Distributionally Ambiguous Optimization for Batch Bayesian Optimization

Nikitas Rontsis   NRONTSIS@GMAIL.COM
Michael A. Osborne   MOSB@ROBOTS.OX.AC.UK
Paul J. Goulart   PAUL.GOULART@ENG.OX.AC.UK
Department of Engineering Science
University of Oxford
Oxford, OX1 3PN UK

Abstract

We propose a novel, theoretically-grounded, acquisition function for Batch Bayesian Optimization informed by insights from distributionally ambiguous optimization. Our acquisition function is a lower bound on the well-known Expected Improvement function, which requires evaluation of a Gaussian expectation over a multivariate piecewise affine function. Our bound is computed instead by evaluating the best-case expectation over all probability distributions consistent with the same mean and variance as the original Gaussian distribution. Unlike alternative approaches, including Expected Improvement, our proposed acquisition function avoids multi-dimensional integrations entirely, and can be computed exactly – even on large batch sizes – as the solution of a tractable convex optimization problem. Our suggested acquisition function can also be optimized efficiently, since first and second derivative information can be calculated inexpensively as by-products of the acquisition function calculation itself. We derive various novel theorems that ground our work theoretically and we demonstrate superior performance via simple motivating examples, benchmark functions and real-world problems.

Keywords: Bayesian Optimization, Convex Optimization, Distributionally Robust Optimization, Batch Optimization, Black-Box Optimization.

1. Introduction

When dealing with numerical optimization problems in engineering applications, one is often faced with the optimization of a process or function that is expensive to evaluate and depends on a number of tuning parameters. Examples include the training of machine learning algorithms (Snoek et al., 2012), algorithms for robotic tasks (Lizotte et al., 2007) or reinforcement learning (Shahriari et al., 2016). Given the cost of evaluating the process, we wish to select the parameters at each stage of evaluation carefully so as to optimize the process using as few experimental evaluations as possible. We are concerned with problem instances wherein there is the capacity to speed up optimization by performing k experiments in *parallel* and, if needed, repeatedly select further batches with cardinality k as part of some sequential decision making process.

It is common to assume a surrogate model for the outcome $f : \mathbb{R}^n \mapsto \mathbb{R}$ of the process to be optimized. This model, which is based on both prior assumptions and past function

evaluations, is used to determine a collection of k input points for the next set of evaluations. Bayesian Optimization provides an elegant surrogate model approach and has been shown to outperform other state-of-the-art algorithms on a number of challenging benchmark functions (Jones, 2001). It models the unknown function f with a Gaussian Process (GP) (Rasmussen and Williams, 2005), a probabilistic function approximator which can incorporate prior knowledge such as smoothness, trends, etc.

A comprehensive introduction to GPs can be found in Rasmussen and Williams (2005). In short, modeling a function with a GP amounts to modeling the function as a realization of a stochastic process. In particular, we assume that the outcomes of function evaluations are normally distributed random variables with known *prior* mean function $m : \mathbb{R}^n \mapsto \mathbb{R}$ and *prior* covariance function $\kappa : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}$. Prior knowledge about f, such as smoothness and trends, can be incorporated through judicious choice of the covariance function κ, while the mean function m is commonly assumed to be zero. A training dataset $\mathcal{D} = (X^{\mathcal{D}}, y^{\mathcal{D}})$ of ℓ past function evaluations $y_i^{\mathcal{D}} = f(X_i^{\mathcal{D}})$ for $i = 1 \ldots \ell$, with $y^{\mathcal{D}} \in \mathbb{R}^\ell, X^{\mathcal{D}} \in \mathbb{R}^{\ell \times n}$ is then used to calculate the *posterior* of f.

The celebrated GP regression equations (Rasmussen and Williams, 2005) give the posterior

$$y|\mathcal{D} \sim \mathcal{N}(\mu(X), \Sigma(X)) \tag{1}$$

on a batch of k test locations $X \in \mathbb{R}^{k \times n}$ as a multi-variate normal distribution in a closed form formula. The posterior mean value μ and variance Σ depend also on the dataset $\mathcal{D}$, but we do not explicitly indicate this dependence in order to simplify the notation. Likewise, the posterior $y|\mathcal{D}$ is a normally distributed random variable whose mean $\mu(X)$ and covariance $\Sigma(X)$ depend on X, but we do not make this explicit.

Given the surrogate model, we wish to identify some selection criterion for choosing the next batch of points to be evaluated. Such a selection criterion is known as an *acquisition function* in the terminology of Bayesian Optimization. Ideally, such an acquisition function would take into account the number of remaining evaluations that we can afford, e.g. by computing a solution via dynamic programming to construct an optimal sequence of policies for future batch selections. However, a probabilistic treatment of such a criterion is computationally intractable, involving multiple nested maximization-marginalization steps (Gonzalez et al., 2016b).

To avoid this computational complexity, myopic acquisition functions that only consider the one-step return are typically used instead. For example, one could choose to maximize the one-step *Expected Improvement* (described more fully in §1.1) over the best evaluation observed so far, or maximize the probability of having an improvement in the next batch over the best evaluation. Other criteria use ideas from the bandit (Desautels et al., 2014) and information theory (Shah and Ghahramani, 2015) literature. In other words, the intractability of the multistep lookahead problem has spurred instead the introduction of a wide variety of myopic heuristics for batch selection.

Notation Used   We denote with $\mathbb{S}^k, \mathbb{S}_+^K$ and $\mathbb{S}_{++}^K$ the set of symmetric, positive semidefinite and positive definite matrices respectively. $A \succeq B$ ($A \succ B$) denotes that $A - B$ is positive (semi)definite. The symbol $\mathbb{E}_\mathbb{P}$ denotes the expectation over the probability distribution $\mathbb{P}$.

1.1. Expected improvement

We will focus on the (one-step) Expected Improvement criterion, which is a standard choice and has been shown to achieve good results in practice (Snoek et al., 2012). In order to give a formal description we first require some definitions related to the optimization procedure of the original process. At each step of the optimization procedure, define $y^{\mathcal{D}} \in \mathbb{R}^{\ell}$ as the vector of ℓ past function values evaluated at the points $X^{\mathcal{D}} \in \mathbb{R}^{\ell \times n}$, and $X \in \mathbb{R}^{k \times n}$ as a candidate set of k points for the next batch of evaluations. Then the classical Expected Improvement acquisition function is defined as

$$\alpha(X) = \underline{y^{\mathcal{D}}} - \mathbb{E}[\min(y_1, \ldots, y_k, \underline{y^{\mathcal{D}}})|\mathcal{D}]$$
$$\text{with } y|\mathcal{D} \sim \mathcal{N}\big(\mu(X), \Sigma(X)\big), \tag{2}$$

where $\underline{y^{\mathcal{D}}}$ is the element-wise minimum of $y^{\mathcal{D}}$, i.e. the minimum value of the function f achieved so far by any known function input. In the above definition we assume perfect knowledge of $\underline{y^{\mathcal{D}}}$, which implies a noiseless objective. A noisy objective requires the introduction of heuristics discussed in detail in Picheny et al. (2013). For the purposes of clarity, a noiseless objective is assumed for the rest of the document. This is not constraining, as most of the heuristics discussed in (Picheny et al., 2013) are compatible with the theoretical analysis presented in the rest of the paper.

Selection of a batch of points to be evaluated with optimal Expected Improvement amounts to finding some $X \in \arg\max[\alpha(X)]$. Unfortunately, direct evaluation of the acquisition function α requires the k–dimensional integration of a piecewise affine function; this is potentially a computationally expensive operation. This is particularly problematic for gradient-based optimization methods, wherein $\alpha(X)$ may be evaluated many times when searching for a maximizer. Regardless of the optimization method used, such a maximizer must also be computed again for every step in the original optimization process, i.e. every time a new batch of points is selected for evaluation. Therefore a tractable acquisition function should be used. In contrast to (2), the acquisition function we will introduce in Section 2 avoids expensive integrations and can be calculated efficiently with standard software tools.

1.2. Related work

Despite the intractability of (2), Chevalier and Ginsbourger (2013) presented an efficient way of calculating α and its derivative $d\alpha/dX$ (Marmin et al., 2015) by decomposing it into a sum of q–dimensional Gaussian Cumulative Distributions, which can be calculated efficiently using the seminal work of Genz and Bretz (2009). However, the number of calls to the q–dimensional Gaussian Cumulative Distribution grows quadratically with respect to the batch size q. To avoid this issue, Gonzalez et al. (2016a) and Ginsbourger et al. (2009) rely on heuristics to derive a multi-point criterion. Both methods choose the batch points in a greedy, sequential way, which restricts them from exploiting the interactions between the batch points in a probabilistic manner.

2. Distributionally ambiguous optimization for Bayesian Optimization

We now proceed to the main contribution of the paper. We draw upon ideas from the Distributionally Ambiguous Optimization community to derive a novel, tractable, acquisition function that lower bounds the expectation in (2). Our acquisition function:

- is theoretically grounded;

- is numerically reliable and consistent, unlike Expected Improvement-based alternatives (see §3);

- is fast and scales well with the batch size; and

- provides first and second order derivative information inexpensively.

In particular, we use the GP posterior (1) derived from the GP to determine the mean $\mu(X)$ and variance $\Sigma(X)$ of $y|\mathcal{D}$ given a candidate batch selection X, but we thereafter ignore the Gaussian assumption and consider only that $y|D$ has a distribution embedded within a family of distributions $\mathcal{P}$ that share the mean $\mu(X)$ and covariance $\Sigma(X)$ calculated by the standard GP regression equations. In other words, we define

$$\mathcal{P}(\mu, \Sigma) = \Big\{ \mathbb{P} \text{ is a p.d.f. on } \mathbb{R}^k \ \Big| \ \mathbb{E}_{\mathbb{P}}[\xi] = \mu, \mathbb{E}_{\mathbb{P}}[\xi\xi^T] = \Sigma + \mu\mu^T \Big\}.$$

We will omit the dependence of μ and Σ on X, and will denote the set $\mathcal{P}(\mu, \Sigma)$ simply as $\mathcal{P}$, where the context is clear. Note in particular that $\mathcal{N}(\mu, \Sigma) \in \mathcal{P}(\mu, \Sigma)$ for any choice of mean μ or covariance Σ.

One can then construct lower and upper bound for the Expected Improvement by minimizing or maximizing over the set $\mathcal{P}$ respectively, i.e. by writing

$$\inf_{\mathbb{P} \in \mathcal{P}} \mathbb{E}_{\mathbb{P}}[g(\xi)] \leq \mathbb{E}_{\mathcal{N}(\mu, \Sigma)}[g(\xi)] \leq \sup_{\mathbb{P} \in \mathcal{P}} \mathbb{E}_{\mathbb{P}}[g(\xi)], \tag{3}$$

where the random vector $\xi \in \mathbb{R}^k$ and the function $g : \mathbb{R}^k \mapsto \mathbb{R}$ are chosen according to the definition (2) of the Expected Improvement i.e., $\xi = y|\mathcal{D}$, and

$$g(\xi) = \underline{y^{\mathcal{D}}} - \min(\xi_1, \ldots, \xi_k, \underline{y^{\mathcal{D}}}) \tag{4}$$

so that $\mathbb{E}_{\mathcal{N}(\mu(X), \Sigma(X))}[g(\xi)] = \alpha(X)$. Thus, the middle term in (3) is equivalent to the Expected Improvement.

Perhaps surprisingly, both of the bounds in (3) are computationally tractable even though they seemingly require optimization over the infinite-dimensional (but convex) set of distributions $\mathcal{P}$. For either case, these bounds can be computed exactly via transformation of the problem to a tractable, convex optimization problem using distributionally ambiguous optimization techniques (Zymler et al., 2013).

We will focus on the upper bound $\sup_{\mathbb{P} \in \mathcal{P}} \mathbb{E}_{\mathbb{P}}[g(\xi)]$ in (3), hence adopting an *optimistic* modeling approach. The lower bound turns out to be of limited use, and we show in Proposition 8 of Appendix D that it is trivial to evaluate and independent of Σ. Hence, we informally assume that the distribution of function values is such that it yields the largest

possible Expected Improvement compatible with the mean and covariance computed by the GP, which we put together in the second order moment matrix Ω of the posterior as

$$\Omega := \begin{bmatrix} \Sigma + \mu\mu^T & \mu \\ \mu^T & 1 \end{bmatrix}. \tag{5}$$

We will occasionally write this explicitly as $\Omega(X)$ to highlight the dependency of the second order moment matrix on X.

The following result says that the upper (i.e. optimistic) bound in (3) can be computed via the solution of a convex semidefinite optimization problem whose objective function is linear in Ω. Semidefinite Problems (SDPs) are convex optimization problems with matrices as decision variables that are constrained to be positive semidefinite. SDPs enjoy strong theoretical results which guarantee that they can be solved globally in polynomial time, as well as a variety of mature software tools (O'Donoghue et al., 2016b), (MOSEK), (Garstka et al., 2019). The reader can refer to (Boyd and Vandenberghe, 2004) and (Vandenberghe and Boyd, 1996) for an introduction to SDPs.

Theorem 1 *For any $\Sigma \succ 0$ the optimal value of the semi-infinite optimization problem*

$$\sup_{\mathbb{P} \in \mathcal{P}(\mu, \Sigma)} \mathbb{E}_{\mathbb{P}}[g(\xi)]$$

coincides with the optimal value of the following semidefinite program:

$$\begin{aligned} p(\Omega) := -\sup \quad & \langle \Omega, M \rangle \\ s.t. \quad & M - C_i \preceq 0, \quad i = 0, \dots, k, \end{aligned} \tag{P}$$

where $M \in \mathbb{S}^{k+1}$ is the decision variable, $C_0 := 0$,

$$C_i := \begin{bmatrix} 0 & e_i/2 \\ e_i^T/2 & -\underline{y^{\mathcal{D}}} \end{bmatrix}, \qquad i = 1, \dots, k, \tag{6}$$

are auxiliary matrices defined using $\underline{y^{\mathcal{D}}}$ and e_i denotes the $k-$dimensional vector with a 1 in the $i-$th coordinate and 0s elsewhere.

Proof See Appendix A. ■

We therefore propose the computationally tractable acquisition function

$$\bar{\alpha}(X) := p\big(\Omega(X)\big) \geq \alpha(X) \quad \forall X \in \mathbb{R}^{k \times n},$$

which we will call *Optimistic Expected Improvement (OEI)*, as it is an optimistic variant of the Expected Improvement function in (2).

This computational tractability comes at the cost of inexactness in the bounds (3), which is a consequence of maximizing over a set of distributions containing the Gaussian distribution as just one of its members. Indeed, we prove in Theorem 10 of Appendix D that the maximizing distribution is discrete with $k + 1$ possible outcomes that can be constructed by the Lagrange multipliers of (P). We show in §3 that this inexactness is of limited consequence in practice and it mainly renders the acquisition function more

explorative. In particular, we show in Figure 1 that the qualitative behavior of OEI closely matches that of QEI. Nevertheless, there remains significant scope for tightening the bounds in (3) via imposition of additional convex constraints on the set $\mathcal{P}$, e.g. by restricting $\mathcal{P}$ to symmetric or unimodal distributions (Van Parys et al., 2015). Most of the results in this work would still apply, *mutatis mutandis*, if such structural constraints were to be included.

In contrast to the side-effect of inexactness, the distributional ambiguity is useful for integrating out the uncertainty of the GP's hyperparameters efficiently for our acquisition function. Given q samples of the hyperparameters, resulting in q second order moment matrices $\{\Omega_i\}_{i=1,...,q}$, we can estimate the resulting second moment matrix $\tilde{\Omega}$ of the marginalized, non-Gaussian, posterior as

$$\tilde{\Omega} \approx \frac{1}{q}\sum_{i=1}^{q}\Omega_i.$$

Due to the distributional ambiguity of our approach, both bounds of (3) can be calculated directly based on $\tilde{\Omega}$, hence avoiding multiple calls to the acquisition function.

Although the value of $p(\Omega)$ for any fixed Ω is computable via solution of an SDP, the non-convexity of the GP posterior (1) that defines the mapping $X \mapsto \Omega(X)$ means that $\bar{\alpha}(X) = p\big(\Omega(X)\big)$ is still non-convex in X. This is unfortunate, since we ultimately wish to maximize $\bar{\alpha}(X)$ in order to identify the next batch of points to be evaluated experimentally.

However we can still optimize $\bar{\alpha}$ locally via non-linear programming. We will establish that a second order method is applicable by showing that $\bar{\alpha}(X)$ is twice differentiable under mild conditions. Such an approach would also be efficient as the Hessian of $\bar{\alpha}$ can be calculated inexpensively. To show these results we will begin by considering the differentiability of p as a function of Ω.

Theorem 2 *The optimal value function $p : \mathbb{S}_{++}^{k+1} \mapsto \mathbb{R}$ defined in problem (P) is differentiable on its domain with $\partial p(\Omega)/\partial \Omega = -\bar{M}(\Omega)$, where $\bar{M}(\Omega)$ is the unique optimal solution of (P) at Ω.*

Proof See Appendix B. ■

The preceding result shows that $\partial p(\Omega)/\partial \Omega$ is produced as a byproduct of evaluation of $\sup_{\mathbb{P}\in\mathcal{P}} \mathbb{E}_{\mathbb{P}}[g(\xi)]$, since it is simply $-\bar{M}(\Omega)$, the negation of the unique optimizer of (P). The simplicity of this result suggests consideration of second derivative information of $p(\Omega)$, i.e. derivatives of $-\bar{M}(\Omega)$. The following result proves that this is well defined and tractable for any $\Omega \succ 0$:

Theorem 3 *The optimal solution function $\bar{M} : \mathbb{S}_{++}^{k+1} \mapsto \mathbb{S}^{k+1}$ in problem (P) is differentiable on $\mathbb{S}_{++}^{k+1}$. Every directional derivative of $\bar{M}(\Omega)$ is the unique solution to a sparse linear system with $\mathcal{O}(k^3)$ nonzeros.*

Proof See Appendix C. ■

We can now consider the differentiability of $\bar{\alpha} = p \circ \Omega$. The following Corollary establishes this under certain conditions.

Corollary 4 *$\bar{\alpha} : \mathbb{R}^{k\times n} \mapsto \mathbb{R}$ is twice differentiable on any X for which $\Sigma(X) \succ 0$ and the mean and kernel functions of the underlying GP are twice differentiable.*

Proof By examining the GP Regression equations (Rasmussen and Williams, 2005) and Equation (5), we conclude that $\Omega(X)$ is twice differentiable on $\mathbb{R}^{k \times n}$ if the kernel and mean functions of the underlying Gaussian Process are twice differentiable. Hence, $\bar{\alpha}(X) = p(\Omega(X))$ is twice differentiable for any $\Omega(X) \succ 0$ as a composition of twice differentiable functions. Examining (5) reveals that $\Omega(X) \succ 0$ is equivalent to $\Sigma(X) \succ 0$, which concludes the proof. ∎

A rank deficient $\Sigma(X) \not\succ 0$ implies perfectly correlated outcomes. At these points both OEI and QEI can be shown to be non-differentiable. However, this is not constraining in practice as both QEI and OEI can be calculated by considering a smaller, equivalent problem. It is also not an issue for ascent based methods for maximizing $\bar{\alpha}$, as a ascent direction can be obtained by an appropriate perturbation of the perfectly correlated points.

We are now in a position to derive expressions for the gradient and the Hessian of $\bar{\alpha} = p \circ \Omega$. For simplicity of notation we consider derivatives over $\bar{x} = \text{vec}(X)$. Application of the chain rule to $\bar{\alpha}(\bar{x}) = p(\Omega(\bar{x}))$ gives:

$$\frac{\partial \bar{\alpha}(\bar{x})}{\partial \bar{x}_{(i)}} = \left\langle \frac{\partial p(\Omega)}{\partial \Omega}, \frac{\partial \Omega(\bar{x})}{\partial \bar{x}_{(i)}} \right\rangle = -\left\langle \bar{M}(\Omega), \frac{\partial \Omega(\bar{x})}{\partial \bar{x}_{(i)}} \right\rangle. \tag{7}$$

Note that the second term in the rightmost inner product above depends on the particular choice of covariance function κ and mean function m. It is straightforward to compute (7) in modern graph-based `autodiff` frameworks, such as the `TensorFlow`-based `GPflow`. Differentiating again (7) gives the Hessian of $\bar{\alpha}$:

$$\begin{aligned}
\frac{\partial^2 \bar{\alpha}(\bar{x})}{\partial \bar{x}_{(i)} \partial \bar{x}_{(j)}} &= -\frac{\partial}{\partial x_{(i)}} \left\langle \bar{M}(\Omega), \frac{\partial \Omega(\bar{x})}{\partial \bar{x}_{(j)}} \right\rangle \\
&= -\left\langle \bar{M}(\Omega), \frac{\partial^2 \Omega(\bar{x})}{\partial \bar{x}_{(i)} \partial \bar{x}_{(j)}} \right\rangle - \left\langle \frac{\partial \bar{M}(\Omega(\bar{x}))}{\partial \bar{x}_{(i)}}, \frac{\partial \Omega(\bar{x})}{\partial \bar{x}_{(j)}} \right\rangle,
\end{aligned} \tag{8}$$

where $\partial \bar{M} / \partial \bar{x}_{(i)}$ is the directional derivative of $\bar{M}(\Omega)$ across the perturbation $\partial \Omega(\bar{x}) / \partial \bar{x}_{(i)}$. According to Theorem 3, each of these directional derivatives exists and can be computed via solution of a sparse linear system.

3. Empirical analysis

In this section we demonstrate the effectiveness of our acquisition function against a number of state-of-the-art alternatives. The acquisition functions we consider are listed in Table 1. We do not compare against PPES as it is substantially more expensive and elaborate than our approach and there is no publicly available implementation of this method.

We show that our acquisition function OEI achieves better performance than alternatives and highlight simple "failure" cases exhibited by competing methods. In making the following comparisons, extra care should be taken in the setup used. This is because Bayesian Optimization is a multifaceted procedure that depends on a collection of disparate elements (e.g. kernel/mean function choice, normalization of data, acquisition function, optimization of the acquisition function) each of which can have a considerable effect on the resulting performance (Snoek et al., 2012; Shahriari et al., 2016). For this reason we test

Table 1: *List of acquisition functions*

Key	Description
OEI	Optimistic Expected Improvement *(Our novel algorithm)*
QEI	Multi-point Expected Improvement (Marmin et al., 2015)
QEI-CL	Constant Liar (*"mix"* strategy) (Ginsbourger et al., 2010)
LP-EI	Local Penalization Expected Improvement (Gonzalez et al., 2016a)
BLCB	Batch Lower Confidence Bound (Desautels et al., 2014)

the different algorithms on a unified testing framework, based on `GPflow`, available online at https://github.com/oxfordcontrol/Bayesian-Optimization.

Our acquisition function is evaluated via solution of a semidefinite program, and as such it benefits from the huge advances of the convex optimization field. A variety of standard tools exist for solving such problems, including `MOSEK` (MOSEK), `SCS` (O'Donoghue et al., 2016a) and `CDCS` (Zheng et al., 2017). We chose the first-order (Boyd and Vandenberghe, 2004), freely-available solver `SCS`, which scales well with batch size and allows for solver warm-starting between acquisition function evaluations.

Warm starting allows for a significant speedup since the acquisition function is evaluated repeatedly at nearby points by the non-linear solver. This results in solving (P) repeatedly for similar Ω. Warm-starting the SDP solver with the previous solution reduces `SCS`'s iterations by 77% when performing the experiments of Figure 3. Moreover, Theorem 2 provides the means for a first-order warm starting. Indeed, the derivative of the solution across the change of the cost matrix Ω can be calculated, allowing us to take a step in the direction of the gradient and warm start from that point. This reduces `SCS`'s iterations by a further 43%.

Indicative timing results for the calculation of OEI, QEI and their derivatives are listed in Table 2. The dominant operation for calculating OEI and its gradient is solving (P). This makes OEI much faster than QEI, which is in line with the complexity of the dominant operation in SDP solvers based on first-order operator splitting methods such as `SCS` or `CDCS` which, for our problem, is $\mathcal{O}(k^4)$. Assume that, given the solution of (P), we want to also calculate the Hessian of OEI. This would entail the following two operations:

Calculating $\partial \bar{M} / \partial X_{(i,j)}$ **given** $\partial \Omega(X) / \partial X_{(i,j)}$. According to Lemma 3 this can be obtained as a solution to a sparse linear system. We used `Intel`® `MKL PARDISO` to solve efficiently these linear systems.

Calculate $\partial \Omega(X) / \partial X_{(i,j)}$ **and apply chain rules** of (8) to get the Hessian of the acquisition function $\bar{\alpha} = p \circ \Omega$ given the gradient and Hessian of p. We used `Tensorflow`'s Automatic Differentiation for this part, without any effort to optimize its performance. Considerable speedups can be brought by e.g. running this part on GPU, or automatically generating low-level code optimized specifically for this operation.

Table 2: *Average execution time of the acquisition function, its gradient and Hessian when running BO in the Eggholder function on an Intel E5-2640v3 CPU. For batch size 40, QEI fails, i.e. it always returns 0 without any warning message. For the execution time of the Hessian we assume knowledge of the solution of* (P). *Its timing is split into two parts as described in the main text. Note that these results present a qualitative picture as OEI and QEI are coded in different languages and use different underlying libraries.*

| Batch Size	QEI: $\alpha(X), \nabla\alpha(X)$	OEI: $\bar{\alpha}(X), \nabla\bar{\alpha}(X)$	$\nabla^2\bar{\alpha}(X)$	
		Solve (P)	$\partial\bar{M}$	`Tensorflow` Part
2	$5.6 \cdot 10^{-3}$	$2.1 \cdot 10^{-4}$	$5.3 \cdot 10^{-4}$	$1.0 \cdot 10^{-2}$
3	$1.2 \cdot 10^{-2}$	$3.8 \cdot 10^{-4}$	$7.5 \cdot 10^{-4}$	$1.4 \cdot 10^{-2}$
6	$1.1 \cdot 10^{-1}$	$1.5 \cdot 10^{-3}$	$1.7 \cdot 10^{-3}$	$2.0 \cdot 10^{-2}$
10	1.1	$8.2 \cdot 10^{-3}$	$5.9 \cdot 10^{-3}$	$3.2 \cdot 10^{-2}$
20	$2.1 \cdot 10^{1}$	$4.7 \cdot 10^{-2}$	$2.3 \cdot 10^{-2}$	$8.7 \cdot 10^{-2}$
40	$-$	$3.4 \cdot 10^{-1}$	$1.4 \cdot 10^{-1}$	$3.5 \cdot 10^{-1}$

Note that the computational tractability of the Hessian is only allowed due to the novelty of Theorem 2 which exploits the "structure" of (P)'s optimizer.

We chose the `KNITRO v10.3` (Byrd et al., 2006) Sequential Quadratic Optimization (SQP) non-linear solver with the default parameters for the optimization of OEI. Explicitly providing the Hessian on the experiments of Figure 3 reduces `KNITRO`'s iterations by 49% as compared to estimating the Hessian via the symmetric-rank-one (SR1) update method included in the `KNITRO` suite. Given the inexpensiveness of calculating the Hessian and the fact that `KNITRO` requests the calculation of the Hessian less than a third as often as it requires the objective function evaluation we conclude that including the Hessian is beneficiary.

We will now present simulation results to demonstrate the performance of OEI in various scenarios.

3.1. Perfect modeling assumptions

We first demonstrate that the competing Expected Improvement based algorithms produce clearly suboptimal choices in a simple 1–dimensional example. We consider a 1–d Gaussian Process on the interval $[-1, 1]$ with a squared exponential kernel (Rasmussen and Williams, 2005) of lengthscale $1/10$, variance 10, noise 10^{-6} and a mean function $m(x) = (5x)^2$. An example posterior of 10 observations is depicted in Figure 1(a). Given the GP and the 10 observations, we depict the optimal 3-point batch chosen by maximizing each acquisition function. Note that in this case we assume perfect modeling assumptions – the GP is completely known and representative of the actual process. We can observe in Figure 1(a) that the OEI choice is very close to the one proposed by QEI while being slightly more explorative, as OEI allows for the possibility of more exotic distributions than the Gaussian.
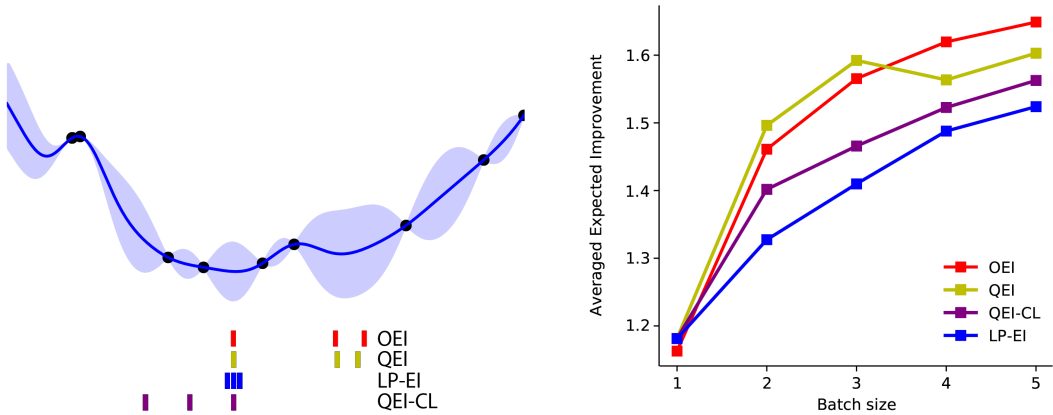
(a) *Suggested 3-point batches of different algorithms for a GP posterior, depicted on [−1, 1], given 10 observations. The thick blue line depicts the GP mean, the light blue shade the uncertainty intervals (± sd) and the black dots the observations. The locations of the batch chosen by each algorithm are depicted with colored vertical lines at the bottom of the figure.*

(b) *Averaged one-step Expected Improvement on 200 GP posteriors of sets of 10 observations with the same generative characteristics (kernel, mean, noise) as the one in Figure 1 for different algorithms across varying batch size.*
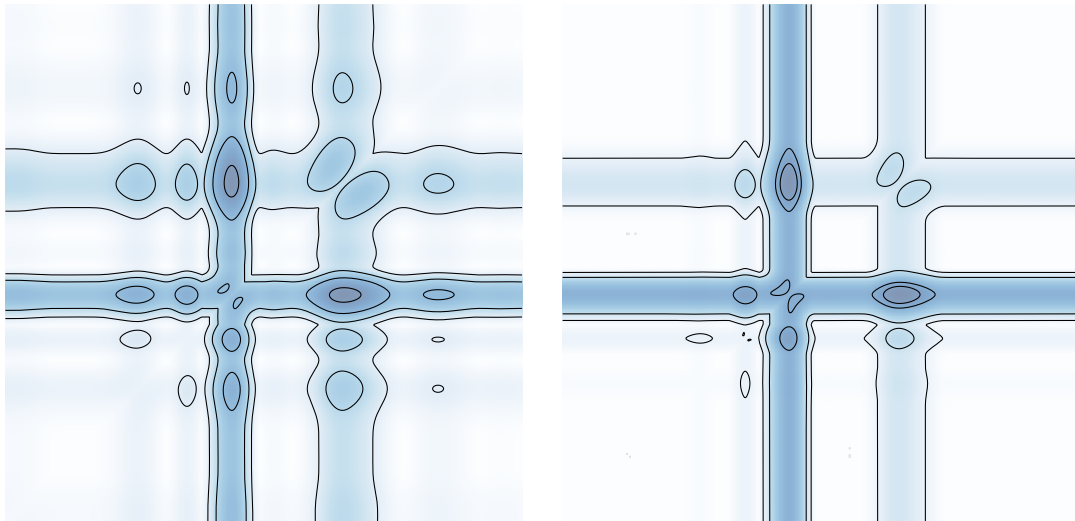
(c) *Contour plots of evaluating 2-point batch selections on the GP posterior of Figure 1(a) across [−1, 1]² with OEI (left) and QEI (right). OEI closely approximates QEI in the sense that their optimizers nearly coincide and their level sets are similar. Note that the level sets of the two figures correspond to different values.*

Figure 1: *Simulation results on Gaussian Process draws.*

In contrast the LP-EI heuristic samples almost at the same point all three times. This can be explained as follows: LP-EI is based on a Lipschitz criterion to penalize areas around previous choices. However, the Lipschitz constant for this function is dominated by the end points of the function (due to the quadratic trend), which is clearly not suitable for the area of the minimizer (around zero), where the Lipschitz constant is approximately zero. On the other hand, QEI-CL favors suboptimal regions. This is because QEI-CL postulates outputs equal to the mean value of the observations which significantly alter the GP posterior.

We proceed to test the algorithms on 200 different posteriors, generated by creating sets of 10 observations drawn from the previously defined GP. For each of the 200 posteriors, each algorithm chooses a batch, the performance of which is evaluated by sampling the multipoint Expected Improvement (2). The averaged results are depicted in Figure 1(b). For a batch size of 1 all of the algorithms perform the same, except for OEI which performs slightly worse due to the relaxation of Gaussianity. For batch sizes 2-3, QEI is the best strategy, while OEI is a very close second. For batch sizes 4-5 OEI performs significantly better. Figure 1(c) explains the very good performance of OEI. Although always different from the sampled estimate, it closely approximates the actual Expected Improvement in the sense that their optimizers and level sets are in close agreement. The deterioration of the performance for QEI in Figure 1(b) on batch sizes 4 and 5 might be related with software issues of the R package `DiceOptim` as there appear to exist points that `DiceOptim`'s results of the multipoint Expected Improvement differ considerably from sampled estimates.

3.2. Synthetic functions

Next, we evaluate the performance of OEI in minimizing synthetic benchmark functions. The functions considered are: the Six-Hump Camel function defined on $[-2, 2] \times [-1, 1]$, the Hartmann 6 dimensional function defined on $[0, 1]^6$ and the Eggholder function, defined on $[-512, 512]^2$. We compare the performance of OEI against QEI, LP-EI and BLCB as well as random uniform sampling. The initial dataset consists of 10 random points for all the functions. A Matern 3/2 kernel is used for the GP modeling (Rasmussen and Williams, 2005). As all of the considered functions are noiseless, we set the likelihood variance to a fixed small number 10^{-6} for numerical reasons. For the purpose of generality, the input domain of every function is scaled to $[-0.5, 0.5]^n$ while the observation dataset y_d is normalized at each iteration, such that $\mathrm{Var}[y^{\mathcal{D}}] = 1$. The same transformations are applied to QEI, LP-EI and BLCB for reasons of consistency. All the acquisition functions except OEI are optimized with the quasi-Newton L-BFGS-B algorithm (Fletcher, 1987) with 20 random restarts. We use point estimates for the kernel's hyperparameters obtained by maximizing the marginal likelihood via L-BFGS-B restarted on 20 random initial points.

First, we consider a small-scale scenario of *batch size 5*. The results of 40 runs of Bayesian Optimization on a mixture of Cosines, the Six-Hump Camel, Eggholder, and 6-d Hartmann functions are depicted in Figure 2. In these functions, OEI approaches the minimums faster than QEI and BLCB while considerably outperforming LP-EI, which exhibits outliers with bad behavior. The explorative nature of OEI can be observed in the optimization of the Hartmann function. OEI quickly reaches the vicinity of the minimum, but then decides not to refine the solution further but explore instead the rest of the 6-d space. Increasing the *batch size* to *20* for the challenging Eggholder and Hartmann functions shows a further
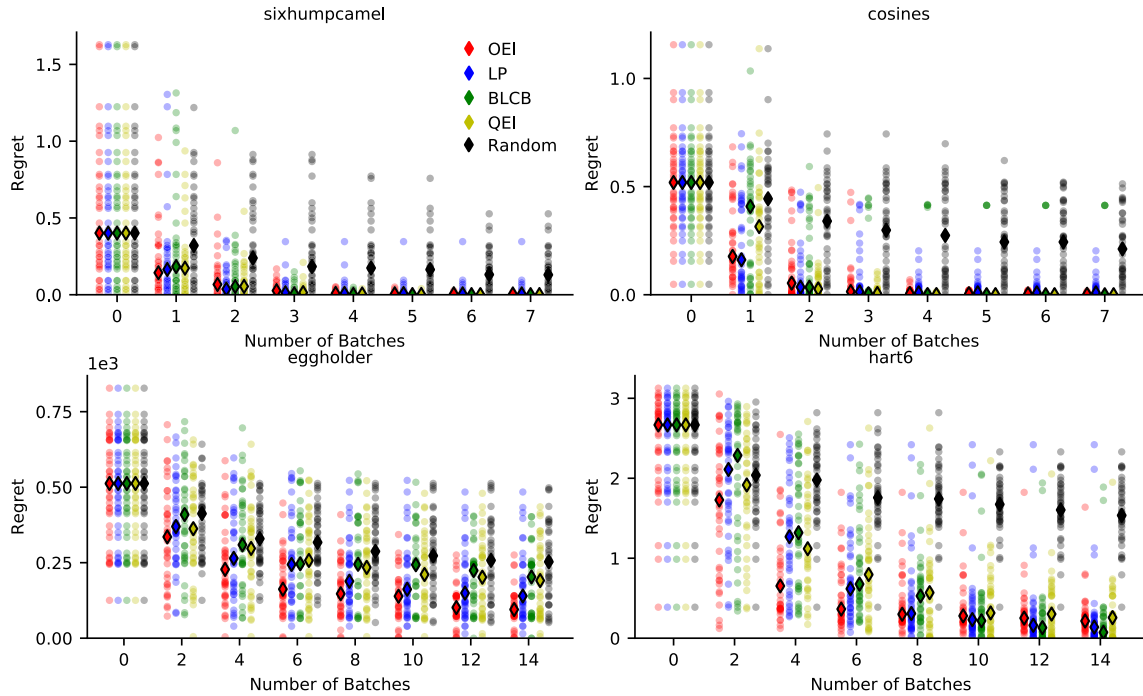
Figure 2: *BO of batch size 5 on synthetic functions. Red, blue, green, yellow and black dots depict runs of OEI, LP-EI, BLCB, QEI and Random algorithms respectively. Diamonds depict the median regret for each algorithm.*
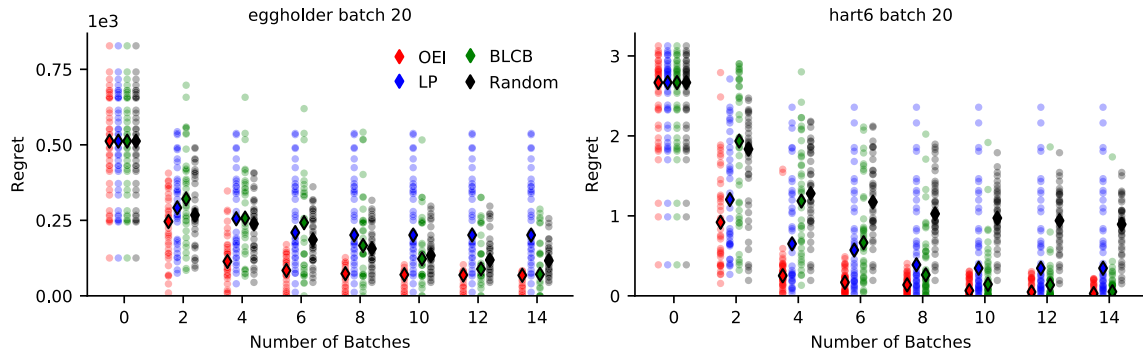

Figure 3: *BO of batch size 20 on the challenging Hartmann 6-d Eggholder 2-d functions where OEI exhibits clearly superior performance. Red, blue, green and black dots depict runs of OEI, LP-EI, BLCB and Random algorithms respectively. Diamonds depict the median regret for each algorithm. Compare the above results with Figure 2 for the same runs with a smaller batch size. Note that QEI is not included in this case, as it does not scale to large batch sizes (see Table 2).*

advantage for OEI. Indeed, as we can observe in Figure 3, OEI successfully exploits the increased batch size. BLCB also improves its performance though not to the extent of OEI. In contrast, LP-EI fails to manage the increased batch size. This is partially expected due to the heuristic based nature of LP-EI: the Lipschitz constant estimated by LP-EI is rarely

suitable for all the 20 locations of the suggested batch. Even worse, LP-EI's performance is *deteriorated* as compared to smaller batch sizes. LP-EI is plagued by numerical issues in the calculation of its gradient, and suggests multiple re-evaluations of the same points. This multiple re-samplings affects the GP modeling, resulting in an inferior overall BO performance.

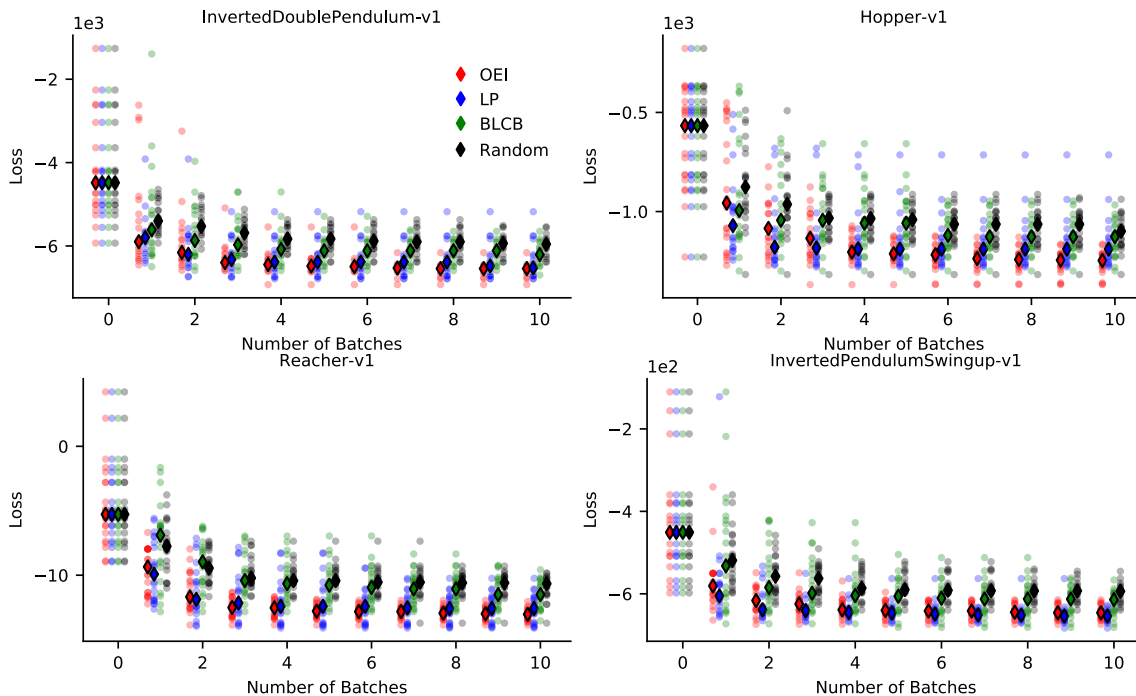### 3.3. Real world example: Tuning a Reinforcement Learning Algorithm on various tasks

Figure 4: *BO of batch size 20 on tuning PPO on a variety of robotic tasks. Red, blue, green and black dots depict runs of OEI, LP-EI, BLCB and Random algorithms respectively. Diamonds depict the median loss of all the runs for each algorithm.*

Finally we perform Bayesian Optimization to tune Proximal Policy Optimization (PPO), a state-of-the-art Deep Reinforcement Learning algorithm that has been shown to outperform several policy gradient reinforcement learning algorithms (Schulman et al., 2017). The problem is particularly challenging, as deep reinforcement learning can be notoriously hard to tune, without any guarantees about convergence or performance. We use the implementation Dhariwal et al. (2017) published by the authors of PPO and tune the reinforcement algorithm on 4 `OpenAI Gym` tasks (Hopper, InvertedDoublePendulum, Reacher and InvertedPendulumSwingup) using the `Roboschool` robot simulator. We tune a set of 5 hyper-parameters which are listed in Table 3. We define as objective function the negative average reward per episode over the entire training period ($4 \cdot 10^5$ timesteps), which favors fast learning (Schulman et al., 2017). All of the other parameters are the same as the original implementation (Schulman et al., 2017).

Table 3: *List of PPO's Hyperparameters used for tuning. Items with asterisk are tuned in the log-space.*

Hyperparameter	Range
Adam step-size	$[10^{-5}, 10^{-3}]^*$
Clipping parameter	$[0.05, 0.5]$
Batch size	$24, \ldots, 256$
Discount Factor (γ)	$1 - [10^{-3}, 10^{-3/2}]^*$
GAE parameter (λ)	$1 - [10^{-2}, 10^{-1}]^*$

We run Bayesian Optimization with batch size of 20, with the same modeling, preprocessing and optimization choices as the ones used in the benchmark functions. The results of 20 runs are depicted in Figure 4. OEI outperforms, on average, BLCB (which performs comparably to Random search), and, w.r.t. the variance of the outcomes, LP-EI, which exhibits occasional outliers stuck in inferior solutions.

4. Conclusions

We have introduced a new acquisition function that is a tractable, probabilistic relaxation of the multi-point Expected Improvement, drawing ideas from the Distributionally Ambiguous Optimization community. Novel theoretical results allowed inexpensive calculation of first and second derivative information resulting in efficient Bayesian Optimization on large batch sizes. In our experiments we performed BO with batch size 20. We found that increasing the batch size further was difficult because of the increased computation cost of solving the SDP problems (P). Novel advances in Semidefinite Programming (Rontsis et al., 2019) might allow for larger batch sizes and we believe that there exists significant scope for further research in this area.

Further directions include applying our distributionally agnostic approach to non-Gaussian setups e.g. to Bayesian Optimization with Student t-processes (Shah et al., 2014), testing performance in noisy setups and examining the asymptotic properties of OEI. Finally, we believe that the distributionally ambiguous techniques used in this document can be applied to other machine learning fields such as Gaussian Process based Reinforcement Learning (Deisenroth et al., 2015).

Acknowledgments

Appendix A. Value of the Expected Improvement Lower Bound

In this section we provide a proof of the first of our main results, Theorem 1, which establishes that for any $\Sigma \succ 0$ one can compute the value of our optimistic upper bound function

$$\sup_{\mathbb{P} \in \mathcal{P}(\mu, \Sigma)} \mathbb{E}_{\mathbb{P}}[g(\xi)] \tag{9}$$

via solution of a convex optimization problem in the form of a semidefinite program.

Proof of Theorem 1:

Recall that the set $\mathcal{P}(\mu, \Sigma)$ is the set of all distributions with mean μ and covariance Σ. Following the approach of Zymler et al. (2013, Lemma A1), we first remodel problem (9) as:

$$
\begin{aligned}
-\inf_{\nu \in \mathcal{M}_+} \quad & \int_{\mathbb{R}^k} -g(\xi)\nu(d\xi) \\
\text{s.t.} \quad & \int_{\mathbb{R}^k} \nu(d\xi) = 1 \\
& \int_{\mathbb{R}^k} \xi \nu(d\xi) = \mu \\
& \int_{\mathbb{R}^k} \xi \xi^T \nu(d\xi) = \Sigma + \mu\mu^T,
\end{aligned}
\tag{10}
$$

where $\mathcal{M}_+$ represents the cone of nonnegative Borel measures on $\mathbb{R}^k$. The optimization problem (10) is a semi-infinite linear program, with infinite dimensional decision variable ν and a finite collection of linear equalities in the form of moment constraints.

As shown by Zymler et al. (2013), the dual of problem (10) has instead a finite dimensional set of decision variables and an infinite collection of constraints, and can be written as

$$
\begin{aligned}
-\sup \quad & \langle \Omega, M \rangle \\
\text{s.t.} \quad & \left[\xi^T\ 1\right] M \left[\xi^T\ 1\right]^T \leq -g(\xi) \quad \forall \xi \in \mathbb{R}^k,
\end{aligned}
\tag{11}
$$

with $M \in \mathbb{S}^{k+1}$ the decision variable and $\Omega \in \mathbb{S}^{k+1}$ the second order moment matrix of ξ (see (5)). Strong duality holds between problems (10) and (11) for any $\Omega \succ 0 \Leftrightarrow \Sigma \succ 0$, i.e. there is zero duality gap and their optimal values coincide.

The dual decision variables in (11) form a matrix M of Lagrange multipliers for the constraints in (10) that is block decomposable as

$$M = \begin{pmatrix} M_{11} & m_{12} \\ m_{12}^T & m_{22} \end{pmatrix},$$

where $M_{11} \in \mathbb{S}^k$ are multipliers for the second moment constraint, $m_{12} \in \mathbb{R}^k$ multipliers for the mean value constraint, and m_{22} a scalar multiplier for the constraint that $\nu \in \mathcal{M}_+$ should integrate to 1, i.e. that ν should be a probability measure.

For our particular problem, we have:

$$
\begin{aligned}
-g(\xi) &= \min(\xi_{(1)}, \dots, \xi_{(k)}, \underline{y}^{\mathcal{D}}) - \underline{y}^{\mathcal{D}} \\
&= \min(\xi_{(1)} - \underline{y}^{\mathcal{D}}, \dots, \xi_{(k)} - \underline{y}^{\mathcal{D}}, 0),
\end{aligned}
$$

as defined in (4), so that (11) can be rewritten as

$$
\begin{aligned}
-\sup \quad & \langle \Omega, M \rangle \\
\text{s.t.} \quad & \begin{bmatrix} \xi^T\ 1 \end{bmatrix} M \begin{bmatrix} \xi^T\ 1 \end{bmatrix}^T \le \xi_{(i)} - \underline{y}^{\mathcal{D}}, \qquad \forall \xi \in \mathbb{R}^k \\
& \begin{bmatrix} \xi^T\ 1 \end{bmatrix} M \begin{bmatrix} \xi^T\ 1 \end{bmatrix}^T \le 0 \qquad i = 1, \dots, k.
\end{aligned}
\tag{12}
$$

The infinite dimensional constraints in (12) can be replaced by the equivalent conic constraints

$$
M - \begin{bmatrix} 0 & e_i/2 \\ e_i^T/2 & -\underline{y}^{\mathcal{D}} \end{bmatrix} \preceq 0 \qquad i = 1, \dots, k,
$$

and $M \preceq 0$ respectively, where e_i denotes the k−dimensional vector with a 1 in the i−th coordinate and 0s elsewhere. Substituting the above constraints in (12) results in (P), which proves the claim. ∎

Appendix B. Gradient of the Expected Improvement Lower Bound

In this section we provide a proof of our second main result, Theorem 2, which shows that the gradient[1] $\partial p/\partial \Omega$ of our upper bound function (3) with respect to Ω coincides with the optimal solution of the semidefinite program (P). We will find it useful to exploit also the dual of this SDP, which we can write as

$$
\begin{aligned}
-\inf \quad & \sum_{i=1}^{k} \langle Y_i, C_i \rangle \\
\text{s.t.} \quad & Y_i \succeq 0, \quad i = 0, \dots, k \\
& \sum_{i=0}^{k} Y_i = \Omega,
\end{aligned}
\tag{13}
$$

and the Karush-Kuhn-Tucker conditions for the pair of primal and dual solutions $\bar{M}$, $\{\bar{Y}_i\}$:

$$
C_i - \bar{M} \succeq 0 \tag{14}
$$

$$
\bar{Y}_i \succeq 0 \tag{15}
$$

$$
\langle \bar{Y}_i, \bar{M} - C_i \rangle = 0 \Rightarrow \bar{Y}_i(\bar{M} - C_i) = 0 \tag{16}
$$

$$
\left. \frac{\partial \mathcal{L}(M, \Omega)}{\partial M} \right|_{\bar{M}} = 0 \Rightarrow \sum_{i=0}^{k} \bar{Y}_i = \Omega, \tag{17}
$$

where $\mathcal{L}$ denotes the Lagrangian of (P).

Before proving Theorem 2 we require three ancillary results. The first of these results establishes that any feasible point M for the optimization problem (P) has strictly negative definite principal minors in the upper left hand corner.

Lemma 5 *For any feasible $M \in \mathbb{S}^{k+1}$ of (P) the upper left $k \times k$ matrix M_{11} is negative definite.*

---

1. Technically, the gradient is not defined, as Ω is by definition symmetric. We can get around this technicality by a slight abuse of notation allowing for a non-symmetric Ω such that $\Omega + \Omega^T \in \mathbb{S}_{++}^{k+1}$.

Proof Let

$$M = \begin{bmatrix} M_{11} & m_{12} \\ m_{12}^T & m_{22} \end{bmatrix},$$

where $M_{11} \in \mathbb{S}^k, m_{12} \in \mathbb{R}^k$ and $m_{22} \in \mathbb{R}$.

From (6) we can infer that $m_{22} + \underline{y^{\mathcal{D}}} < 0$, otherwise (6) would require $m_{12} + e_i/2 = 0 \; \forall i = 1, \ldots k$, which is impossible.

Since $M \preceq 0$, we have $m_{22} \leq 0$. Assume though, for now, that $m_{22} < 0$. Applying then a standard Schur complement identity in (6) results in:

$$M_{11} \preceq (m_{12} - e_i)(m_{22} + \underline{y^{\mathcal{D}}})^{-1}(m_{12} - e_i)^T$$
$$M_{11} \preceq m_{12} m_{22}^{-1} m_{12}^T \quad i = 1, \ldots, k.$$

Summing the above results in

$$M_{11} \preceq \frac{(m_{22} + \underline{y^{\mathcal{D}}})^{-1}}{k+1} \sum_{i=1}^{k} (m_{12} - e_i)(m_{12} - e_i)^T + \frac{m_{22}^{-1}}{k+1} m_{12} m_{12}^T,$$

which results in $M_{11} \prec 0$, since $\mathrm{span}\big(\{m_{12}, \{m_{12} - e_i\}_{i=i,\ldots,k}\}\big) \supseteq \mathrm{span}\big(\{m_{12} - m_{12} + e_i\}_{i=i,\ldots,k}\big) = \mathbb{R}^k$.

Finally, in the case where $m_{22} = 0$ we have $m_{12} = 0$, since $M \preceq 0$. Applying the above results in $M_{11} \preceq (m_{22} + \underline{y^{\mathcal{D}}})^{-1}/k \sum_{i=1}^{k} \frac{1}{k} e_i e_i^T \prec 0$. ∎

The second auxiliary results lists some useful properties of the dual solution:

Lemma 6 *The optimal Lagrange multipliers of* (P) *are of rank one with* $\mathcal{R}(\bar{Y}_i) = \mathcal{N}(\bar{M} - C_i)$, $\forall i = 0, \ldots, k$, *where* $\mathcal{N}(\cdot)$ *and* $\mathcal{R}(\cdot)$ *denote the nullspace and the range of a matrix.*

Proof Lemma 5 implies that $[x^T 0](\bar{M} - C_i)[x^T 0]^T = [x^T 0]\bar{M}[x^T 0]^T < 0, \forall x \in \mathbb{R}^k$ (recall that C_i is nonzero only in the last column or the last row), which means that $\mathrm{rank}(\bar{M} - C_i) \geq k$. Due to the complementary slackness condition (16), the span of $\bar{Y}_i$ is orthogonal to the span of $\bar{M} - C_i$ and consequently $\mathrm{rank}(\bar{Y}_i) \leq 1$. However, according to (17) we have

$$\mathrm{rank} \sum_{i=0}^{k} \bar{Y}_i = \mathrm{rank}(\Omega) \overset{\Omega \succ 0}{\Longrightarrow} \sum_{i=0}^{k} \mathrm{rank}(\bar{Y}_i) \geq k + 1$$

which results in

$$\mathrm{rank}(\bar{M} - C_i) = k, \quad \mathrm{rank}(\bar{Y}_i) = 1,$$

and, using (16):

$$\mathcal{R}(\bar{Y}_i) = \mathcal{N}(\bar{M} - C_i), \quad i = 0, \ldots, k.$$

∎

Our final ancillary result considers the (directional) derivative of the function p when its argument is varied linearly along some direction $\bar{\Omega}$.

Lemma 7 *Given any $\bar{\Omega} \in \mathbb{S}^{k+1}$ and any moment matrix $\Omega \in \mathbb{S}_{++}^{k+1}$, define the scalar function $q(\cdot\,;\Omega) : \mathbb{R} \to \mathbb{R}$ as*

$$q(\gamma;\Omega) := p(\Omega + \gamma\bar{\Omega}).$$

Then $q(\cdot\,;\Omega)$ is differentiable at 0 with $\partial q(\gamma;\Omega)/\partial\gamma|_{\gamma=0} = -\langle \bar{\Omega}, \bar{M}(\Omega) \rangle$, where $\bar{M}(\Omega)$ is the optimal solution of (P) at Ω.

Proof Define the set Γ_Ω as

$$\Gamma_\Omega := \{\gamma \mid \gamma \in \operatorname{dom} q(\cdot\,;\Omega)\} = \{\gamma \mid (\Omega + \gamma\bar{\Omega}) \in \operatorname{dom} p\},$$

i.e. the set of all γ for which problem (P) has a bounded solution given the moment matrix $\Omega + \gamma\bar{\Omega}$. In order to prove the result it is then sufficient to show:

   i $0 \in \operatorname{int} \Gamma_\Omega$, and

   ii The solution of (P) at Ω is unique.

The remainder of the proof then follows from Goldfarb and Scheinberg (1999, Lemma 3.3), wherein it is shown that $0 \in \operatorname{int} \Gamma_\Omega$ implies that $-\langle \bar{\Omega}, \bar{M}(\Omega) \rangle$ is a subgradient of $q(\cdot\,;\Omega)$ at 0, and subsequent remarks in Goldfarb and Scheinberg (1999) establish that uniqueness of the solution $M(\Omega)$ ensure differentiability.

We will now show that both of the conditions (i) and (ii) above are satisfied.

(i): Proof that $0 \in \operatorname{int} \Gamma_\Omega$:

It is well-known that if both of the primal and dual problems (P) and (13) are strictly feasible then their optimal values coincide, i.e. Slater's condition holds and we obtain strong duality; see (Boyd and Vandenberghe, 2004, Section 5.2.3) and (Ramana et al., 1997).

For (P) it is obvious that one can construct a strictly feasible point. For (13), $Y_i = \Omega/(k+1)$ defines a strictly feasible point for any $\Omega \succ 0$. Hence (P) is solvable for any $\Omega + \gamma\bar{\Omega}$ with γ sufficiently small. As a result, $0 \in \operatorname{int} \Gamma$.

(ii): Proof that the solution to (P) at Ω is unique:

We will begin by showing that the range of the dual variables $\mathcal{R}(Y_i), i = 0,\ldots k$ remain the same for every primal-dual solution. Assume that there exists another optimal primal-dual pair denoted by $\tilde{M} = \bar{M} + \delta M$, and $\tilde{Y}_i$. Due to Lemma 6, there exist $\bar{y}_i, \tilde{y}_i \in \mathbb{R}^{k+1}$ such that

$$\bar{Y}_i = \bar{y}_i\bar{y}_i^T, \quad \tilde{Y}_i = \tilde{y}_i\tilde{y}_i^T \quad \forall i = 0,\ldots,k. \tag{18}$$

Obviously, $\tilde{y}_i \in \mathcal{R}(\tilde{Y}_i) = \mathcal{N}(\tilde{M} - C_i)$ and, by definition, we have

$$\tilde{y}_i^T(\tilde{M} - C_i)\tilde{y}_i = 0 \quad i = 0,\ldots,k. \tag{19}$$

Moreover, as $\bar{M}$ is feasible we have $\tilde{y}_i^T(\bar{M} - C_i)\tilde{y}_i \leq 0$, resulting in

$$\tilde{y}_i^T \delta M \tilde{y}_i \geq 0, \quad i = 0,\ldots,k. \tag{20}$$

Since $\bar{M}$ and $\tilde{M}$ have the same objective value we conclude that $\langle \Omega, \delta M \rangle = 0$. Moreover, according to (17) and (18) we can decompose Ω as $\sum_{i=0}^{k} \tilde{y}_i \tilde{y}_i^T$. Hence

$$\text{tr}(\Omega \delta M) = 0 \implies \text{tr}(\delta M \sum_{i=0}^{k} \tilde{y}_i \tilde{y}_i^T) = 0 \implies \sum_{i=0}^{k} \text{tr}(\delta M \tilde{y}_i \tilde{y}_i^T) = 0$$

$$\implies \sum_{i=0}^{k} \tilde{y}_i^T \delta M \tilde{y}_i = 0 \overset{(20)}{\implies} \tilde{y}_i^T \delta M \tilde{y}_i = 0 \qquad \forall i = 0, \ldots, k$$

$$\overset{(19)}{\implies} \tilde{y}_i (\bar{M} - C_i) \tilde{y}_i^T = 0 \qquad \forall i = 0, \ldots, k.$$

Hence, $\tilde{y}_i$ is, like $\bar{y}_i$, a null vector of $\bar{M} - C_i$. Since the null space of $\bar{M} - C_i$ is of rank one, we get $\tilde{y}_i = \lambda_i \bar{y}_i$ for some $\lambda_i \in \mathbb{R}$, resulting in, $\bar{Y}_i = \lambda_i^2 \tilde{Y}_i$.

Now we can show that the dual solution is unique. Assume that $\bar{Y}_i \neq \tilde{Y}_i$, i.e. $\lambda_i^2 \neq 1$ for some $i = \{1, \ldots, k\}$. Feasibility of $\tilde{Y}_i$ and $\bar{Y}_i$ gives

$$\left. \begin{array}{c} \sum_{i=0}^{k} \bar{Y}_i = \Omega \Leftrightarrow \sum_{i=0}^{k} \lambda_i^2 \tilde{Y}_i = \Omega \\ \sum_{i=0}^{k} \tilde{Y}_i = \Omega \end{array} \right| \Rightarrow \sum_{i=0}^{k} (1 - \lambda_i^2) \tilde{Y}_i = 0,$$

i.e. $\{\mathcal{R}(\tilde{Y}_i)\}$ are linearly dependent. This contradicts Lemma 6 and (17) which suggest linear independence, as each $\mathcal{R}(Y_i)$ is of rank one and together they span the whole space $\mathbb{R}^{k+1}$. Hence, $\bar{Y}_i = \tilde{Y}_i \ \forall i = \{0, \ldots, k\}$, i.e. the dual solution is unique.

Finally, the uniqueness of the primal solution can be established by the uniqueness for the dual solution. Indeed, summing (16) gives

$$\sum_{i=0}^{k} \bar{Y}_i (\bar{M} - C_i) = 0 \overset{(17)}{\Leftrightarrow} \Omega \bar{M} = \sum_{i=0}^{k} \bar{Y}_i C_i \Leftrightarrow \bar{M} = \Omega^{-1} \sum_{i=0}^{k} \bar{Y}_i C_i.$$

$\blacksquare$

Proof of Theorem 2:

Given the preceding support results of this section, we are now in a position to prove Theorem 2.

First, we will show that $p : \mathbb{S}_{++}^{k+1} \mapsto \mathbb{R}$ is differentiable on its domain. First, note that p is concave due to (Rockafellar and Wets, 2009, Corollary 3.32) and hence continuous on $\text{int}(\text{dom } p) = S_{++}^{k+1}$ (Rockafellar and Wets, 2009, Theorem 2.35). Also, note that due to Lemma 7 the regular directional derivatives (Rockafellar and Wets, 2009, Theorem 8.22) of p exist and are a linear map of the direction. Hence, according to (Rockafellar and Wets, 2009, Theorem 9.18 (a, f)) p is differentiable on $\mathbb{S}_{++}^{k+1}$.

Consider now the derivative of the solution of (P) when perturbing Ω across a specific direction $\bar{\Omega}$, i.e. $\partial q(\gamma; \Omega)/\partial \gamma$ with $q(\gamma; \Omega) = p(\Omega + \gamma \bar{\Omega})$. Lemma 7 shows that $\partial q(\gamma; \Omega)/\partial \gamma|_0 = -\langle \bar{\Omega}, \bar{M} \rangle$ when $\Omega \succ 0$. The proof then follows element-wise from Lemma 7 by choosing $\bar{\Omega}$ a sparse symmetric matrix with $\bar{\Omega}_{(i,j)} = \bar{\Omega}_{(j,i)} = 1/2$ the only nonzero elements. $\blacksquare$

Appendix C. Derivative of the Optimal Solution

In this section we will provide a constructive proof of Theorem 3, and show in particular that $\dot{\bar{M}}$, the directional derivative of $\bar{M}(\Omega)$ when perturbing Ω linearly across a direction $\bar{\Omega} \in \mathbb{S}^{k+1}$, can be computed by solution of the following linear system

$$
\begin{bmatrix}
\bar{S}_0 & & 0 & (\bar{y}_0^T \otimes I)\Pi_+ \\
& \ddots & & \vdots \\
0 & & \bar{S}_k & (\bar{y}_k^T \otimes I)\Pi_+ \\
\Pi(\bar{y}_0 \oplus \bar{y}_0) & \cdots & \Pi(\bar{y}_0 \oplus \bar{y}_0) & 0
\end{bmatrix}
\begin{bmatrix}
\dot{\bar{y}}_0 \\
\vdots \\
\dot{\bar{y}}_k \\
\text{vec}_\text{u}(\dot{\bar{M}})
\end{bmatrix}
=
\begin{bmatrix}
0 \\
\vdots \\
0 \\
\text{vec}_\text{u}(\bar{\Omega})
\end{bmatrix},
$$

where

- $\bar{S}_i = \bar{M}_i - C_i, \quad i = 0, \ldots, k$

- y_i is defined such that $\bar{Y}_i = \bar{y}_i \bar{y}_i^T$, i.e. the non-zero eigenvector of the Lagrange multiplier $\bar{Y}_i$, which was shown to be unique in Lemma (6).

- $\text{vec}(\cdot)$, is the operator that stacks the columns of a matrix to a vector, and $\text{vec}_\text{u}(\cdot)$ the operator that stacks only the upper triangular elements in a similar fashion

- Π, is the matrix that maps $\text{vec}(Z) \mapsto \text{vec}_\text{u}(Z)$ where $Z \in \mathbb{S}$, and Π^+ performing the inverse operation.

- $\otimes$ and $\oplus$ denote the Kronecker product and sum respectively.

Proof Lemma 7 of Appendix B, guarantees that the solutions of (P) and (13) are unique for any $\Omega \succ 0$. Hence, according to Freund and Jarre (2004), the directional derivatives $\dot{\bar{M}}, \dot{\bar{Y}}_i$ of $\bar{M}, \bar{Y}$ along the perturbation $\bar{\Omega}$ exist and are given as the unique solution to the following overdetermined system:

$$
\begin{aligned}
\sum_{i=0}^{k} \dot{\bar{Y}}_i &= \bar{\Omega} \\
\dot{\bar{Y}}_i \bar{S}_i - \bar{Y}_i \dot{\bar{M}} &= 0 \\
\dot{\bar{M}}, \dot{\bar{Y}}_i &\in \mathbb{S}^{k+1} \quad i = 0, \ldots, k.
\end{aligned}
\tag{21}
$$

The above linear system is over-determined, and has symmetric constraints. This renders standard solution methods, such as LU decomposition, inapplicable. Expressing the above system in a standard matrix form results in a matrix with $\mathcal{O}(k^4)$ zeros, which makes its solution very costly.

To avoid theses issues, we will exploit Lemma 6 of Appendix B to express the dual solution $\bar{Y}_i$ compactly as $\bar{Y}_i = \bar{y}_i \bar{y}_i^T$. One can choose a differentiable mapping $\bar{Y}_i(t) \mapsto \bar{y}_i(t)$, e.g. $\bar{y}_i(t) = \sqrt{\lambda_i(t)} u_i(t)$ where $\lambda_i(t)$ is the only positive eigenvalue of $\bar{Y}_i(t)$ and $u_i(t)$ its corresponding unit-norm eigenvector. Differentiability of $\bar{y}_i(t)$ comes from differentiability of $\bar{Y}_i(t), \lambda_i(t), u_i(t)$ (Kato, 1976) and positivity of $\lambda_i(t)$ due to Lemma 6. The chain rule

then applies for $\dot{\bar{Y}}_i = \dot{\bar{y}}_i \bar{y}_i^T + \bar{y}_i \dot{\bar{y}}_i^T$. Hence (21) can be expressed as

$$\sum_{i=0}^{k} \dot{\bar{y}}_i \bar{y}_i^T + \bar{y}_i \dot{\bar{y}}_i^T = \bar{\Omega} \tag{22}$$

$$(\dot{\bar{y}}_i \bar{y}_i^T + \bar{y}_i \dot{\bar{y}}_i^T)\bar{S}_i - \bar{y}_i \bar{y}_i^T \dot{\bar{M}} = 0, \quad i = 0, \ldots, k. \tag{23}$$

Exploiting $\bar{y}_i^T \bar{S}_i = 0$ from (16) and that $y_i \neq 0$ gives

$$(23) \Leftrightarrow \bar{y}_i(\dot{\bar{y}}_i^T \bar{S}_i - \bar{y}_i^T \dot{\bar{M}}) = 0 \overset{y_i \neq 0}{\Longleftrightarrow} \dot{\bar{y}}_i^T \bar{S}_i - \bar{y}_i^T \dot{\bar{M}}, \quad i = 0, \ldots, k \tag{24}$$

We can express equations (22) and (24) into the standard matrix form by using the vec operator and the identity $\text{vec}(AXB) = (B^T \otimes A)\text{vec}(X)$, which gives

$$\sum_{i=0}^{k} (\bar{y}_i \otimes I + I \otimes \bar{y}_i)\dot{\bar{y}}_i = \text{vec}(\bar{\Omega})$$

$$S_i \dot{\bar{y}}_i - (\bar{y}_i^T \otimes I)\text{vec}(\dot{\bar{M}}) = 0, \quad i = 0, \ldots, k.$$

Finally, eliminating the symmetric constraint via $\text{vec}_{\text{u}}(\cdot)$, Π and Π^+ gives:

$$\sum_{i=0}^{k} \Pi(\bar{y}_i \otimes I + I \otimes \bar{y}_i)\dot{\bar{y}}_i = \text{vec}_{\text{u}}(\bar{\Omega})$$
$$S_i \dot{\bar{y}}_i - (\bar{y}_i^T \otimes I)\Pi^+ \text{vec}_{\text{u}}(\dot{\bar{M}}) = 0, \quad i = 0, \ldots, k, \tag{25}$$

leading to the suggested linear system. The system is square and non-singular since it is equivalent to (21) which has a unique solution.

Finally, it remains to show that $M(\Omega)$ is a differentiable for any $\Omega \succ 0$. First note that M is outer semicontinuous (Rockafellar and Wets, 2009, Definition 5.4) on $\mathbb{S}_{++}^{k+1}$ as, according to Theorem (2), p is continuous on $\mathbb{S}_{++}^{k+1}$. Since M is unique on $\mathbb{S}_{++}^{k+1}$ it is also continuous. Finally, note that due to Equation (25) the regular directional derivatives (Rockafellar and Wets, 2009, Theorem 8.22) of M exist and are a linear map of the direction. Hence, according to (Rockafellar and Wets, 2009, Theorem 9.18 (a, f)) M is differentiable on $\mathbb{S}_{++}^{k+1}$. ∎

Appendix D. Construction of the worst and best-case distributions

In this appendix we construct distributions in $\mathcal{P}(\mu, \Sigma)$ that achieve the bounds of (3). First, regarding the lower bound $\inf_{\mathbb{P} \in \mathcal{P}(\mu,\Sigma)} \mathbb{E}_{\mathbb{P}}[g(\xi)]$, where g is the *convex piecewise affine function* defined by (4), the following Proposition implies that it is trivial, in the sense that it is independent of the variance Σ.

Proposition 8 *For any convex piecewise affine function* $\max_{i=1,\ldots,l}(a_i + b_i^T \xi)$ *the worst case expectation* $\inf_{\mathbb{P} \in \mathcal{P}(\mu,\Sigma)} \mathbb{E}_{\mathbb{P}}[\max_{i=1,\ldots,l}(a_i + b_i^T \xi)]$ *is equal to* $\max_{i=1,\ldots,l}(a_i + b_i^T \mu)$.

Proof First note that the worst case expectation is bounded below

$$\mathbb{E}\left[\max_{i=1,\ldots,l}(a_i + b_i^T \xi)\right] \geq \max_{i=1,\ldots,l}\mathbb{E}[a_i + b_i^T \xi] = \max_{i=1,\ldots,l}(a_i + b_i^T \mu). \tag{26}$$

We will construct a sequence of parametric distributions that, in the limit, achieve the above mentioned lower bound.

Assume the one-dimensional, uncorrelated random variables z, w

$$z \sim \mathcal{U}\left(-\frac{1}{\epsilon}, \frac{1}{\epsilon}\right), \quad w \sim \mathcal{N}(0, \epsilon),$$

i.e. z is uniformly distributed in $(-\epsilon^{-1}, \epsilon^{-1})$, and w is a zero mean Gaussian with variance ϵ, where $\epsilon \in \mathbb{R}_{++}$.

Now, assuming $0 < \epsilon \leq \sqrt{\frac{1}{3}}$, consider the random variable x with the mixture distribution

$$x = \begin{cases} z & \text{with probability } 3\epsilon^2 \\ w & \text{with probability } 1 - 3\epsilon^2. \end{cases}$$

Since both of the mixing distributions are zero mean, the resulting distribution is zero mean with variance

$$\mathbb{E}(x^2) = 3\epsilon^2\mathbb{E}(z^2) + (1 - 3\epsilon^2)\mathbb{E}(w^2) = 1 + \epsilon(1 - 3\epsilon^2).$$

In the limit $\epsilon \to 0$ the random variable x has zero mean and variance one, but its probability distribution function is infinitesimal everywhere outside the origin.

Assuming that y is a vector of independent variables distributed identically to x for $\epsilon \to 0$, the random vector $\xi = \Sigma^{1/2}y + \mu$ has covariance matrix Σ and mean value μ, with its probability distribution being infinitesimal everywhere expect in μ. For this random vector (26) is tight. ∎

Thus, according to Proposition 8 the worst case expectation $\inf_{\mathbb{P} \in \mathcal{P}(\mu,\Sigma)}\mathbb{E}_{\mathbb{P}}[g(\xi)]$ does not depend on Σ and is equal to $\underline{y^{\mathcal{D}}} - \min(\mu_1, \ldots, \mu_k, \underline{y^{\mathcal{D}}})$.

Next, we will construct a distribution that achieves the upper bound of (3). In particular we will show that for any $\Omega \succ 0$ the optimal distribution $\bar{\mathbb{P}}_{\mu,\Sigma}(\Omega)$ of problem (P) is discrete and can be given in closed form by the solution of (13), i.e. the dual of (P). To show this, we will need the following ancillary result:

Proposition 9 *For any $\Omega \succ 0$ we have $\langle \bar{Y}_i(\Omega), C_i \rangle < 0 \; \forall i = 1, \ldots, k$ where $\bar{Y}_i(\Omega)$ is the dual solution of (P).*

Proof Assume that there exists $i \in \{1, \ldots, k\}$ such that $\langle \bar{Y}_i, C_i \rangle \geq 0$. Then, the following set of dual variables

$$\begin{aligned} \tilde{Y}_0 &= \bar{Y}_0 + \bar{Y}_i \\ \tilde{Y}_i &= 0 \\ \tilde{Y}_j &= \bar{Y}_j, \quad j \in \{1, \ldots, k\} \cap j \neq i, \end{aligned} \tag{27}$$

is also optimal, as $\sum_{i=1}^{k}\langle \tilde{Y}_i, C_i \rangle = \sum_{i \neq j}\langle \bar{Y}_i, C_i \rangle \leq \sum_{i=1}^{k}\langle \bar{Y}_i, C_i \rangle$. This is a contradiction to Lemma (7). ∎

The following Theorem will now construct the best-case distribution:

Theorem 10 *For any $\Omega \in \mathbb{S}_{++}^{k+1}$ the probability distribution $\bar{\mathbb{P}}(\Omega)$ that minimizes (12) is discrete with probability mass function*

$$f_y(\xi) = \begin{cases} p_i, & \xi = \xi_i, \qquad i = 0, \dots, k \\ 0, & \text{otherwise,} \end{cases} \tag{28}$$

where the points ξ_i and the probability masses p_i are uniquely defined by the solution of (13) and $\Omega = \begin{bmatrix} \Sigma + \mu\mu^T & \mu \\ \mu^T & 1 \end{bmatrix}$ as following

$$\bar{Y}_i = p_i \begin{bmatrix} \xi_i \\ 1 \end{bmatrix} \begin{bmatrix} \xi_i^T & 1 \end{bmatrix}, \quad i = 1, \dots k$$

$$p_0 = 1 - \sum_{i=1}^{k} p_i \quad \xi_0 = \mu - \sum_{i=1}^{k} p_i \xi_i.$$

Proof Consider the dual of (11) defined in (12) which can be reformulated as following:

$$\begin{aligned} -\sup \quad & \langle \Omega, M \rangle \\ \text{s.t.} \quad & \begin{bmatrix} \xi^T & 1 \end{bmatrix} (M - C_i) \begin{bmatrix} \xi^T & 1 \end{bmatrix}^T \leq 0 \\ & \forall \xi \in \mathbb{R}^k, \quad i = 0, \dots k. \end{aligned} \tag{29}$$

According to Lemma 6, $\mathcal{N}(\bar{M} - C_i) = 1 = \mathcal{R}(\bar{Y}_i)$, $i = 0, \dots, k$. Hence, at optimality, the infinite dimensional constraint of (29) is active in at most $k + 1$ points. Due to slack complementarity between the primal-dual pair (11)-(12) these are the only points for which the optimal probability measure of (11) is nonzero. As a result, the optimal distribution $\bar{P}_{\mu,\Sigma}$ is discrete with at most $k + 1$ points.

Denote as ξ_i the possible outcomes of the optimal probability distribution $\bar{P}_{\mu,\Sigma}$, with ξ_i corresponding to the i-th constraint of (28). Each ξ_i has a nonzero probability if and only if there exists a (unique) $\begin{bmatrix} \xi_i^T & 1 \end{bmatrix} \in \mathcal{N}(\bar{M} - C_i) = \mathcal{R}(\bar{Y}_i)$. This is always the case for $i = 1, \dots, k$ as otherwise the last row and column of Y_i would be zero and, due to the special structure of C_i, we would get $\langle \bar{Y}_i, C_i \rangle = 0$ contradicting Proposition 27. Decomposing $\bar{Y}_i$ as

$$\bar{Y}_i = p_i \begin{bmatrix} \xi_i \\ 1 \end{bmatrix} \begin{bmatrix} \xi_i^T & 1 \end{bmatrix}, \quad i = 1, \dots k$$

results in the following dual optimal cost

$$-\sum_{i=1}^{k} \langle \bar{Y}_i, C_i \rangle = -\sum_{i=1}^{k} p_i (\xi_{i(i)} - \underline{y_d}),$$

from which we can verify that the suggested probability distribution (28) achieves the optimal cost. Uniqueness of the probability distribution comes from uniqueness and linear independence of $\bar{Y}_i$ (Lemma (7)).

Finally, it is worth noting that a similar result that concerns a broader class of distributionally robust optimization problems is given in (Van Parys, B. P.G., 2015, Theorem 5.1) which shows that the optimal distribution $\bar{\mathbb{P}}_{\mu,\Sigma}$ has a discrete support. However, (Van Parys, B. P.G., 2015) does not show how to construct the optimal distribution nor does he prove that it consists of exactly $k + 1$ points. ■

References

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

R. H. Byrd, J. Nocedal, and R. A. Waltz. *Knitro: An Integrated Package for Nonlinear Optimization*, pages 35–59. Springer US, Boston, MA, 2006. ISBN 978-0-387-30065-8.

C. Chevalier and D. Ginsbourger. *Fast Computation of the Multi-Points Expected Improvement with Applications in Batch Selection*, pages 59–69. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-44973-4.

M. P. Deisenroth, D. Fox, and C. E. Rasmussen. Gaussian Processes for Data-Efficient Learning in Robotics and Control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):408–423, 2015. ISSN 0162-8828.

T. Desautels, Krause A., and J. W. Burdick. Parallelizing Exploration-Exploitation Trade-offs in Gaussian Process Bandit Optimization. *Journal of Machine Learning Research*, 15:4053–4103, 2014. URL http://jmlr.org/papers/v15/desautels14a.html.

P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, Z. Sidor, and Y. Wu. OpenAI Baselines. https://github.com/openai/baselines, 2017.

R. Fletcher. *Practical Methods of Optimization; (Second Edition)*. Wiley-Interscience, New York, NY, USA, 1987. ISBN 0-471-91547-5.

R. W. Freund and F. Jarre. A Sensitivity Result for Semidefinite Programs. *Oper. Res. Lett.*, 32(2):126–132, 2004. ISSN 0167-6377.

M. Garstka, M. Cannon, and P. J. Goulart. COSMO: A conic operator splitting method for convex conic problems. arXiv 1901.10887, 2019.

A. Genz and F. Bretz. *Computation of Multivariate Normal and T Probabilities*. Springer Publishing Company, Incorporated, 1st edition, 2009. ISBN 364201688X, 9783642016882.

D. Ginsbourger, R. Le Riche, L. Carraro, and D. Mi. A multi-points criterion for deterministic parallel global optimization based on gaussian processes. *Journal of Global Optimization, in revision*, 2009.

D. Ginsbourger, R. Le Riche, and L. Carraro. *Kriging Is Well-Suited to Parallelize Optimization*, pages 131–162. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-10701-6.

D. Goldfarb and K. Scheinberg. On parametric semidefinite programming. *Applied Numerical Mathematics*, 29(3):361 – 377, 1999. ISSN 0168-9274.

J. Gonzalez, Z. Dai, P. Hennig, and N. Lawrence. Batch Bayesian Optimization via Local Penalization. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 648–657, Cadiz, Spain, 2016a. PMLR. URL http://proceedings.mlr.press/v51/gonzalez16a.html.

J. Gonzalez, M. A. Osborne, and N. D. Lawrence. GLASSES : Relieving The Myopia Of Bayesian Optimisation. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2016b. URL `http://arxiv.org/pdf/1510.06299v1.pdf`.

D. R. Jones. A Taxonomy of Global Optimization Methods Based on Response Surfaces. *Journal of Global Optimization*, 21(4):345–383, 2001. ISSN 1573-2916.

T. Kato. *Perturbation theory for linear operators; 2nd ed.* Grundlehren Math. Wiss. Springer, Berlin, 1976. URL `https://cds.cern.ch/record/101545`.

D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans. Automatic Gait Optimization with Gaussian Process Regression. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 944–949, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

S. Marmin, C. Chevalier, and D. Ginsbourger. *Differentiating the Multipoint Expected Improvement for Optimal Batch Design*, pages 37–48. Springer International Publishing, Cham, 2015. ISBN 978-3-319-27926-8.

ApS MOSEK. The MOSEK optimization toolbox for Python manual. URL `http://www.mosek.com/`.

B O'Donoghue, E. Chu, N. Parikh, and S. Boyd. Conic Optimization via Operator Splitting and Homogeneous Self-Dual Embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, 2016a.

B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd. SCS: Splitting Conic Solver, version 1.2.6. `https://github.com/cvxgrp/scs`, 2016b.

V. Picheny, T. Wagner, and D. Ginsbourger. A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization*, 48(3):607–626, 2013. ISSN 1615-1488.

M. V. Ramana, L. Tunçel, and H. Wolkowicz. Strong duality for semidefinite programming. *SIAM Journal on Optimization*, 7(3):641–662, 1997.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. ISBN 026218253X.

R. T. Rockafellar and R. J.B. Wets. *Variational Analysis*, volume 317. Springer Science & Business Media, 2009.

N. Rontsis, P. J. Goulart, and Y. Nakatsukasa. Efficient Semidefinite Programming with approximate ADMM. *arXiv preprint arXiv:1912.02767*, 2019.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal Policy Optimization Algorithms. *ArXiv e-prints*, 2017.

A. Shah and Z. Ghahramani. Parallel Predictive Entropy Search for Batch Global Optimization of Expensive Objective Functions. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 3330–3338. Curran Associates, Inc., 2015.

A. Shah, A. G. Wilson, and Z. Ghahramani. Student-t Processes as Alternatives to Gaussian Processes. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, pages 877–885, 2014. URL `http://jmlr.org/proceedings/papers/v33/shah14.html`.

B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE*, 104(1): 148–175, 2016. ISSN 0018-9219.

J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian Optimization of Machine Learning Algorithms. In *Advances in Neural Information Processing Systems*, pages 2951–2959, 2012.

B. P.G. Van Parys, P. J. Goulart, and M. Morari. Distributionally robust expectation inequalities for structured distributions. *Optimization-Online e-prints*, 2015. URL `http://www.optimization-online.org/DB_HTML/2015/05/4896.html`.

Van Parys, B. P.G. *Distributionally robust control and optimization*. PhD thesis, ETH Zurich, 2015.

L. Vandenberghe and S. Boyd. Semidefinite Programming. *SIAM Review*, 38(1):49–95, 1996. ISSN 00361445. URL `http://www.jstor.org/stable/2132974`.

Y. Zheng, G. Fantuzzi, A. Papachristodoulou, P. Goulart, and A. Wynn. Chordal decomposition in operator-splitting methods for sparse semidefinite programs. 2017. URL `https://arxiv.org/abs/1707.05058`.

S. Zymler, D. Kuhn, and B Rustem. Distributionally robust joint chance constraints with second-order moment information. *Mathematical Programming*, 137(1):167–198, 2013. ISSN 1436-4646.