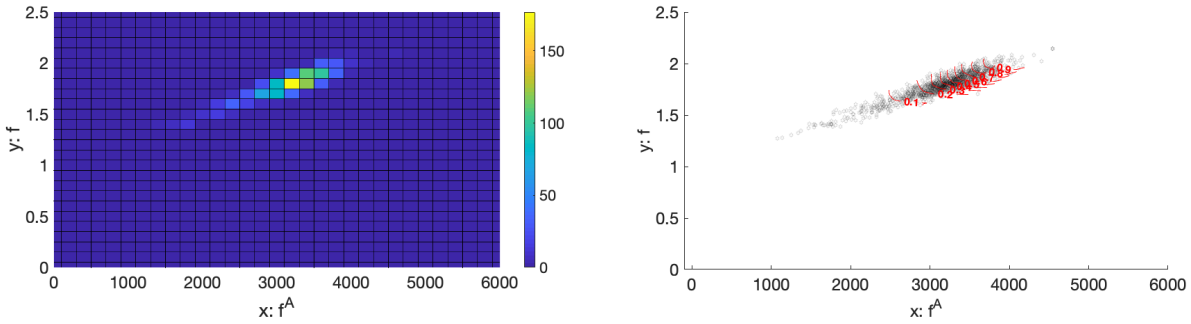
From Figure 5, it can be seen that the observations of Proposed-Covariance and Proposed-Combined (Figures 5d and 5h respectively) cover a much larger range of f^A values than those of Standard and Proposed-Mean (Figures 5b and 5f respectively), showing that the use of the k^{f^A} covariance function does indeed encourage exploration in the f^A-space. Note that the f^A can also be viewed as a 1-D projection of $\mathbf{x}$. Hence, exploration in the f^A-space can be used a proxy to visualize exploration in the high-dimensional $\mathbf{x}$-space.

Furthermore, since the minimum of f^A coincides with the minimum of f and the f^A value at the minimum is unique in this problem, Proposed-Covariance and Proposed-Combined can easily find solutions with f close to 0 when exploring in regions with f^A values close to 0, as shown by the observations in the lower left corner of Figures 5d and 5h. On the other hand, Standard and Proposed-Mean, which do not use the k^{f^A} covariance function, do not even explore regions with f^A values below 200. This can reduce their chances of finding solutions with f values close to 0. Therefore, Figure 5 helps to show how the k^{f^A} helps to tackle high-dimensional problems.

4.5 100-D Griewank Function with New Types of Biases

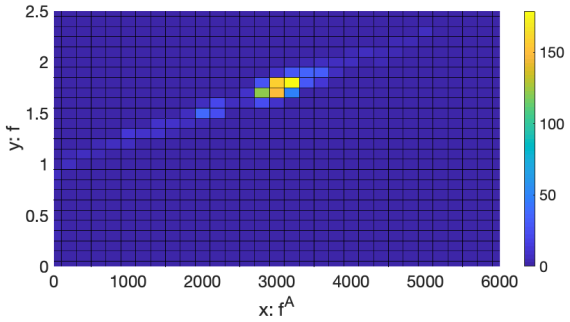
In choosing the analytical model f^A for the Griewank function problem, we picked a quadratic model (Eq. (21)) such that the minimum of the quadratic model lies at the same position as the global minimum of the Griewank function (i.e. $\mathbf{x} = \mathbf{0}$), which resulted in very efficient optimization for Proposed-Covariance and Proposed-Combined. However, in real-life applications, it is highly unlikely that the chosen analytical model would have an optimum that perfectly aligns with the optimum of the objective function. Here, we further consider scenarios where there are more biases in f^A in modeling the objective function to see if there is still added value in using a model with biases in the prior mean function and/or covariance. The different types of biases in f^A that we consider are:

- **Inverted** (i.e. $f^A(\mathbf{x}) \leftarrow -f^A(\mathbf{x})$): The analytical model is inverted so that it is a completely inaccurate model of the objective function. Instead, the inverted model is now anti-correlated with the objective

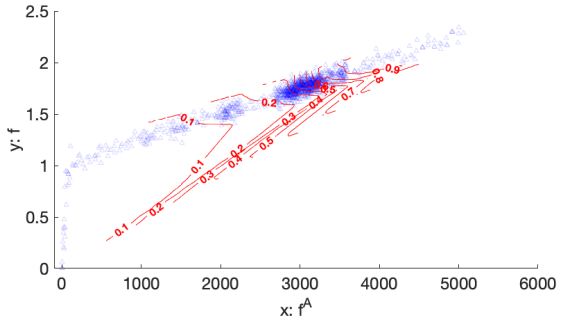


(a) Standard (Histogram)

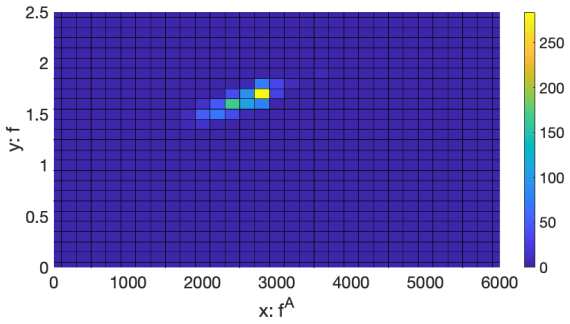
(b) Standard (2-D ecdf)



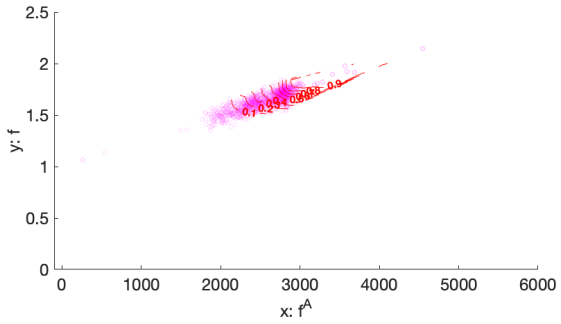
(c) Proposed-Covariance (Histogram)



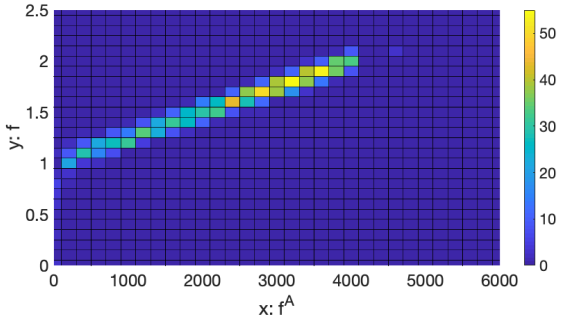
(d) Proposed-Covariance (2-D ecdf)



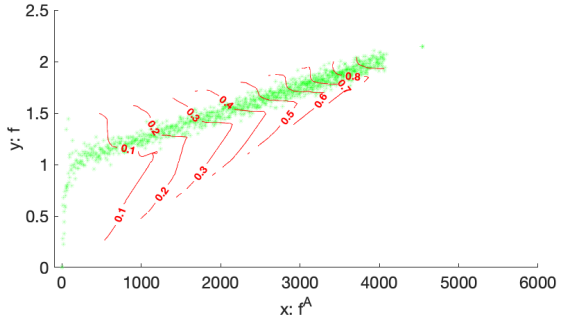
(e) Proposed-Mean (Histogram)



(f) Proposed-Mean (2-D ecdf)



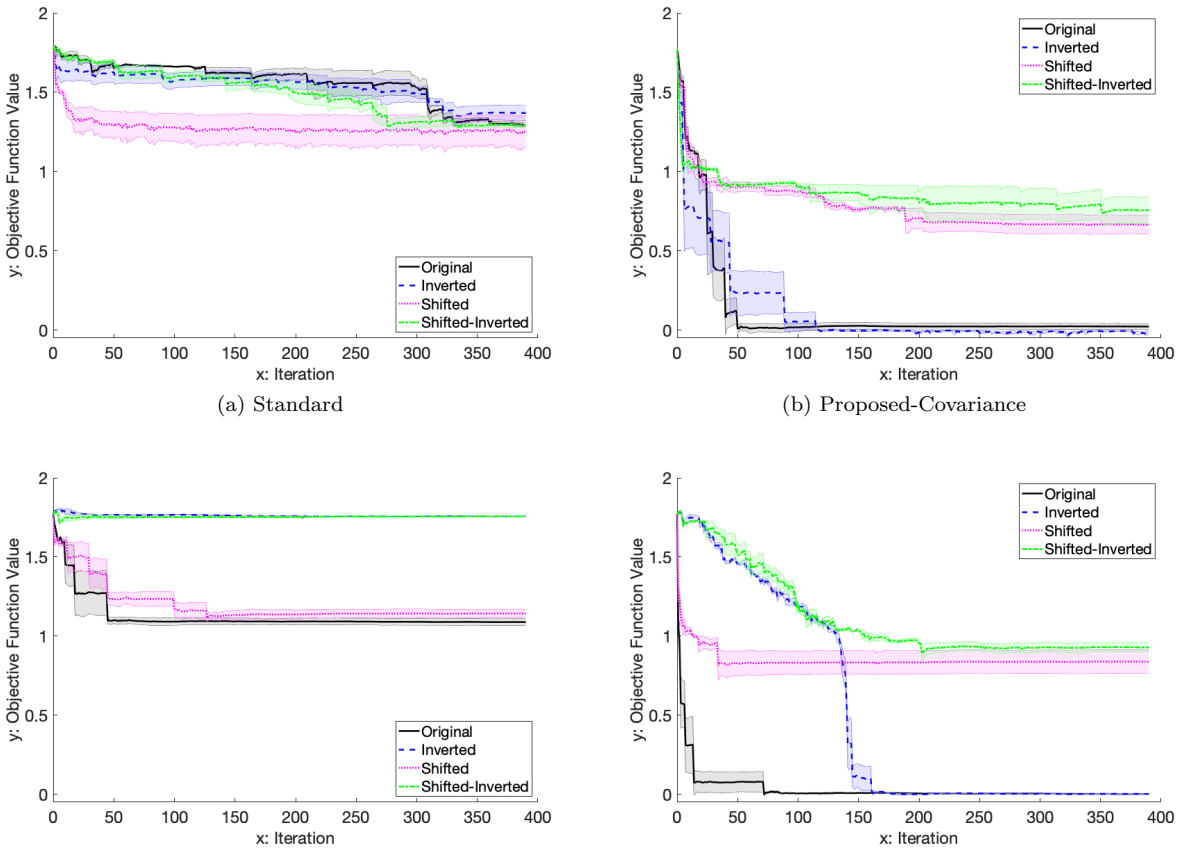
(g) Proposed-Combined (Histogram)



(h) Proposed-Combined (2-D ecdf)

Figure 5: Histograms and 2-D empirical cumulative distribution functions illustrating exploration in f^A-space and the corresponding objective function values.

function, with its maximum coinciding with the objective function minimum at $\mathbf{x} = \mathbf{0}$. This illustrates the



(a) Standard (b) Proposed-Covariance

(c) Proposed-Mean (d) Proposed-Combined

Figure 6: Optimization performance of the different GP priors for the 100-D Griewank function, when bias is introduced to f^A.

case where the analytical model is inversely biased in modeling the objective function, but is still able to provide information through (anti-)correlation.

- **Shifted** (i.e. $f^A(\mathbf{x}) \leftarrow f^A(\mathbf{x}-\mathbf{1})$): The analytical model is shifted in the $\mathbf{x}$-space, so that the minimum of the model lies at $\mathbf{x} = \mathbf{1}$ instead. This illustrates the case where the minimum of f^A and the global minimum of the objective function no longer coincide, leading to a reduction in correlation between f^A and the objective function.
- **Shifted-Inverted** (i.e. $f^A(\mathbf{x}) \leftarrow -f^A(\mathbf{x}-\mathbf{1})$): The analytical model is first shifted in the $\mathbf{x}$-space, before being inverted. This means that the maximum of f^A lies at $\mathbf{x} = \mathbf{1}$, and does not coincide with the global minimum of the objective function. While f^A still has some negative correlation with the objective function, the anti-correlation would not be as strong as that of the Inverted model, illustrating the case where f^A has a combination of the two biases above.

The optimization performance of the 4 different GP priors using the analytical models with the different biases are plotted in Figure 6. Each plot represents one of the 4 different GP priors. In each plot, each curve represents the mean of the best objective function estimate of 3 optimization run, with the respective shaded region depicting ± 1 standard error away from the mean. The black solid line represents the Original f^A, the blue dashed line shows the Inverted f^A, the magenta dotted line indicates the Shifted f^A, and the green dash-dot line corresponds to the Shifted-Inverted f^A.

Standard does not make use of f^A in the prior, hence the 4 curves in Figure 6a correspond to the optimization performance of the exact same GP prior, which should result in similar performance, yielding an average best objective function value of around 1.3 at the end of 390 optimization iterations. Any differences in the 4 curves are due to randomness during optimization.

Proposed-Covariance makes use of f^A in the covariance function. As can be seen from Figure 6b, Proposed-Covariance was still able to find a solution with objective function close to 0 when using the Inverted f^A, although it required more iterations on average than when using the Original f^A. This shows that Proposed-Covariance can be less sensitive to inversion bias in f^A. However, when using the Shifted f^A and Shifted-Inverted f^A, Proposed-Covariance was unable to find a solution with objective function close to 0 within the computational budget. This shows that as the correlation (or mutual information) between f^A and the objective function decreases, the added value of using k^{f^A} as the covariance function decreases too.

Proposed-Mean (Figure 6c) uses f^A only in the prior mean function. When using the Shifted f^A, the optimization performance showed a slight deterioration compared to that with the Original f^A, illustrating the impact of using an analytical model, where the minimum does not align with the global minimum of the objective function, in the prior mean function. Furthermore, as the set of hyperparameter values used is the same as that used for the Original f^A (i.e. $\alpha = 0.001$), this meant that the prior belief when using the Inverted and Shifted-Inverted f^A was that the objective function behaves like a negative quadratic function. Since this is untrue, the optimization performance of Proposed-Mean with the Inverted and Shifted-Inverted f^A was very poor. While choosing a suitable set of hyperparameter values would certainly help to improve the results when f^A is inverted, it can be difficult to find the right set of hyperparameter values in practice. Even if we choose to estimate the hyperparameter values (e.g. though maximum likelihood estimation), the limited number of observations poses a huge challenge to finding the right set of hyperparameter values for a high-dimensional problem.

Proposed-Combined (Figure 6d) takes advantage of f^A in both the prior mean function and the covariance function. When using the Inverted f^A, Proposed-Combined was still able to find a solution with objective function value close to 0. However, it required about 160 optimization iterations for the mean best objective function value to reach 0, compared with around 70 when using the Original f^A. This larger increase in number of iterations needed, compared with Proposed-Covariance, can be attributed to the fact that Proposed-Combined uses the Inverted f^A in the prior mean function as well, which makes it a bad prior. However, it was still able to recover, highlighting the usefulness k^{f^A} as the covariance function even if f^A is biased. When working with the Shifted and Shifted-Inverted f^A, Proposed-Combined was unable to find a solution with objective function value close to 0 within the computational budget. Similar to Proposed-Covariance, this highlights decreased benefits of using k^{f^A} as the covariance function when the correlation (or mutual information) between f^A and the objective function is reduced. Working with the Shifted f^A, Proposed-Combined was able to reach its best mean objective function value (around 0.8) with just 30 iterations, compared with 200 iterations when working with the Shifted-Inverted f^A. Again, this can be attributed to the inversion of f^A and its use in the prior mean function.

From Figure 6, we also see that Proposed-Covariance and Proposed-Combined with the Shifted and Shifted-Inverted f^A still perform better than Standard and Proposed-Mean. Even when working with the Shifted and Shifted-Inverted f^A, Proposed-Covariance and Proposed-Combined were still able to find solutions with objective function values below 1, while Standard and Proposed-Mean were unable to do so even when working with the Original f^A. This helps to show the effectiveness of the k^{f^A} covariance function in tackling high-dimensional problems, even when f^A does not perfectly model the objective function.

We next consider the amount of exploration done in the f^A-space by each GP prior, along with the corresponding objective function values f, for the different f^A models. Even though f^A is correlated to f, its minimum may not coincide with the minimum of f (as in Shifted and Shifted-Inverted). As such, considering f^A as a metric of exploration allows us to see if the different GP priors will search in regions with bigger f^A values in their attempts to minimize f. Figure 7 plots the observations and 2-D ecdfs to illustrate the distribution of f^A and f values explored. Each row of plots represent one of the 4 different GP priors, while each column represents one of the types of bias in f^A. The first column of plots shows the distribution of observations for the Original f^A, and are the exact same plots as the right column of Figure 5. Note that for the Inverted (second column) f^A and Shifted-Inverted (last column) f^A, the definition of the 2-D ecdf was modified to highlight the anti-correlation between f^A and f. The modified 2-D ecdf definition is

$$ecdf(f^A, f) = \frac{1}{N} \times |\{(f_i^A, f_i) : f_i^A \geq f^A, f_i \leq f\}|, \tag{23}$$

where the first "$\leq$" sign was changed to "$\geq$". This changes the direction of the contour lines, so that the ecdf value increases as f^A becomes more negative, allowing the distribution of points to be visualized more easily. Otherwise, the interpretation of the 2-D ecdf for Inverted f^A and Shifted-Inverted f^A remain the same. For Shifted f^A (third column), the 2-D ecdf as defined by Eq. (22) is used.

We first focus on the Standard GP prior (first row of Figure 7). As previously mentioned, Standard does not make use of f^A in the prior, hence the exploration behavior should be independent of the type of bias

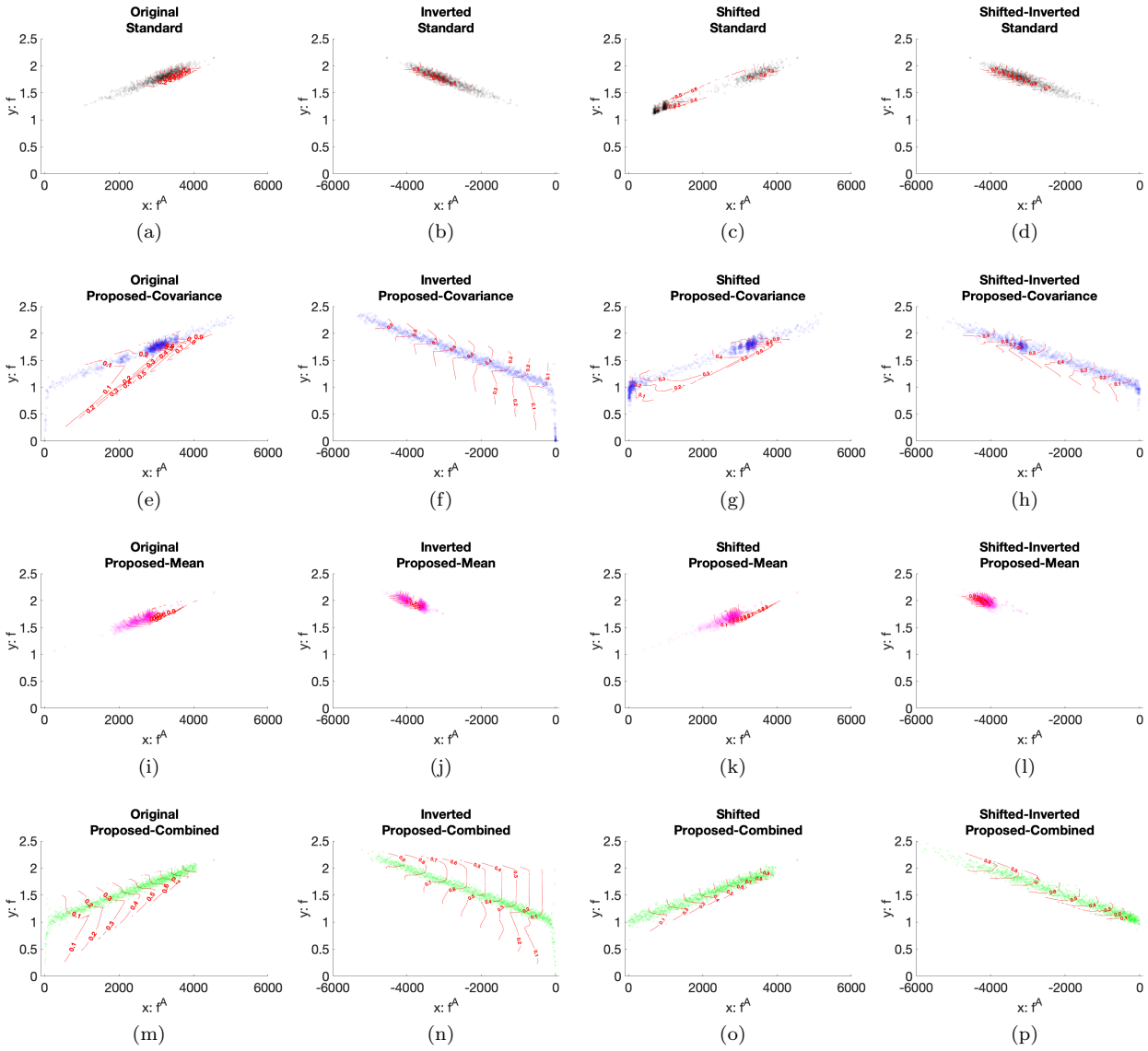


Figure 7: 2-D empirical cumulative distribution functions illustrating exploration in f^A-space when bias is introduced to f^A.

in f^A. As such the distribution of observations for Standard using the Inverted f^A and Shifted-Inverted f^A (Figures 7b and 7d) are essentially similar, and are mirror images of Figure 7a. The different distribution of observations seen in Figure 7c for the Shifted f^A can be attributed to randomness, where the BO run managed to identify a relatively good solution with f value around 1.1, leading to a cluster of observations around that region. In all cases, Standard does not explore as large a range of f^A values as compared to Proposed-Covariance or Proposed-Combined.

Moving on to the Inverted f^A (second column of Figure 7), we first note the inverse relationship between f^A and f, due to the anti-correlation between Inverted f^A and f. We also see that Proposed-Covariance and Proposed-Combined (Figures 7f and 7n respectively) explored a much larger range of f^A values, compared to Standard and Proposed-Mean (Figures 7b and 7j respectively). Similar to the case with Original f^A, the k^{f^A} covariance function encourages exploration in the f^A-space, regardless of the sign and magnitude of the correlation between f^A and f. Moreover, the maximum of the Inverted f^A (i.e. $f^A = 0$) is aligned with the minimum of f, with the maximum f^A value occurring at a unique point (i.e. $\mathbf{x} = \mathbf{0}$), hence Proposed-Covariance and Proposed-Combined were still able to efficiently find solutions with f close to 0 when exploring in regions with f^A values close to 0. In contrast, Proposed-Mean (Figure 7j) only uses Inverted f^A in its prior mean function, which is completely

inaccurate in describing the objective function. As a result of the misinformation in the prior, Proposed-Mean tends to explore regions with smaller (i.e. more negative) f^A values and greater f values, compared to when using the Original f^A. At the same time, the range of f^A values explored by Proposed-Mean (-3033 to -4659) is smaller when using the Inverted f^A (264 to 4544), compared to Original f^A.

The Shifted f^A model (third column of Figure 7) illustrates the case when the minimum of f^A no longer aligns with the global minimum of f. With Shifted f^A, the global minimum has an f^A value of 10, which is an f^A value that no longer occurs at a unique point. In other words, there are other points, which are not the global minimum, in the feasible region with f^A values of 10 as well. While the k^{f^A} covariance function in Proposed-Covariance and Proposed-Combined still ensures that they still explore a large range of f^A values, as shown in Figures 7g and 7o, this implies that exploring the f^A-space does not always lead to finding the global minimum of f. However, from Figure 7g, Proposed-Covariance seemed to have allocated more simulation budget to exploring the region with f^A values close to 0, as compared to Proposed-Combined where the observations are more uniformly distributed (Figure 7o). This could have led to the better optimization performance for Proposed-Covariance (mean of 0.665) compared to Proposed-Combined (mean of 0.838) when using Shifted f^A, as shown in Figure 6. In the case of Proposed-Mean (Figure 7k), the exploration behavior when using the Shifted f^A is similar to that when using the Original f^A – Proposed-Mean does not explore as large a range of f^A values as compared to Proposed-Covariance and Proposed-Combined. As a result, this also limits the best f value it can find to above 1.

Working with the Shifted-Inverted f^A model (last column of Figure 7) results in similar exploration behavior as when using the Inverted f^A model for the 4 different GP priors. Proposed-Covariance and Proposed-Combined (Figures 7h and 7p respectively) continue to explore a much larger range of f^A values as compared to Standard and Proposed-Mean (Figures 7d and 7l). However, as maximum of Shifted-Inverted f^A does not coincide with the global minimum of f, the global minimum has an f^A value of -10 which is an f^A value that does not occur at a unique point. This thus implies that simply exploring the f^A-space does not guarantee that the minimum of f will be found. In the case of Proposed-Mean, the inversed relationship of the Shifted-Inverted f^A and f again resulted in Proposed-Mean exploring regions with smaller (more negative) f^A values, along with a smaller range of f^A values compared to Original f^A and Shifted f^A.

In general, the experiments in this section has shown that the use of k^{f^A} as the covariance function encourages exploration in the f^A-space regardless of biases in f^A when modeling the objective function. To efficiently identify the minimum in the objective function, the optimum in f^A should ideally coincide with the minimum in the objective function, and the f^A value at the objective function minimum should occur only at a unique point in the feasible region. However, as the Shifted and Shifted-Inverted cases show, the correlation between f^A and f can still improve high-dimensional BO performance when using the k^{f^A} covariance function, compared to a general-purpose squared exponential covariance function, highlighting the robustness of the proposed method.

5 Case Study: Midtown Manhattan Traffic Signal Control

5.1 Traffic Signal Optimization Problem

In this case study, we apply our proposed method to a high-dimensional fixed time traffic signal optimization problem for the large-scale area of Midtown Manhattan (MTM) in New York City. For a review of traffic signal optimization terminology, see Osorio (2010, Appendix A, pages 119-121). In fixed time signal control, the signal plan is cyclic (i.e. periodic) with a fixed cycle time (i.e. the time required to complete one sequence of signals). The decision variables in this problem are the green splits (i.e. normalized green times) of each signal phase of each intersection in the network. Other control variables, such as the offsets, stage structure, cycle times, etc., are predetermined and kept constant. The notation used for formulation of the traffic signal optimization problem is given below:

The formulation of the problem is then given by:

$$\min_{\mathbf{x}} f(\mathbf{x}, \mathbf{z}; \mathbf{p}) = \mathbb{E}[F(\mathbf{x}, \mathbf{z}; \mathbf{p})] \tag{24}$$

$$\text{subject to } \sum_{j \in \mathcal{P}(\ell)} x_j = \frac{c_\ell - d_\ell}{c_\ell}, \ \forall \ell \in \mathcal{I} \tag{25}$$

$$\mathbf{x} \geq \mathbf{x}^{LB}. \tag{26}$$

The objective function for this case study (Eq. (24)) is the expected travel time of vehicles, as evaluated by a stochastic traffic simulator whose output is represented by the random variable F. The objective function depends

f	SO objective function (expected travel time of vehicles in the road network);
F	random variable denoting the travel time of vehicles in the road network;
x_j	green split of signal phase j (decision variable);
$\mathbf{x}$	vector of all green splits (decision vector);
$\mathbf{z}$	vector of endogenous simulation variables.
	Exogenous problem parameters:
c_ℓ	cycle time of intersection ℓ;
d_ℓ	fixed cycle time of intersection ℓ;
$\mathbf{x}^{LB}$	vector of minimal green splits;
$\mathbf{p}$	vector of exogenous simulation parameters;
$\mathcal{I}$	set of signal controlled intersection indices;
$\mathcal{P}(\ell)$	set of signal phase indices of intersection ℓ.

on a vector of exogenous parameters $\mathbf{p}$, which accounts for the road network topology and fixed lane attributes (e.g. lane length, maximum speed, grade), for instance. The endogenous simulation variables $\mathbf{z}$ represents, for example, route choice decisions, as well as link-level and network-level performance metrics like travel times, speeds, densities, delays, etc. For more details about how the objective function is computed from the stochastic traffic simulator, we refer the reader to Appendix D.

The feasible region of $\mathbf{x}$ is defined by the constraints (25) and (26). Eq. (25) represents the cycle time constraint, which states that for a given intersection, the sum of green splits must be equal to the proportion of cycle time that can be optimized (i.e. not fixed). In practice, it is common to assign fixed amounts of time to certain traffic phases, typically for safety considerations and to comply with local transportation regulations. For instance, a fixed amount of the cycle time is typically assigned to all-red periods, where the signal is red for all traffic movements between some signal phases for safety reasons. Eq. (26) represents the lower bounds of the green splits, where the minimum green splits are typically determined by the local transportation authorities based on safety considerations.

For the traffic signal optimization problem, the underlying road network of the MTM area is known beforehand. Hence, this problem-specific prior information can be used to inform the GP prior through the prior mean function and/or the covariance function. To do this, we model the road network using a finite capacity queueing network model (Osorio and Chong 2015, Eq. 6) (see Appendix C for more details).

The MTM area being simulated in this case study is demarcated by a rectangle in the map shown in Figure 8. In this problem, we simulate traffic from 3pm - 6pm, and optimize the signal plans for the peak hour of 5pm - 6pm. We control a total of 97 signalized intersections, with 259 green splits (i.e. decision variables). Due to the linear cycle time equality constraint (Eq. (25)), this can be considered a 162-D problem (i.e. $259 - 97 = 162$), making it high-dimensional in the field of BO.

The MTM simulation model is implemented using the Aimsun software (TSS-Transport Simulation Systems 2015). It consists of a total of 698 roads, 2756 lanes and 444 intersection. The complete network topology of the simulation model is illustrated in Figure 9. During the simulated interval of 5pm - 6pm, the expected demand over 29,000 trips per hour, distributed across more than 3500 origin-destination pairs. In this simulation model, the minimum green time that can be assigned to each signal phase was 6 seconds. Hence, the corresponding elements in the vector of minimum green splits $\mathbf{x}^{LB}$ (Eq. (26)) are the ratio of 6 seconds to the cycle time of the intersection that the green split belongs to.

5.2 Experimental Set-Up

In this case study, BO was implemented as described in Algorithm 1, with 10 initial observations provided to fit the GP posteriors. The computational budget was taken as 55 iterations (i.e. 45 optimization iterations). As with the Griewank function experiments in Section 4, the objective function estimate was obtained by taking the mean of 4 simulations, with 2 additional simulations of the best solution at the end of each optimization iteration. This leads to a total of 310 simulation runs, 40 of which are spent on the 10 initial observations.

For each GP prior, we consider 3 distinct initial sets, each containing the 10 initial observations used to fit the GP posteriors at the start of the BO run. The 10 initial observations in each set were obtained by sampling within the feasible region (as defined by Eq. (25) and (26)) uniformly at random using the code of Stafford (2006), and taking the mean of 4 simulation runs for each point. In addition, one of the uniformly random points in Initial Set 3 was replaced by an existing signal plan, which was previously used by the New York City Department of

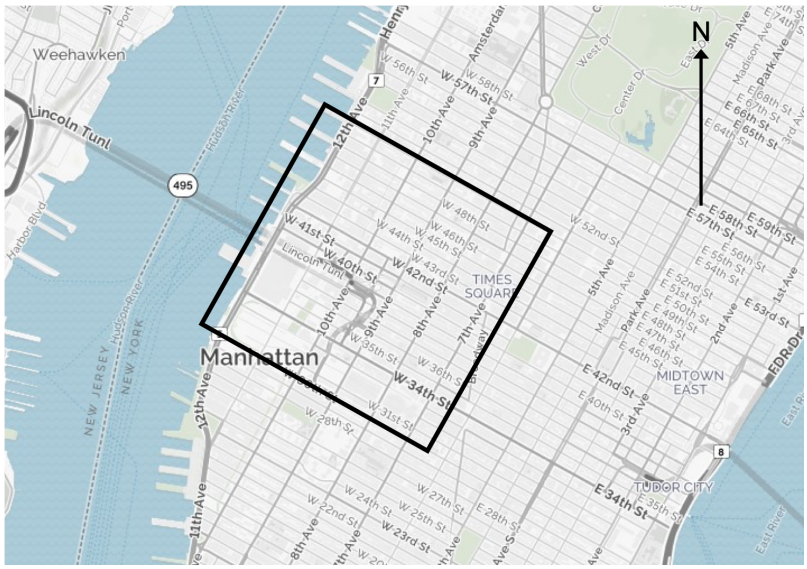


Figure 8: Map of Midtown Manhattan with the simulated area demarcated by a rectangle (MapQuest.com, Inc 2018).

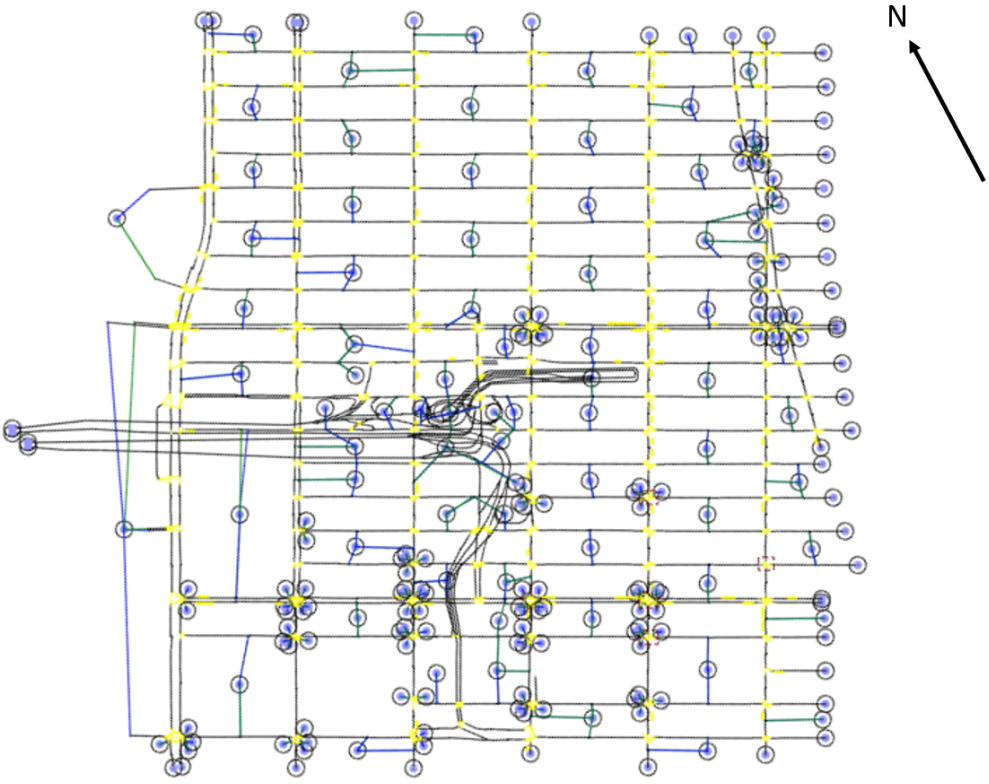
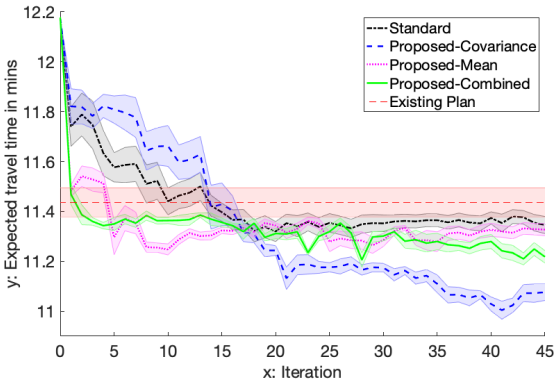
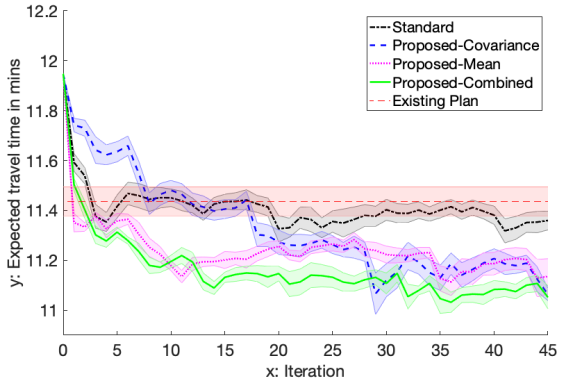


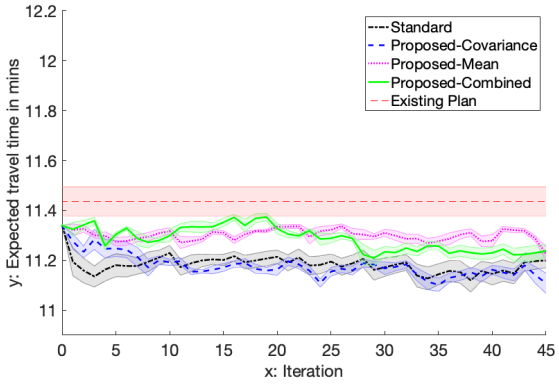
Figure 9: Midtown Manhattan model in Aimsun.



(a) Initial Set 1



(b) Initial Set 2



(c) Initial Set 3

Figure 10: Mean performance at a given iteration achieved by the different GP priors.

Transportation (NYCDOT) for the MTM area. This existing signal plan is known to perform well, and hence was included in Initial Set 3 to investigate how BO will be affected when the initial set contains a solution with good performance. We also use the existing plan as a benchmark for the performance of the signal plans proposed by BO with the 4 different GP priors. For more implementation details, we refer the reader to Appendix D.

5.3 Results

We compare the optimization performance of the 4 different GP priors in Figure 10. Each plot in Figure 10 shows a comparison of the optimization performance for a different initial set. The x-axis represents the optimization iteration (i.e. does not count the 10 initial observations), while the y-axis shows the expected travel time (objective function value). In each plot, each line depicts the mean of the best solution at a given iteration for 5 BO runs, with the shaded regions showing the values ± 1 standard error away from the mean. The mean travel time and its standard error based on 50 simulations using the existing plan is also indicated in each plot by the thin red dashed line and the red shaded region. This allows for comparison with the performance of BO with the 4 different GP priors.

We further tested if the differences in optimization performance of the different GP priors were statistically significant, by performing a one-sided paired t-test based on the mean performance of the 5 BO runs for each initial set. The results of the t-test are given in Table 2. For each row in the table, the null hypothesis assumes that the first GP prior (in Column 1) obtained a mean performance that is worse than or equal to the mean performance of the second GP prior (in Column 1). In contrast, the alternative hypothesis states that the first GP prior obtained a mean performance that is better than the second GP prior. For instance, the last row of Table 2 tests the alternative hypothesis that Proposed-Covariance obtained a better mean performance than Proposed-Combined. Both the t-statistics and p-values are shown for each test and for all 3 initial sets. Each t-test is considered at the 10% level of significance. With 4 degrees of freedom, the corresponding critical value of

Table 2: One-sided paired t-test results

Test	Initial Set 1		Initial Set 2		Initial Set 3	
	t-statistic	p-value	t-statistic	p-value	t-statistic	p-value
Proposed-Covariance vs. Standard	**-2.546**	**0.0318**	**-3.625**	**0.0111**	-0.851	0.221
Proposed-Mean vs. Standard	-0.369	0.365	**-1.956**	**0.0611**	0.577	0.702
Proposed-Combined vs. Standard	**-1.675**	**0.0846**	**-3.024**	**0.0195**	0.521	0.685
Proposed-Covariance vs. Proposed-Mean	**-2.943**	**0.0211**	-0.448	0.339	-1.020	0.183
Proposed-Combined vs. Proposed-Mean	**-2.687**	**0.0274**	-0.690	0.264	0.176	0.565
Proposed-Covariance vs. Proposed-Combined	**-1.738**	**0.0786**	0.095	0.536	**-1.759**	**0.0767**

the t-statistic is -1.533. The t-tests with t-statistics smaller than the critical value, which are displayed in bold, have their null hypotheses rejected.

We first consider Initial Set 1 (Figure 10a). Of all the GP priors, Proposed-Covariance (blue dashed line) was able to identify the best point on average after expending the computational budget. In fact, Proposed-Covariance was statistically significantly better than all of the other 3 GP priors as seen in Table 2. It identified solutions that reduced the expected travel time by 3.2% on average compared to the existing plan. Proposed-Combined (green solid line) obtained solutions with the next best performance, and is statistically significantly better than Proposed-Mean and Standard. This is followed by Proposed-Mean (magenta dotted line) and then Standard (black dash-dot line). However, the mean performance of Proposed-Mean is not statistically better than that of Standard. While Proposed-Covariance was the best GP prior based on results at the end of the BO runs, Figure 10a also shows that it is the slowest in finding solutions better than the existing plan. Proposed-Mean and Proposed-Combined were actually able to identify solutions better than the existing plan much more quickly (5 and 2 iterations respectively) than both Proposed-Covariance (15 iterations). Standard was unable to identify solutions that are significantly better than the existing plan. This suggests that f^A in the prior mean function is being exploited to find good solutions quickly.

For Initial Set 2 (Figure 10b), Proposed-Covariance, Proposed-Mean and Proposed-Combined were all able to outperform Standard at the end of the BO runs. The differences in performance compared to Standard were statistically significant at the 10% level of significance as seen in Table 2, highlighting the usefulness of incorporating problem-specific information in the GP prior. Proposed-Covariance also registered a 3.3% reduction in expected travel time on average relative to the existing plan. However, there is no significant difference between Proposed-Covariance, Proposed-Mean and Proposed-Combined. Furthermore, as with Initial Set 1, Proposed-Mean and Proposed-Combined were again able to quickly identify solutions better than the existing plan (1 and 3 iterations respectively), compared to Proposed-Covariance (18 iterations). Standard was again unable to identify solutions that are significantly better than the existing plan.

Initial Set 3 (Figure 10c) contains the existing plan as one of the initial points. It is the point with the best performance in the initial set. The reason the starting mean performance of all the GP priors is better than that of the existing plan (i.e. the red dashed line) is due to chance. With a signal plan with good performance already in the initial set, Standard was able to perform well in this case, as it simply explored around the vicinity of the existing plan in the feasible region to find better solutions (more evidence of this behavior is shown later in Figures 11 and 12). As a result, Standard was able to find solutions by the end of the BO runs which was statistically on par with those obtained by Proposed-Covariance, Proposed-Mean and Proposed-Combined, as shown by the results of the t-tests in Table 2. Hence, this shows that if the initial set contains an observation with good performance, the importance of access to problem-specific prior information may be diminished. In this case, Proposed-Covariance reduced the expected travel time by 2.9% on average, compared to the existing plan.

In general, Figure 10 showed that there is value in using the k^{f^A} covariance function, as it is able to consistently identify solutions with better or equal performance than GP priors using the squared exponential covariance function. In addition, the results also show that placing f^A in the prior mean function allows BO to quickly

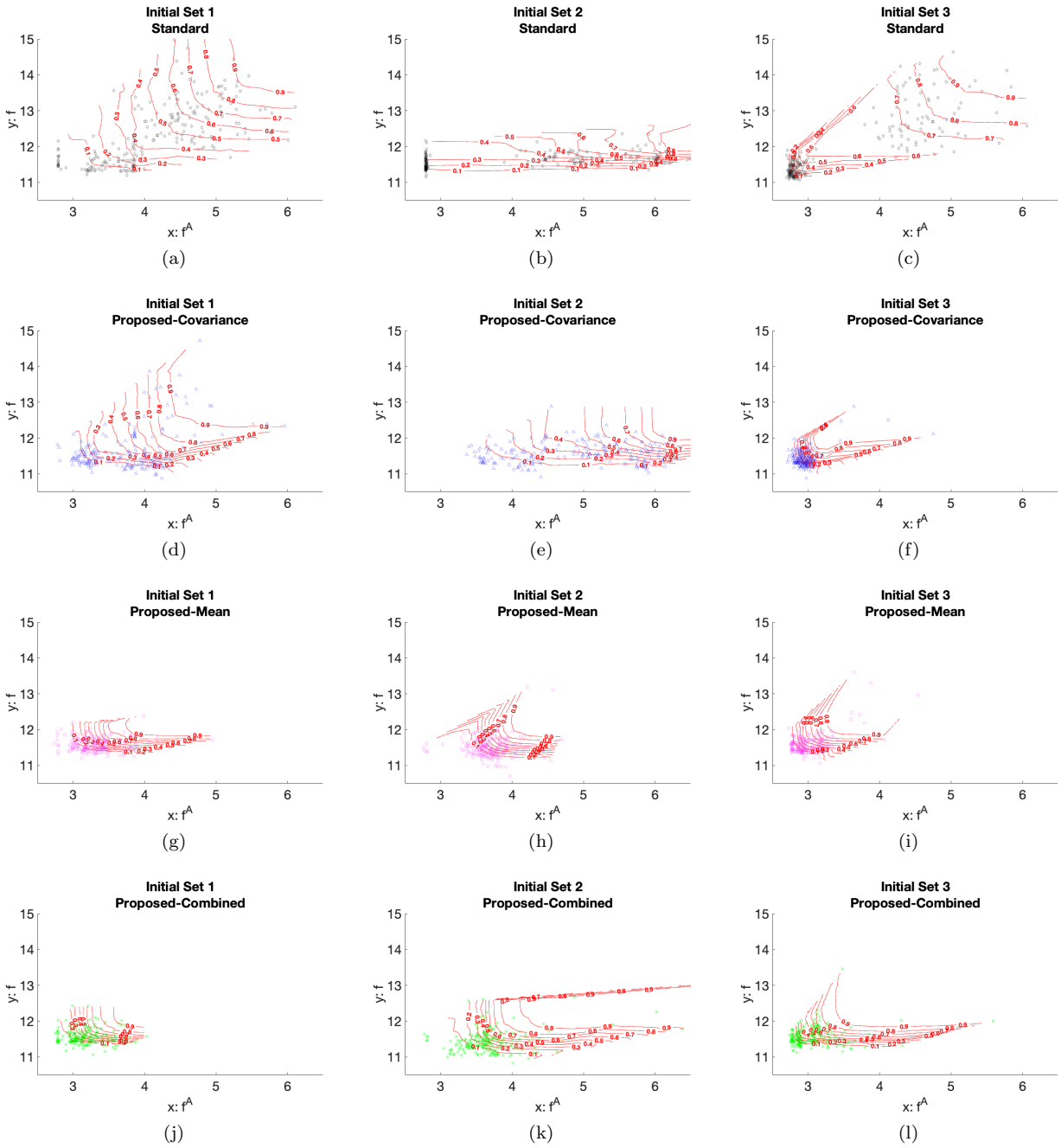


Figure 11: 2-D empirical cumulative distribution functions illustrating exploration in f^A-space.

find good solutions within a few iterations. However, as Figures 10a and 10c show, placing f^A in the prior mean function could limit the best solution found in the long run.

Figure 11 shows the f^A values and the corresponding f values for all evaluated points. It provides insights into the amount of exploration done in the f^A-space by each GP prior. In each plot of Figure 11, the f^A and f values of every simulated point is illustrated, along with the 2-D ecdf to provide a visualization of the distribution of f^A and f values explored. The 2-D ecdf here is computed based on Eq. (22). Each row of plots represent one of the 4 different GP priors being tested, while each column shows the results for a different initial set.

Focusing first on Initial Set 1 (first column of Figure 11), we see that Standard explored quite a large range of f^A values (2.79 - 6.11) compared to Proposed-Mean and Proposed-Combined. However, the f values that it found were not as good, with the best f value being 11.22. In contrast, Proposed-Covariance also explored a large range

of f^A values (2.80 - 5.96), but the solutions it identified included more with smaller f values, particularly around the region with f^A values between 3 and 4.5. This shows that the minimum of f^A may not necessarily coincide with the minimum of f, but the k^{f^A} covariance function can still help to identify solutions with smaller f values. Proposed-Mean and Proposed-Combined explored a smaller range of f^A values than Standard and Proposed-Covariance. In fact, most of the evaluated points had f^A values less than 3.5, indicating that the Proposed-Mean and Proposed-Combined were exploiting the use of f^A in the prior mean function with the assumption that f^A and f are strongly correlated.

Moving on to Initial Set 2 (second column of Figure 11), Standard seemed to have found a point with good performance (with f^A value of 2.79) and kept exploring around the vicinity, which explains the cluster of evaluated points with f^A values of about 2.8. Further evidence of this behavior is shown later in Figure 12. Proposed-Covariance again explored a large range of f^A values (3.36 - 7.85). Furthermore, it identified many points with large f^A but small f values, providing further evidence that the k^{f^A} covariance function is very helpful in the search for small f values. From Figure 11h, Proposed-Mean mainly focused on finding points with smaller f^A again. On the other hand, Proposed-Combined explored a larger range of f^A values compared to Proposed-Mean this time. However, the main focus was still on the regions with smaller f^A values.

For Initial Set 3 (third column of Figure 11), which contains the existing plan as one of the initial observations, Standard spent most of the computational budget exploring points close to the existing plan with the hope of finding better solutions. However, it also explored other regions with larger f^A values, but the points explored with large f^A values also had large f values. Proposed-Covariance, Proposed-Mean and Proposed-Combined similarly explored only in the vicinity of the existing plan (see Figure 12 for more evidence). As a result, their exploration was limited to just a small range of f^A values. This could be the result of having an observation with good performance in the initial set, which could affect the hyperparameter values obtained through maximum likelihood estimation, such that the effect of f^A in the covariance function on exploration in the f^A-space is diminished. Hence, this shows the sensitivity of BO to the initial set.

To visualize the distribution of the observations in the feasible region (i.e. the space of feasible signal plans), we made use of multidimensional scaling (MDS, see e.g. Cox and Cox (2008)) to project the points onto a 2-D space. Simply put, MDS is a projection technique that retains the original pairwise distances between points as much as possible. Here, we used the Matlab function *cmdscale* to compute the projection based on the pairwise Euclidean distances between the observations. Figure 12 illustrates the positions of the observations relative to one another in this 2-D space. The contour lines in the plots indicate the objective function value at a given point. Each column represents a different initial set, while each row shows a different GP prior. Note that the observations from all the GP priors and all 3 initial sets were used to define the MDS 2-D space, hence the 2-D space and contour line layout in each plot are the same. This allows for direct comparison across the plots for different initial sets. Furthermore, we note that the relative errors of representing the observations in just 2 dimensions are naturally large (92%, 81% and 82% for Initial Sets 1,2 and 3 respectively). Hence, we do not use the MDS plots to draw independent conclusions, but merely use it to substantiate other claims.

Comparing across the 3 columns in Figure 12, we see that the regions explored in the MDS 2-D projected space can be be quite different for each initial set. This supports the notion that BO is sensitive to the initial set used. The first and second columns of Figure 12 even show that the Proposed-Covariance explores significantly different regions of the 2-D space compared to Proposed-Mean and Proposed-Combined, highlighting that the different GP priors react differently to the initial sets as expected. The first column of Figure 12 also supports the explanation for the cluster of observations by Standard with f^A values around 2.8 as seen in Figure 11b – there is a cluster of observations by Standard around $(-0.25, -0.3)$ in Figure 12b, suggesting that Standard got stuck in a local minimum, and hence explored only in the vicinity. For Initial Set 3, we previously observed that all 4 GP priors explored only an area with small f^A values, suggesting that they were exploring only in the vicinity of the existing plan. The third column of Figure 12 reinforces this claim, as it shows most of the observations surrounding the initial observation at $(-0.30, -0.07)$ which corresponds to the existing plan.

6 Conclusion

In this paper, we showed how problem-specific information in the form of analytical transportation models can be used for exploration, in addition to exploitation, to tackle high-dimensional BO efficiently. Exploitation of the problem-specific information can be achieved by using the analytical transportation model in the prior mean function of the GP, allowing the algorithm to identify good solutions more quickly. This is true when compared to using a non-informative prior mean function. On top of this, we showed that incorporating an analytical

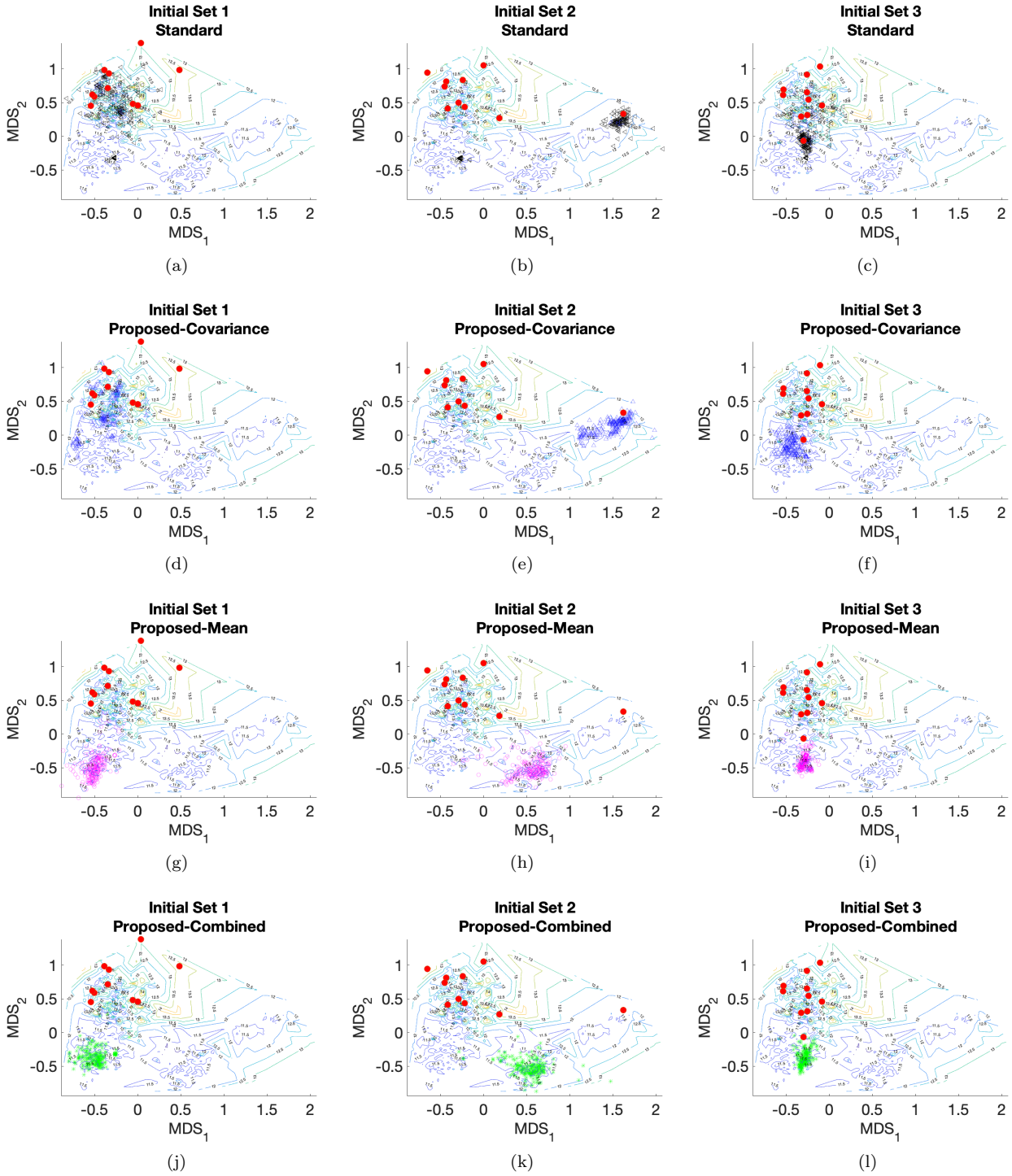


Figure 12: Multidimensional scaling 2-D projections of all observations in the feasible region.

transportation model in the GP covariance function can encourage exploration in areas of the feasible region with different analytical transportation model values from points which have already been evaluated. This allows better solutions to be found even if the analytical transportation model is not an accurate representation of the objective function. In fact, having the analytical transportation model in the GP covariance function resulted in better optimization performance than not providing any problem-specific information in the covariance function. This is true regardless of whether the analytical transportation model was being exploited in the prior mean

function.

At the same time, incorporating the problem-specific prior information in the covariance function (as opposed to the prior mean function only) helped the optimization be more robust to inaccuracies in the analytical transportation model. This opens up the possibility of using models which are less accurate but are still (anti-)correlated to the objective function. For instance, the analytical transportation model does not even have to be modeling the objective function itself, but instead it could be modeling another quantity which is (anti-)correlated to the objective function (e.g. network throughput is anticorrelated with the expected travel time).

Also, while the proposed analytical model-based covariance function was demonstrated on a transportation case study problem, we would like to emphasize the generality of the method, as it is not limited to just transportation optimization or SO problems. The proposed covariance function can be applied to any problem type in general where the objective function is expensive to evaluate, and where there is problem-specific prior information available in the form of an analytical model.

Despite the effectiveness of the proposed covariance function for high-dimensional BO, we note that it can be difficult to find a good set of hyperparameter values at times. This is especially so with limited observations in high-dimensional settings. Furthermore, the initial set used to fit the GP posterior at the start of the BO run can affect the estimated hyperparameter values, and has a strong impact on the eventual outcome. As such, more studies can be done to identify ways to better estimate hyperparameter values that can allow for more efficient optimization to take place.

We also found that BO was sensitive to the set of initial points used to fit the GP at the start of the optimization process. This was regardless of whether problem-specific prior information was incorporated in the GP prior mean function and/or the covariance function or not. Hence, work could be done to better sample for initial points while keeping computational costs in check, especially for high-dimensional BO problems.

Another area that can be studied further is the choice of functional forms for the analytical transportation model-based covariance function. For the purpose of this paper, we focused on using a squared exponential form of the difference between analytical model values of two points in the proposed covariance function. However, other functional forms might provide a better fit of the GP posterior to the objective function and hence more efficient optimization.

As part of our ongoing research, we are investigating possible ways to extend the use of analytical transportation model-based covariance functions and BO to high-dimensional dynamic problems. In transportation, it is common to encounter dynamic optimization problems, where the objective function and decision variables are time-dependent, as practitioners try to account for the spatiotemporal dynamics of travelers in the system. To tackle dynamic problems, the GP posterior model would have to account for time variations of the objective function. Due to the difficulties of modeling a high-dimensional dynamic problem accurately, we believe that BO, together with the use of problem-specific prior information, can be a good option for solving such simulation-based optimization problems.

Acknowledgement

The authors would like to thank the New York City Department of Transportation (NYCDOT), in particular Jingqin Gao, Mohamad Talas and Michael Marsico, for providing us with the simulation model for Midtown Manhattan. Timothy Tay would also like to thank the Agency for Science, Technology and Research (A*STAR) Singapore for funding his work.

A List of Notation Used

f	objective function;
f^A	approximate analytical model of the objective function;
F	random variable denoting the stochastic output of a simulation run;
$\mathbf{x}$	vector of decision variables;
$\mathbf{z}$	vector of endogenous simulation variables;
$\mathbf{p}$	vector of deterministic exogenous parameters;
χ	feasible region;
D	number of dimensions of the feasible region;
ϵ	i.i.d. Gaussian noise;
	GP notation:
t	number of observations;
t_0	number of initial observations;
T	budget for number of objective function evaluations;
m	prior mean function of a GP;
$\mathbf{m}$	vector of prior mean function values;
β	prior mean constant;
α	analytical model scaling constant;
k	covariance function of a GP;
$\mathbf{k}$	vector of covariance function values;
K	covariance matrix;
k_{SE}	squared exponential covariance function;
k_{f^A}	analytical model-based covariance function;
μ	posterior (predictive) mean function of a GP;
σ^2	posterior (predictive) variance of a GP;
σ_0^2	covariance amplitude;
ℓ	covariance characteristic length-scale;
ℓ_{f^A}	analytical model length-scale;
τ^2	variance of Gaussian noise;
I	identity matrix;
	Signal control problem exogenous parameters:
c_ℓ	cycle time of intersection ℓ;
d_ℓ	fixed cycle time of intersection ℓ;
$\mathbf{x}^{LB}$	vector of minimal green splits;
$\mathcal{I}$	set of signal controlled intersection indices;
$\mathcal{P}(\ell)$	set of signal phase indices of intersection ℓ.

B Implementation Details for Validation with 100-D Griewank Function

To identify the fixed set of hyperparameter values for use in Section 4.4, we executed a grid search to find the best set of hyperparameter values. The hyperparameter values tested can be found in Table 3. The first column of the table identifies the GP prior, while the second and third columns shows the values tested for the mean function hyperparameter (β for Standard and Proposed-Covariance, and α for Proposed-Mean and Proposed-Combined). The fourth and fifth columns show the values tested for the covariance amplitude and characteristic length-scale respectively. The last column represents the analytical model length-scale values, which is applicable only to the GP priors with k_{f^A} as the covariance function (i.e. Proposed-Covariance and Proposed-Combined). The values in bold represent the best set of fixed hyperparameter values for each GP prior. This same set of fixed hyperparameter values were used in Section 4.5 as well.

The tested values shown in Table 3 were chosen to represent different orders of magnitude, with the goal of

Table 3: Fixed Hyperparameter Values Tested

GP Prior	$\boldsymbol{\alpha}$	$\boldsymbol{\beta}$	$\boldsymbol{\sigma_0^2}$	$\boldsymbol{\ell}$	$\boldsymbol{\ell_{fA}}$
Standard	–	{0, 0.1, **1**}	{0.5, **5**}	{10, 100, **1000**}	–
Proposed-Covariance	–	{0, **0.1**, 1}	{**0.5**, 5}	{10, **100**, 1000}	{0.1, 1, **10**}
Proposed-Mean	{**0.001**, 0.01, 0.1}	–	{**0.5**, 5}	{**10**, 100, 1000}	–
Proposed-Combined	{**0.001**, 0.01, 0.1}	–	{0.5, **5**}	{10, **100**, 1000}	{0.1, 1, **10**}

finding hyperparameter values of the right order of magnitude. As such, the best set of hyperparameter values found may not be the optimal set, as no further fine-tuning of the hyperparameter values was done. For each GP prior, the best set of hyperparameters was chosen by comparing the mean of the best objective function estimate of 3 optimization runs (as in Figure 3) for every set. The set that required the fewest number of iterations to reach within 0.05 of the global minimum value of 0 is taken as the best.

C Queueing Network Model

In Section 5, we model the road network using a finite capacity queueing network model (Osorio and Chong 2015, Eq. 6), which accounts for vehicular spillbacks (when downstream lanes are full, thus blocking traffic flow from upstream lanes) through the queueing theoretic concept of blocking. In this queueing network model, each lane in the network is represented by a finite space capacity $M/M/1/k$ queue, where k denotes the finite space capacity of that lane. The green splits of the traffic signal optimization problem are then related to the queueing network model, through their effect on the service rates of the queues at the corresponding intersections (see Eq. 18 of Osorio and Chong (2015)). Based on the queueing network model, the analytical approximation f^A of the objective function (i.e. expected travel time) can be derived (Osorio and Chong 2015, Eq. 11).

In this Appendix, we define the the queueing network model and show how it can provide an analytical approximation of the objective function. For more details about the queueing network model and the derivation of the analytical approximation of the objective function, we refer the reader to Osorio and Chong (2015, Sections 3, 4 and Appendix A). We use the notation of Osorio and Chong (2015) in this Appendix, where the index i refers to a given queue:

γ_i	external arrival rate;
λ_i^{eff}	effective arrival rate;
μ_i	service rate;
ρ_i^{eff}	effective traffic intensity;
k_i	upper bound of the queue length;
N_i	total number of vehicles in queue i;
$P(N_i = k_i)$	probability of queue i being full;
p_{ij}	transition probability from queue i to queue j;
$\mathcal{D}_i$	set of downstream queues of queue i.

The queuing network model is given by the following system of equations:

$$\lambda_i^{\text{eff}} = \gamma_i\big(1 - P(N_i = k_i)\big) + \sum_j p_{ji}\lambda_j \tag{27a}$$

$$\rho_i^{\text{eff}} = \frac{\lambda_i^{\text{eff}}}{\mu_i} + \Big(\sum_{j\in\mathcal{D}_i} p_{ij}P(N_j = k_j)\Big)\Big(\sum_{j\in\mathcal{D}_i}\rho_j^{\text{eff}}\Big) \tag{27b}$$

$$P(N_i = k_i) = \frac{1-\rho_i^{\text{eff}}}{1-(\rho_i^{\text{eff}})^{k_i+1}}(\rho_i^{\text{eff}})^{k_i}. \tag{27c}$$

Eq. (27a) is a flow conservation equation relating the demand rate of queue i (left hand side of the equality) to the sum of the demand rate of vehicle trips that start in queue i (first term of the right hand side) and of the demand rate of vehicle trips that arise from upstream queues (second term of the right hand side). More specifically, the demand rate for trips that start in queue i is represented by γ_i, and the term $(1 - P(N_i = k_i))$ enforces that trips can only start in queue i if it is not full. Eq. (27b) defines the traffic intensity. The first term of the right hand

side is the traffic intensity when the queue is not full (i.e., when there is no spillback). The second term accounts for the impact in queue i due to spillbacks from its downstream queues. Eq. (27c) gives the expression of the spillback probability (i.e., the blocking probability) as defined for an $M/M/1/k_i$ queue.

The endogenous variables of the above system of equations are related to the decision vector (the green split vector $\mathbf{x}$) by the following linear equations:

$$\mu_i = s\left(e_i + \sum_{j \in \mathcal{P}_2(i)} x_j\right) \quad \forall i \in \mathcal{L}, \tag{28}$$

where s denotes an exogenous scalar that represents the saturation flow rate (i.e., maximum queue discharge rate), and e_i is an exogenous parameter that represents the ratio of fixed green time to cycle time for signalized queue i. Eq. (28) states that the service rate of a signalized queue i is given by the saturation rate scaled by the proportion of cycle time the queue has a green phase. This proportion is given by the term in parenthesis, which depends on the fixed (i.e., not optimized) time (term e_i) and the variable time (summation term, which represents the sum of the green splits of the signal phases of queue i).

The expected travel time (i.e. the objective function) can be approximated by the queueing network model. This is done by applying Little's law (Little 1961) to the entire network:

$$f^A(\mathbf{x}) = \frac{\sum_i \mathbb{E}[N_i]}{\sum_i \gamma_i (1 - P(N_i = k_i))} \tag{29}$$

where $\mathbb{E}[N_i]$ represents the expected number of vehicles in lane i. The numerator of Eq. (29) represents the expected number of vehicles in the entire network, while the denominator represents the expected arrival rate to the network. The ratio of these two quantities provide the expected travel time of the network according to Little's law.

The expected number of vehicles in lane i can be computed approximately as such:

$$\mathbb{E}[N_i] = \rho_i^{\text{eff}} \left(\frac{1}{1 - \rho_i^{\text{eff}}} - \frac{(k_i + 1)(\rho_i^{\text{eff}})^{k_i}}{1 - (\rho_i^{\text{eff}})^{k_i + 1}}\right) \tag{30}$$

The derivation of Eq. (30) can be found in Appendix A of Osorio and Chong (2015).

D Case Study Implementation Details

The case study objective function (expected travel time) in Eq. (24) is evaluated by running simulations of the MTM model in Aimsun. In each simulation run, the travel time of each vehicle is taken as the total amount of time that vehicle spent in the network. For vehicles which manage to complete their trips between 5pm - 6pm, the total amount of time spent in the network is the difference between the time they enter the network and the time they exit the network. For vehicles which do not complete their trips by the end of the simulation (i.e. 6:00pm), their total time spent in the network are counted as the difference between the time they entered the network and 6:00pm. The reason for counting the time spent in the network by vehicles that have not completed their trips is to ensure that gridlocks in the network are penalized. The expected travel time is then taken to be the mean amount of time that each vehicle spends in the network.

Unlike the optimization of the 100-D Griewank function in Section 4.4, we decided to optimize the hyperparameters at every iteration, due to the difficulties involved in identifying a good set of fixed hyperparameters when working with a computational-expensive simulator. To prevent the hyperparameter values from diverging, we optimize the hyperparameters by performing maximum likelihood estimation using a fixed, pre-selected set of hyperparameter values (Table 4) as the initial point at every 5^{th} iteration (i.e. 5^{th}, 10^{th}, 15^{th}, etc.). For all other iterations, the maximum likelihood estimation initial point was taken as the optimized hyperparameter values from the previous iteration.

References

Ankenman, Bruce, Barry L. Nelson, Jeremy Staum. 2010. Stochastic Kriging for simulation metamodeling. *Operations Research* **58**(2) 371–382.

Table 4: Pre-selected hyperparameter values

GP Prior	α	β	σ_0^2	ℓ	ℓ_{fA}
Standard	–	10	15	0.5	–
Proposed-Covariance	–	10	15	0.5	100
Proposed-Mean	3	–	15	0.5	–
Proposed-Combined	3	–	15	0.5	100

Barceló, Jaume, et al. 2010. *Fundamentals of traffic simulation*, vol. 145. Springer.

Bergstra, James, Daniel Yamins, David Cox. 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. *International Conference on Machine Learning*. 115–123.

Bergstra, James S, Rémi Bardenet, Yoshua Bengio, Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems*. 2546–2554.

Brochu, Eric, Vlad M Cora, Nando De Freitas. 2010. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599* .

Chen, Bo, Rui Castro, Andreas Krause. 2012. Joint optimization and variable selection of high-dimensional Gaussian processes. *arXiv preprint arXiv:1206.6396* .

Chen, Xiqun, Lei Zhang, Xiang He, Chenfeng Xiong, Zhiheng Li. 2014. Surrogate-based optimization of expensive-to-evaluate objective for optimal highway toll charges in transportation network. *Computer-Aided Civil and Infrastructure Engineering* **29**(5) 359–381.

Chong, Linsen, Carolina Osorio. 2018. A simulation-based optimization algorithm for dynamic large-scale urban transportation problems. *Transportation Science* **52**(3) 637–656.

Cox, Michael AA, Trevor F Cox. 2008. Multidimensional scaling. *Handbook of data visualization*. Springer, 315–347.

Frazier, Peter, Warren Powell, Savas Dayanik. 2009. The knowledge-gradient policy for correlated normal beliefs. *INFORMS journal on Computing* **21**(4) 599–613.

Frazier, Peter I. 2018. Bayesian optimization. *Recent Advances in Optimization and Modeling of Contemporary Problems*. INFORMS, 255–278.

Frean, Marcus, Phillip Boyle. 2008. Using Gaussian processes to optimize expensive functions. *Australasian Joint Conference on Artificial Intelligence*. Springer, 258–267.

Greenhall, Adam. 2016. Experimentation in a ridesharing marketplace: Simulating a ridesharing marketplace. URL `https://eng.lyft.com/https-medium-com-adamgreenhall-simulating-a-ridesharing-marketplace-36007a8a31f2`. Accessed: 2019-01-29.

Griewank, Andreas O. 1981. Generalized descent for global optimization. *Journal of Optimization Theory and Applications* **34**(1) 11–39.

Hennig, Philipp, Christian J Schuler. 2012. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research* **13**(6).

Hernández-Lobato, José Miguel, Matthew W Hoffman, Zoubin Ghahramani. 2014. Predictive entropy search for efficient global optimization of black-box functions. *Advances in Neural Information Processing Systems*. 918–926.

Hutter, Frank. 2009. Automated configuration of algorithms for solving hard computational problems. Ph.D. thesis, University of British Columbia.

Hutter, Frank, Holger H Hoos, Kevin Leyton-Brown. 2011. Sequential model-based optimization for general algorithm configuration. *International Conference on Learning and Intelligent Optimization*. Springer, 507–523.

Jin, Junchen, Xiaoliang Ma, Iisakki Kosonen. 2017. A stochastic optimization framework for road traffic controls based on evolutionary algorithms and traffic simulation. *Advances in Engineering Software* **114** 348–360.

Jones, Donald R, Cary D Perttunen, Bruce E Stuckman. 1993. Lipschitzian optimization without the Lipschitz constant. *Journal of Optimization Theory and Applications* **79**(1) 157–181.

Jones, Donald R, Matthias Schonlau, William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* **13**(4) 455–492.

Kandasamy, Kirthevasan, Gautam Dasarathy, Junier B Oliva, Jeff Schneider, Barnabás Póczos. 2016. Gaussian process bandit optimisation with multi-fidelity evaluations. *Advances in Neural Information Processing Systems*. 992–1000.

Kandasamy, Kirthevasan, Jeff Schneider, Barnabás Póczos. 2015. High dimensional Bayesian optimisation and bandits via additive models. *International Conference on Machine Learning*. 295–304.

Kleijnen, Jack PC. 2017. Regression and Kriging metamodels with their experimental designs in simulation: a review. *European Journal of Operational Research* **256**(1) 1–16.

Kushner, Harold J. 1964. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering* **86**(1) 97–106.

Li, Cheng, Sunil Gupta, Santu Rana, Vu Nguyen, Svetha Venkatesh, Alistair Shilton. 2017. High dimensional Bayesian optimization using dropout. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 2096–2102.

Li, Chun-Liang, Kirthevasan Kandasamy, Barnabás Póczos, Jeff Schneider. 2016. High dimensional Bayesian optimization via restricted projection pursuit models. *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. 884–892.

Little, John DC. 1961. A proof for the queuing formula: L= λ w. *Operations research* **9**(3) 383–387.

Liu, Haitao, Jianfei Cai, Yew-Soon Ong. 2018. Remarks on multi-output Gaussian process regression. *Knowledge-Based Systems* **144** 102–121.

Lu, Lu, Yan Xu, Constantinos Antoniou, Moshe Ben-Akiva. 2015. An enhanced SPSA algorithm for the calibration of dynamic traffic assignment models. *Transportation Research Part C: Emerging Technologies* **51** 149–166.

MapQuest.com, Inc. 2018. New York City, NY, scale undetermined; generated by Timothy Tay using "MapQuest.com, Inc". URL `http://www.mapquest.com`. Accessed: 2018-07-01.

Mockus, Jonas. 1994. Application of Bayesian approach to numerical methods of global and stochastic optimization. *Journal of Global Optimization* **4**(4) 347–365.

Mockus, Jonas, Vytautas Tiesis, Antanas Zilinskas. 1978. The application of Bayesian methods for seeking the extremum. *Towards Global Optimization* **2**(117-129) 2.

Moriconi, Riccardo, KS Sesh Kumar, Marc Peter Deisenroth. 2020. High-dimensional Bayesian optimization with projections using quantile Gaussian processes. *Optimization Letters* **14**(1) 51–64.

Munteanu, Alexander, Amin Nayebi, Matthias Poloczek. 2019. A framework for Bayesian optimization in embedded subspaces. *International Conference on Machine Learning*. 4752–4761.

Osorio, Carolina. 2010. Mitigating network congestion: analytical models, optimization methods and their applications. Ph.D. thesis, École Polytechnique Fédérale de Lausanne.

Osorio, Carolina, B Atastoy. 2017. Efficient simulation-based toll optimization for large-scale networks. *Technical Report*. Massachusetts Institute of Technology.

Osorio, Carolina, Michel Bierlaire. 2009. An analytic finite capacity queueing network model capturing the propagation of congestion and blocking. *European Journal of Operational Research* **196**(3) 996–1007.

Osorio, Carolina, Michel Bierlaire. 2013. A simulation-based optimization framework for urban transportation problems. *Operations Research* **61**(6) 1333–1345.

Osorio, Carolina, Linsen Chong. 2015. A computationally efficient simulation-based optimization algorithm for large-scale urban transportation problems. *Transportation Science* **49**(3) 623–636.

Osorio, Carolina, Kanchana Nanduri. 2015a. Energy-efficient urban traffic management: a microscopic simulation-based approach. *Transportation Science* **49**(3) 637–651.

Osorio, Carolina, Kanchana Nanduri. 2015b. Urban transportation emissions mitigation: Coupling high-resolution vehicular emissions and traffic models for traffic signal optimization. *Transportation Research Part B: Methodological* **81** 520–538.

Paz, Alexander, Victor Molano, Ember Martinez, Carlos Gaviria, Cristian Arteaga. 2015. Calibration of traffic flow models using a memetic algorithm. *Transportation Research Part C: Emerging Technologies* **55** 432–443.

Pell, Andreas, Andreas Meingast, Oliver Schauer. 2017. Trends in real-time traffic simulation. *Transportation Research Procedia* **25** 1477–1484.

Poloczek, Matthias, Jialei Wang, Peter Frazier. 2017. Multi-information source optimization. *Advances in Neural Information Processing Systems*. 4288–4298.

Rana, Santu, Cheng Li, Sunil Gupta, Vu Nguyen, Svetha Venkatesh. 2017. High dimensional Bayesian optimization with elastic Gaussian process. *International Conference on Machine Learning*. 2883–2891.

Rasmussen, Carl Edward, Hannes Nickisch. 2018. Gaussian processes for machine learning, version 4.2. URL `http://www.gaussianprocess.org/gpml/code/matlab/doc/`. Accessed: 2020-08-18.

Sasena, Michael J, Panos Papalambros, Pierre Goovaerts. 2002. Exploration of metamodeling sampling criteria for constrained global optimization. *Engineering Optimization* **34**(3) 263–278.

Schultz, Laura, Vadim Sokolov. 2018. Bayesian optimization for transportation simulators. *Procedia Computer Science* **130** 973–978.

Sebastiani, Mariana Teixeira, Ricardo Lüders, Keiko Veronica Ono Fonseca. 2016. Evaluating electric bus operation for a real-world BRT public transportation using simulation optimization. *IEEE Transactions on Intelligent Transportation Systems* **17**(10) 2777–2786.

Shahriari, Bobak, Kevin Swersky, Ziyu Wang, Ryan P Adams, Nando De Freitas. 2015. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE* **104**(1) 148–175.

Snoek, Jasper, Hugo Larochelle, Ryan P Adams. 2012. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*. 2951–2959.

Snoek, Jasper, Kevin Swersky, Rich Zemel, Ryan Adams. 2014. Input warping for Bayesian optimization of non-stationary functions. *International Conference on Machine Learning*. 1674–1682.

Srinivas, Niranjan, Andreas Krause, Sham M Kakade, Matthias Seeger. 2009. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995* .

Stafford, R. 2006. The theory behind the "randfixedsum" function. URL `http://www.mathworks.com/matlabcentral/fileexchange/9700-random-vectors-with-fixed-sum`. Accessed: 2017-11-25.

Stevanovic, Jelka, Aleksandar Stevanovic, Peter T Martin, Thomas Bauer. 2008. Stochastic optimization of traffic control and transit priority settings in VISSIM. *Transportation Research Part C: Emerging Technologies* **16**(3) 332–349.

Stone, Tom. 2021. Florida DOT picks Caliper Corporation for traffic modeling. URL `https://www.traffictechnologytoday.com/news/data/florida-dot-picks-caliper-corporation-for-traffic-modeling.html` Accessed: 2021-04-28.

Sun, Lihua, L Jeff Hong, Zhaolin Hu. 2014. Balancing exploitation and exploration in discrete optimization via simulation through a Gaussian process-based search. *Operations Research* **62**(6) 1416–1438.

Swersky, Kevin, Jasper Snoek, Ryan P Adams. 2013. Multi-task Bayesian optimization. *Advances in Neural Information Processing Systems*. 2004–2012.

Teklu, Fitsum, Agachai Sumalee, David Watling. 2007. A genetic algorithm approach for optimizing traffic control signals considering routing. *Computer-Aided Civil and Infrastructure Engineering* **22**(1) 31–43.

TSS-Transport Simulation Systems. 2009. Manhattan traffic model (MTM). URL `https://www.aimsun.com/aimsun-next-case-studies/manhattan-traffic-model-mtm/`. Accessed: 2021-04-28.

TSS-Transport Simulation Systems. 2015. Aimsun 8 users' manual. Version 8.0. (Barcelona, Spain).

TSS-Transport Simulation Systems. 2019. London operational network evaluation (ONE) model. URL `https://www.aimsun.com/aimsun-next-case-studies/london-operational-network-evaluation-one-model/`. Accessed: 2021-04-28.

Tympakianaki, Athina, Haris N Koutsopoulos, Erik Jenelius. 2015. c-SPSA: Cluster-wise simultaneous perturbation stochastic approximation algorithm and its application to dynamic origin–destination matrix estimation. *Transportation Research Part C: Emerging Technologies* **55** 231–245.

Tympakianaki, Athina, Haris N Koutsopoulos, Erik Jenelius. 2018. Robust SPSA algorithms for dynamic OD matrix estimation. *Procedia Computer Science* **130** 57–64.

Wang, Ziyu, Frank Hutter, Masrour Zoghi, David Matheson, Nando de Feitas. 2016. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research* **55** 361–387.

Williams, Christopher KI, Carl Edward Rasmussen. 2006. *Gaussian processes for machine learning*, vol. 2. MIT Press Cambridge, MA.

Wu, Jian, Peter Frazier. 2016. The parallel knowledge gradient method for batch bayesian optimization. *Advances in Neural Information Processing Systems*. 3126–3134.

Yun, Ilsoo, et al. 2006. Application of stochastic optimization method for an urban corridor. *Proceedings of the 2006 Winter Simulation Conference*. IEEE, 1493–1499.

Zhang, Chao, Carolina Osorio, Gunnar Flötteröd. 2017. Efficient calibration techniques for large-scale traffic simulators. *Transportation Research Part B: Methodological* **97** 214–239.

Zhou, Tianli, Carolina Osorio, Evan Fields. 2018. A data-driven discrete simulation-based optimization algorithm for large-scale two-way car-sharing network design. *Transportation Science* Under review. Available at: http://web.mit.edu/osorioc/www/papers/zhoOsoFieCarSharing.pdf.