Pareto Frontier Learning with Expensive Correlated Objectives

Amar Shah AS793@CAM.AC.UK
Zoubin Ghahramani ZOUBIN@ENG.CAM.AC.UK
Machine Learning Group, Department of Engineering, University of Cambridge

Abstract

There has been a surge of research interest in developing tools and analysis for *Bayesian optimization*, the task of finding the global maximizer of an unknown, expensive function through sequential evaluation using Bayesian decision theory. However, many interesting problems involve optimizing *multiple*, expensive to evaluate objectives simultaneously, and relatively little research has addressed this setting from a Bayesian theoretic standpoint. A prevailing choice when tackling this problem, is to model the multiple objectives as being independent, typically for ease of computation. In practice, objectives are *correlated* to some extent. In this work, we incorporate the modelling of inter-task correlations, developing an approximation to overcome intractable integrals. We illustrate the power of modelling dependencies between objectives on a range of synthetic and real world multi-objective optimization problems.

1. Introduction

Most engineering problems require making design choices which aim to simultaneously optimize multiple objectives. For example, in designing a new drug, a pharmaceutical scientist may strive to simultaneously maximize the likelihood of curing an illness, minimize the chance of unwanted side-effects and minimize the cost of drug development. Typically there would not exist a particular option for which each objective is fully optimized. Subsequently, the scientist would wish to consider a range of options which trade off the multiple objectives. The ultimate choice should be *Pareto optimal*; there should not exist an alternative option which can improve on the chosen option in every objective simultaneously.

The objective functions are often *unknown*, and can only be ascertained through pointwise evaluation. However, it can be *expensive* to evaluate these objectives, in the sense that they may necessitate large computational, economical or other resource. The challenge is to find a set of Pareto optimal points in as few sequential evaluations of the multiple objective functions as possible, so as to minimize the total expense.

A Bayesian theoretic approach to this task would be to probabilistically model the multiple unknown objective functions. Where the function should be evaluated next is decided by maximizing the expected value of a chosen *acquisition*, or utility function, based on the posterior distribution of the objective functions given evaluations. In the single objective scenario, this approach is called *Bayesian optimization* (Mockus, 1989), and has been successful in a variety of difficult, expensive global optimization tasks including drug discovery (Negoescu et al., 2011) and robot gait control (Lizotte et al., 2007).

Two key choices must be made under this Bayesian theoretic framework: (i) a choice of prior over the objective functions and (ii) a choice of acquisition function which instructs where the functions should be evaluated next. Gaussian processes (Rasmussen and Williams, 2006) are the popular choice for modelling objective functions, since they are nonparameteric and permit analytic calculations. Other models have also shown promising results in Bayesian optimization e.g. Student-t processes (Shah et al., 2014) and deep neural networks (Snoek et al., 2015). A commonly used acquisition function is the *expected improvement* (Mockus et al., 1978), and alternative options are discussed in the next section.

Whilst Bayesian optimization techniques have been developed to decide on multiple locations in which a single objective should be evaluated next (Contal et al., 2013; Shah and Ghahramani, 2015), relatively little research has focussed on Bayesian approaches to decide on where to evaluate multiple objectives next (Picheny, 2014; Hernández-Lobato et al., 2016). This is in part due to it

Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48. Copyright 2016 by the author(s).

being more difficult to find a set of Pareto optimal points for multiple objectives than it is to find an optimizer of a single function. Furthermore, it is not entirely clear how one can quantitatively evaluate the quality of a proposed *Pareto frontier*, or set of Pareto optimal points. Zitzler (1999) considers a volume based measure of the Pareto frontier, whose expected increase is possible to compute analytically under the assumption of independently distributed Gaussian process objectives (Emmerich et al., 2008). However, in practice, objectives are *correlated*. Returning to the drug discovery example, we would expect the cost of drug development to be high when the chance of unwanted side effects is high, and incorporating this belief should affect our decision process.

The key contribution of this work is to include the modelling of correlations amongst objective functions using multi-output Gaussian process priors. To the best of our knowledge, no approach in the literature has yet considered a Bayesian approach to modelling and optimizing multiple correlated objectives. We introduce an approximation to an intractable multidimensional integral, which results in an elegant deterministic and differentiable approximation to the expected increase in volume. Empirical evidence suggests that our approach is beneficial on a range of multi-objective optimization tasks.

In Section 2, we formalize notation and discuss the notion of Pareto hypervolume. Next, in Section 3, we describe how Gaussian processes may be used to model the objective functions and derive our approximation which is used to determine where to evaluate the correlated objectives next. Finally, we run our algorithm and several comparisons on synthetic and real objective functions.

2. Hypervolume Based Pareto Learning

In this section, we discuss how the quality of a Pareto set can be measured quantitatively, and review a previously proposed method to directly improve this measure. We first introduce notation to formalize the problem.

2.1. Pareto Efficiency and Hypervolume

Our aim is to jointly maximize $L \geq 2$ bounded objectives $f_l : \mathcal{X} \to \mathbb{R}$ for $l = 1, ..., L$. Concretely, we wish to find a set of *Pareto efficient* points. Given distinct $\boldsymbol{y}_i \in \mathbb{R}^L$ for $i = 1, ..., n$, we write $\boldsymbol{y}_j \succeq \boldsymbol{y}_k$ when $y_{j,l} \geq y_{k,l}$ for each $l = 1, ..., L$, and say "$\boldsymbol{y}_j$ *dominates* $\boldsymbol{y}_k$". For the set of distinct points $\mathcal{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_n\}$, the subset of Pareto efficient points, $\mathcal{P}(\mathcal{Y}) \subseteq \mathcal{Y}$, is defined as $\mathcal{P}(\mathcal{Y}) = \{\boldsymbol{y}_i \in \mathcal{Y} : \boldsymbol{y}_j \not\succeq \boldsymbol{y}_i \forall \boldsymbol{y}_j \in \mathcal{Y} \backslash \{\boldsymbol{y}_i\}\}$. In other words, the Pareto efficient set is the set of non-dominated points, and is always non empty. A dominated point, by definition, is a suboptimal choice since there exists a point which achieves a higher value for each of the L objectives.

In a Bayesian optimization setting, input locations $x_1, x_2, ... \in \mathcal{X}$, at which the expensive objective functions are evaluated, are chosen sequentially. Given function evaluations $\boldsymbol{y}_s = [f_1(x_s), ..., f_L(x_s)]^\top$ for $s = 1, ..., t$, $x_{t+1} \in \mathcal{X}$ is chosen with the goal of improving the set of Pareto points as much as possible with as few future function evaluations. However, in its current frame, this is a qualitative goal and not a quantitative formulation. In the case of single objective Bayesian optimization of a function $f : \mathcal{X} \to \mathbb{R}$, a common approach is to maximize a future *reward*, $r_T = [f(\tilde{x}_T) - f(x^*)]$, where $x^\star \in \mathrm{argmax}_{x \in \mathcal{X}} f(x)$ is an optimal input and $\tilde{x}_T$ is our guess of where the maximizer of f is after evaluating f at T input locations (Brochu et al., 2009). How can the quantitative single objective framework be generalized to the multi objective case? The *Pareto hypervolume* is an appropriate measure of the quality of a set of Pareto efficient points (Zitzler, 1999).

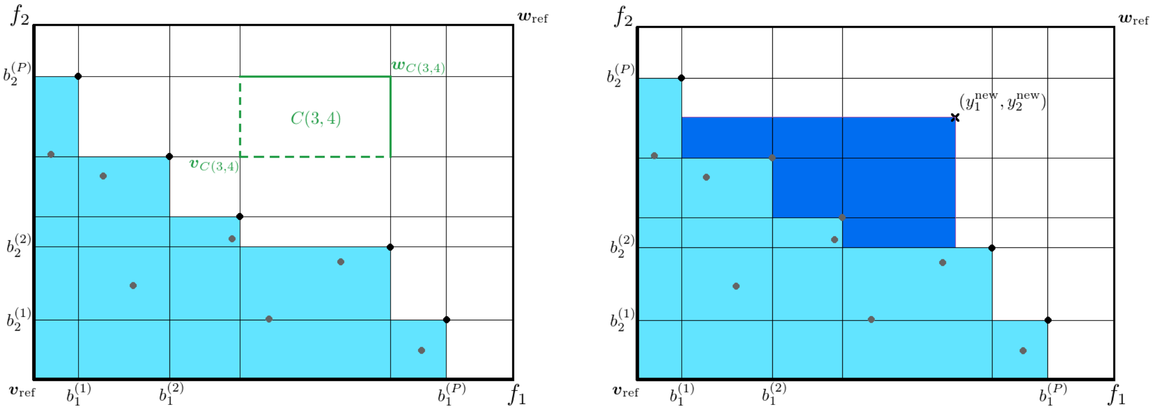
Given a set of distinct points $\mathcal{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_n\}$, we have defined its Pareto efficient subset, $\mathcal{P}(\mathcal{Y})$. Define a reference point, $\boldsymbol{v}_{\mathrm{ref}} \in \mathbb{R}^L$, which is dominated by each element of $\mathcal{P}(\mathcal{Y})$ i.e. $\boldsymbol{u} \succeq \boldsymbol{v}_{\mathrm{ref}}$ for each $\boldsymbol{u} \in \mathcal{P}(\mathcal{Y})$. The Pareto hypervolume of $\mathcal{P}(\mathcal{Y})$ with respect to $\boldsymbol{v}_{\mathrm{ref}}$ is

$$\begin{aligned} &\mathrm{Vol}_{\boldsymbol{v}_{\mathrm{ref}}}\big(\mathcal{P}(\mathcal{Y})\big) \qquad (1) \\ &\quad = \int_{\mathbb{R}^L} \mathbb{I}\big[\boldsymbol{y} \succeq \boldsymbol{v}_{\mathrm{ref}}\big] \Big[1 - \prod_{\boldsymbol{u} \in \mathcal{P}(\mathcal{Y})} \mathbb{I}\big[\boldsymbol{u} \not\succeq \boldsymbol{y}\big]\Big] \, d\boldsymbol{y} \end{aligned}$$

where $\mathbb{I}(.)$ is the indicator function, which outputs 1 if its argument is true and 0 otherwise. $\mathrm{Vol}_{\boldsymbol{v}_{\mathrm{ref}}}\big(\mathcal{P}(\mathcal{Y})\big)$ measures the volume of points in $\mathbb{R}^L$ which dominate $\boldsymbol{v}_{\mathrm{ref}}$ but are dominated by at least one element of the Pareto set, $\mathcal{P}(\mathcal{Y})$. The shaded region of Figure 1(a) illustrates this volume. The Pareto hypervolume is a monotone function since $\mathrm{Vol}_{\boldsymbol{v}_{\mathrm{ref}}}\big(\mathcal{P}\big(\mathcal{Y} \cup \{\boldsymbol{y}\}\big)\big) \geq \mathrm{Vol}_{\boldsymbol{v}_{\mathrm{ref}}}\big(\mathcal{P}(\mathcal{Y})\big)$.

The more dominant the set of Pareto points, the larger the Pareto hypervolume. Conversely, a marginally dominant set of Pareto points will have a small Pareto hypervolume. This makes the Pareto hypervolume a reasonable measure of how "good" a proposed set of Pareto efficient points is.

Note that the units of the hypervolume measure is the product of the units of each of the objectives, f_l. Whilst the scale of units does not affect performance in most commonly used single objective Bayesian optimization algorithms, the relative scales of objectives $f_1, ..., f_L$ will affect the hypervolume measure that we propose here.



(a) Pareto hypervolume

(b) Increase in Pareto hypervolume

Figure 1. Two objective example illustrating Pareto optimality, Pareto hypervolume and our notation. (a) Observations $(f_1(x_i), f_2(x_i))$ for $i = 1, ..., 12$ shown by dots, with the dark dots representing the set of Pareto efficient observations. Grey dots each have at least one dark dot to the top-right of it i.e. they are dominated. f_1 values are shown on the x-axis and f_2 values on the y-axis. The Pareto efficient points induce a grid cell partitioning of the relevant region which has bottom left corner, $\boldsymbol{v}_{\text{ref}}$, and top right corner, $\boldsymbol{w}_{\text{ref}}$. Cell $C(3,4)$ defines the cuboid from $\boldsymbol{v}_{C(3,4)} \equiv \left(b_1^{(3)}, b_2^{(4)}\right)$ to $\boldsymbol{w}_{C(3,4)} \equiv \left(b_1^{(4)}, b_2^{(5)}\right)$. The volume of the shaded region, in which points are dominated by at least one Pareto efficient point, represents the Pareto hypervolume with respect to reference point $\boldsymbol{v}_{\text{ref}}$. (b) Change in Pareto frontier when a new observation is made at input location x^{new} with value $\left(f_1(x^{\text{new}}), f_2(x^{\text{new}})\right) = \left(y_1^{\text{new}}, y_2^{\text{new}}\right)$. The new observation dominates 2 points which were previously Pareto optimal. The consequent increase in Pareto hypervolume is equal to the volume of the darker shaded region. The darker shaded region is the sum of cuboidal volumes over the previously non-dominated cells in $\mathcal{C}_{\text{nd}}$. Note that had the new observation $\left(y_1^{\text{new}}, y_2^{\text{new}}\right)$ been in the lightly shaded region, it would have been dominated, the set of Pareto efficient points would not have changed and hence the increase in Pareto hypervolume would have been 0.

2.2. Expected Improvement in Pareto Hypervolume

Analogous to the single objective case, we can formulate a multi objective Bayesian optimization problem as maximizing a future reward, $r_T = \left[\text{Vol}_{\boldsymbol{v}_{\text{ref}}}\left(\mathcal{P}(\tilde{\mathcal{Y}}_T)\right) - \text{Vol}_{\boldsymbol{v}_{\text{ref}}}\left(\mathcal{P}(\mathcal{Y}^*)\right)\right]$, where $\mathcal{Y}^*$ is the true Pareto frontier and $\tilde{\mathcal{Y}}_T$ is the suggested Pareto frontier after T evaluations of each of the objectives.

Computing the expected effect of a decision made at time step $t = 1$ on a regret at time $T \gg 1$ is computationally infeasible in the Bayesian optimization setting (Osborne et al., 2009). A common greedy, but computationally feasible alternative is to repeatedly maximize the expected one step ahead reward, examples of which include expected improvement, probability of improvement (Kushner, 1964), upper confidence bound (Cox and John, 1992) and entropy search (Hennig and Schuler, 2012).

Emmerich (2005) introduced the idea of *expected improvement in Pareto hypervolume*, defined as

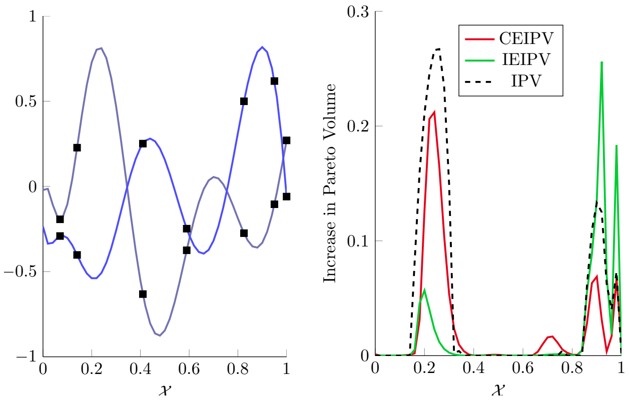
$$\text{EIPV}(x_{t+1}|\mathcal{D}) = \tag{2}$$

$$\mathbb{E}_{p(\boldsymbol{y}(x_{t+1})|\mathcal{D})}\left[\text{Vol}\Big(\mathcal{P}\big(\mathcal{Y} \cup \{\boldsymbol{y}(x_{t+1})\}\big)\Big) - \text{Vol}\Big(\mathcal{P}(\mathcal{Y})\Big)\right]$$

where $\boldsymbol{y}_s = [f_1(x_s), ..., f_L(x_s)]^\top$, $\mathcal{Y} = \{\boldsymbol{y}_s\}_{s=1}^t$, $\mathcal{D} = \{x_s, \boldsymbol{y}_s\}_{s=1}^t$ and $\boldsymbol{v}_{\text{ref}}$ is dropped for convenience.

Given that each f_l is bounded above, we choose a reference point $\boldsymbol{w}_{\text{ref}}$, such that $\boldsymbol{w}_{\text{ref}} \succeq [f_1(x), ..., f_L(x)]^\top$ for any $x \in \mathcal{X}$ (it is ossible to set $w_{\text{ref},l} = \infty$). The cuboidal set of interest becomes $\mathbb{A} \equiv \{\boldsymbol{y} \in \mathbb{R}^L : \boldsymbol{w}_{\text{ref}} \succeq \boldsymbol{y} \succeq \boldsymbol{v}_{\text{ref}}\}$. Let the Pareto efficient subset be $\mathcal{P}(\mathcal{Y}) = \{\boldsymbol{u}_1, ..., \boldsymbol{u}_P\}$ for $1 \leq P \leq t$, and set $\boldsymbol{u}_0 = \boldsymbol{v}_{\text{ref}}$ and $\boldsymbol{u}_{P+1} = \boldsymbol{w}_{\text{ref}}$. Now let $b_j^{(0)} \leq ... \leq b_j^{(P+1)}$ be the sorted list of j^{th} coordinates of $\boldsymbol{u}_0, ..., \boldsymbol{u}_{P+1}$. The grid coordinates $b_j^{(p)}$ induce a cuboidal partitioning of $\mathbb{A}$. Specifically, for each $(i_1, ..., i_L) \in \{0, ..., P\}^L$, we define the grid cell $C(i_1, ..., i_L)$ as the cuboid $\left(b_1^{(i_1)}, b_1^{(i_1+1)}\right] \times \left(b_2^{(i_2)}, b_2^{(i_2+1)}\right] \times ... \times \left(b_L^{(i_L)}, b_L^{(i_L+1)}\right]$, then grid cells are disjoint and their union equals $\mathbb{A}$. Analogous to the definitions of $\boldsymbol{v}_{\text{ref}}$ and $\boldsymbol{w}_{\text{ref}}$ with respect to $\mathbb{A}$, we define $\boldsymbol{v}_{C(i_1,...,i_L)} = \left(b_1^{(i_1)}, ..., b_L^{(i_L)}\right)^\top$ and $\boldsymbol{w}_{C(i_1,...,i_L)} = \left(b_1^{(i_1+1)}, ..., b_L^{(i_L+1)}\right)^\top$. Hence for any $\boldsymbol{y} \in C(i_1, ..., i_L)$, $\boldsymbol{w}_{C(i_1,...,i_L)} \succeq \boldsymbol{y} \succeq \boldsymbol{v}_{C(i_1,...,i_L)}$. The set of grid cells is defined as $\mathcal{C} \equiv \left\{C(i_1, ..., i_L) : (i_1, ..., i_L) \in \{0, ..., P\}^L\right\}$. An example of this grid partitioning with $L = 2$ objectives is shown in Figure 1(a).

Note that there are two key types of grid cells: those whose points are dominated by at least one member of the Pareto efficient set and those whose points are not dominated by any member of the Pareto efficient set



(a) Synthetic objectives (b) Inrease in PV at $x \in \mathcal{X}$

Figure 2. An illustration that modelling correlations amongst objectives is beneficial. (a) Two objectives on $\mathcal{X} = [0, 1]$ with observed function values shown by black squares. (b) Plots of CEIPV, IEIPV and IPV at new input locations. IPV is the actual increase in Pareto volume. CEIPV better matches the true IPV, as it is able to model negative correlation between objectives.

(shaded differently in Figure 1(b)). Subsequent new observations amongst the non-dominated cells change the Pareto frontier, whilst observations in the dominated cells do not. We define the set of non-dominated cells as $\mathcal{C}_{\text{nd}} \equiv \{C \in \mathcal{C} : \forall \boldsymbol{y} \in C, \boldsymbol{u} \in \mathcal{P}(\mathcal{Y}), \boldsymbol{u} \not\succeq \boldsymbol{y}\}$.

With the notation, definitions and illustrations developed, the increase in Pareto volume from a new observation, $\boldsymbol{y}^{\text{new}}$, is given by $\sum_{C\in\mathcal{C}_{\text{nd}}} \text{Vol}_{\boldsymbol{v}_C}\left(\{\boldsymbol{y}^{\text{new}}\}\right)$ (see the supplementary material for a formal derivation). This is the volume of points which were previously non-dominated, but are rendered dominated by $\boldsymbol{y}^{\text{new}}$. Consequently, the expected increase in Pareto volume is

$$\begin{aligned} \text{EIPV}(x|\mathcal{D}) &= \sum_{C\in\mathcal{C}_{\text{nd}}} \int_{C} \text{Vol}_{\boldsymbol{v}_C}\left(\{\boldsymbol{y}\}\right) p(\boldsymbol{y}|\mathcal{D})\ d\boldsymbol{y} \\ &= \sum_{C\in\mathcal{C}_{\text{nd}}} \int_{\boldsymbol{v}_C}^{\boldsymbol{w}_C} \prod_{l=1}^{L} (y_l - v_{C,l}) p(\boldsymbol{y}|\mathcal{D})\ d\boldsymbol{y}. \quad (3) \end{aligned}$$

We have developed an acquisition function to decide where multiple objectives should be evaluated next in pursuit of finding a Pareto frontier. Next, we shall discuss various Gaussian process based measures for $p(\boldsymbol{y}|\mathcal{D})$.

3. Pareto Learning with Gaussian Processes

Our setting is one in which evaluating objectives $f_1, ..., f_L$ is expensive and we therefore would like to learn as much as possible per set of evaluations. Modelling objectives accurately and quantifying uncertainty about predictions are both key to deciding where we should evaluate next. Gaussian processes are ideal for modelling the objectives, as they are nonparametric, provide uncertainty estimates about function values and often permit analytically tractable inference. We review how independent GP models on each f_l lead to an analytic expression for EIPV, and introduce a novel analytic approximate of EIPV when we model the f_l as correlated.

3.1. Independent Gaussian Process Objectives

Emmerich et al. (2008) show that in the case that $f_1, ..., f_L$ are independent Gaussian process draws, the expected improvement in Pareto hypervolume can be calculated analytically. We denote the EIPV under independent GP objectives, as IEIPV and compute it below,

$$\begin{aligned} \text{IEIPV}(x|\mathcal{D}) &= \sum_{C\in\mathcal{C}_{\text{nd}}} \int_{\boldsymbol{v}_C}^{\boldsymbol{w}_C} \prod_{l=1}^{L} (y_l - v_{C,l}) p(\boldsymbol{y}|\mathcal{D})\ d\boldsymbol{y} \\ &= \sum_{C\in\mathcal{C}_{\text{nd}}} \prod_{l=1}^{L} \int_{v_{C,l}}^{w_{C,l}} (y_l - v_{C,l}) \phi\Big(\frac{y_l - \mu_l}{\sigma_l}\Big)\ dy_l \\ &= \sum_{C\in\mathcal{C}_{\text{nd}}} \prod_{l=1}^{L} \sigma_l^2 \Big[\big(\phi(\beta_{C,l}) - \phi(\alpha_{C,l})\big) \\ &\qquad + \beta_l \big(\Phi(\beta_{C,l}) - \Phi(\alpha_{C,l})\big) \Big], \quad (4) \end{aligned}$$

where $f_l(x)|\mathcal{D} \sim \mathcal{N}\big(f_l(x); \mu_l, \sigma_l^2\big)$, $\alpha_{C,l} = (w_{C,l} - \mu_l)/\sigma_l$, $\beta_{C,l} = (v_{C,l} - \mu_l)/\sigma_l$, and ϕ and Φ are the standard Gaussian pdf and cdf respectively. Under the assumption of independent Gaussian process objectives, not only is the EIPV analytically computable, its derivative is too. This is achieved by computing the derivatives of μ_l and σ_l with respect to the input location x and simple applications of the chain and product rule. The assumption of independence of objectives is crucial in the above derivation as it allows us to write an integral of a product as a product of univariate simple integrals. This is a luxury which is not enjoyed when the objectives are modelled as being correlated.

3.2. Correlated Gaussian Process Objectives

Denote the integral over cell, C, in equation (3) as $\Psi_C(x) \equiv \int_{\boldsymbol{v}_C}^{\boldsymbol{w}_C} \prod_{l=1}^{L} (y_l(x) - v_{C,l}) p(\boldsymbol{y}(x)|\mathcal{D}) d\boldsymbol{y}$. Modelling the objectives f_l as correlated Gaussian processes would lead to a posterior $\boldsymbol{f}(x)|\mathcal{D} \sim \mathcal{N}\big(\boldsymbol{f}(x); \boldsymbol{\mu}, \boldsymbol{\Sigma}\big)$, where $\boldsymbol{\Sigma}$ is non-diagonal. Whilst univariate Gaussian integrals are often analytically computable, the opposite is true for general multivariate Gaussians. Under a correlated

GP, the integral Ψ_C is no longer tractable. Note that

$$\Psi_C(x) = \int_{-\infty}^{\infty} \prod_{l=1}^{L} (y_l - v_{C,l}) \mathbb{I}\big[v_{C,l} < y_l \le w_{C,l}\big] p(\boldsymbol{y}|\mathcal{D}) d\boldsymbol{y}. \tag{5}$$

Define the form of the expression inside the product of equation (5) as $h(y) \equiv (y - v)\mathbb{I}[v < y \le w]$. Our approach is to approximate $h(y)$ with a scaled Gaussian probability density function, $\tilde{h}(y) = z\mathcal{N}(y; \lambda, \tau^2)$, where we set z, λ, τ to moment match $h(y)$ as follows

$$\begin{aligned} z &= \int_{-\infty}^{\infty} h(y) dy &&= \frac{1}{2}(w - v)^2 \\ \lambda &= z^{-1} \int_{-\infty}^{\infty} y h(y) dy &&= \frac{1}{2}(2w + v) \\ \tau^2 &= z^{-1} \int_{-\infty}^{\infty} (y - \lambda)^2 h(y) dy &&= \frac{1}{18}(w - v)^2. \end{aligned} \tag{6}$$

Note that the approximation parameters z, λ, τ do not depend on the input, x. The nature of our approximation is similar to that made in expectation propagation (Minka, 2001). The important difference is that expectation propagation requires approximation parameters to be learned in order to well approximate the entire integral Ψ_C, whilst our approach simply aims to well approximate the integrand, $h(y)$. Whilst the expectation propagation approach is more appropriate for our task statistically speaking, the parameters it would learn would strongly depend on x, which makes EP computationally infeasible. EP requires relearning the approximation parameters at each new x location whilst our approximation does not require this. Incorporating our proposed approximation strategy results in the following analytic expression for an approximation to Ψ_C,

$$\begin{aligned} \Psi_C \approx \tilde{\Psi}_C &\equiv \int_{-\infty}^{\infty} \prod_{l=1}^{L} z_{C,l} \mathcal{N}(y_l; \lambda_{C,l}, \tau_{C,l}^2) \mathcal{N}(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\, d\boldsymbol{y} \\ &= \prod_{l=1}^{L} z_{C,l} \int_{-\infty}^{\infty} \mathcal{N}\big(\boldsymbol{y}; \mathrm{diag}(\boldsymbol{\lambda}_C), \mathrm{diag}(\boldsymbol{\tau}_C^2)\big) \mathcal{N}(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\, d\boldsymbol{y} \\ &= \prod_{l=1}^{L} z_{C,l} \times \exp\Bigg(-\frac{1}{2}\Big(\boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu} + \log\det(\boldsymbol{\Sigma})\Big) \\ &\qquad + \frac{1}{2}\Big(\boldsymbol{\nu}_C^\top \boldsymbol{\Omega}_C^{-1} \boldsymbol{\nu}_C + \log\det(\boldsymbol{\Omega}_C)\Big) \\ &\qquad - \frac{1}{2}\sum_{l=1}^{L}\bigg(\frac{\lambda_{C,l}^2}{\tau_{C,l}^2} + \log\big(2\pi\tau_{C,l}^2\big)\bigg)\Bigg), \end{aligned} \tag{7}$$

where $\boldsymbol{\Omega}_C^{-1} = \boldsymbol{\Sigma}^{-1} + \mathrm{diag}\big(\boldsymbol{\tau}_C^2\big)^{-1}$ and $\boldsymbol{\Omega}_C^{-1}\boldsymbol{\nu}_C = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \mathrm{diag}\big(\boldsymbol{\tau}_C^2\big)^{-1}\boldsymbol{\lambda}_C$. The final equality comes from the fact that the product of multivariate Gaussian probability density functions leads to a scaled multivariate Gaussian probability density function, which can be integrated analytically (see the supplementary material for a derivation). We denote the approximated expected improvement in Pareto hypervolume under the correlated objective case as $\mathrm{CEIPV}(x|\mathcal{D}) = \sum_{C \in \mathcal{C}_{\mathrm{nd}}} \tilde{\Psi}_C(x)$.

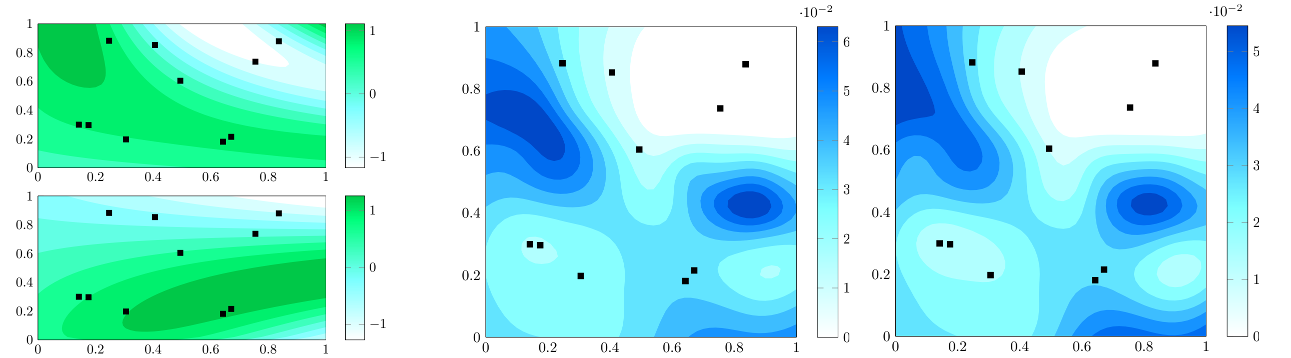
Since the parameters $\boldsymbol{z}_C, \boldsymbol{\lambda}_C, \boldsymbol{\tau}_C$ do not depend on input location, x, we compute $\partial\tilde{\Psi}_C/\partial x$ by computing $\partial\boldsymbol{\mu}/\partial x$ and $\partial\boldsymbol{\Sigma}/\partial x$, repeated applications of the chain and product rules, and use of matrix derivative rules.

3.3. Correlated Gaussian Process Models

Until now, in this section, we have assumed a correlated Gaussian process model without specifying its form. In this subsection we propose two correlated output Gaussian process models to use in the CEIPV framework.

Multi-task GPs. Bonilla et al. (2008) developed a framework to model correlated functions on the same input domain, $\mathcal{X}$. The idea involves a Kronecker factorization of the covariance between $f_l(x)$ and $f_{l'}(x')$, separating the intra-task covariance matrix from the inter-task covariance. Specifically, $\mathrm{Cov}(f_l(x), f_{l'}(x')) = \mathrm{K}_{l,l'} k(x, x')$, where $\mathbf{K}$ is a positive semi-definite matrix specifying inter-task similarities and k is a covariance function over $\mathcal{X}$. Suppose for inputs $x_1, ..., x_n$, $\mathbf{G}$ is such that $\mathrm{G}_{i,j} = k(x_i, x_j)$, then the full covariance matrix across tasks and data points is $\mathbf{K} \otimes \mathbf{G}$, where $\otimes$ represents a Kronecker product. A benefit of this approach, is that matrix inversion costs only $O(L^3 + n^3)$ since $\big[\mathbf{K} \otimes \mathbf{G}\big]^{-1} = \mathbf{K}^{-1} \otimes \mathbf{G}^{-1}$. However, a potential downside is that each function is marginally identically distributed up to scaling, which may be a poor assumption for multiple objective Pareto frontier learning. Swersky et al. (2013) utilize this model for Bayesian optimization of single objectives to transfer knowledge from previously solved similar optimization problems.

Semiparametric Latent Factor GP. In this framework introduced by Teh et al. (2004), the idea is to take linear combinations of nonparametric models. In our context, we define the covariance between $f_l(x)$ and $f_{l'}(x')$, as $\mathrm{Cov}(f_l(x), f_{l'}(x')) = \sum_{s=1}^{L} \mathrm{A}_{l,s}\mathrm{A}_{l',s} k_s(x, x')$, where $\mathbf{A}$ is lower triangular, and k_s are covariance functions over $\mathcal{X}$. Here, each function is no longer marginally identically distributed, but the downside is that matrix inversion now has complexity $O(L^3 n^3)$. In typical settings this cost would be prohibitive, but note that under a Bayesian optimization framework this would not be a problem. We will be interested in up to 3 objective functions to optimize, and n is typically small, of the order of 100.



(a) Synthetic objective functions (b) $\mathrm{EIPV}(x)$ (c) $\mathrm{CEIPV}(x)$

Figure 3. We empirically assess the quality of the CEIPV approximation to a numerical integration based estimate of EIPV on a 2 objective problem, under the Semiparametric Latent Factor GPs model. (a) Synthetic objective functions $f_1, f_2 : [0, 1]^2 \to \mathbb{R}$. Dark regions correspond to high function values and faint regions correspond to low function values. Black squares correspond to input locations of 10 function evaluations, which are identical for both functions. (b) Ground truth $\mathrm{EIPV}(x)$, where each $\Psi_C(x)$ is computed using numerical integration over $\mathbb{R}^2$. (c) Our approximation, $\mathrm{CEIPV}(x)$. Dark regions correspond to input locations, $x \in \mathbb{R}$, with high utility, whilst faint regions correspond to inputs with low utility in terms of where to evaluate the objectives next.

4. Experiments

In this section, we provide empirical comparisons assessing the performance of the proposed CEIPV method. We denote the CEIPV framework under the semiparamteric latent factor GPs and multi-task models, as CEIPV-SLF and CEIPV-MT respectively. Three algorithms are used for comparison: IEIPV, ParEGO (Knowles, 2006) and Random. ParEGO is a method, which, at each iteration, defines a single objective function by taking a random convex combination of the multiple objectives, and maximizing the expected improvement under the pseudo single objective to decide where to evaluate all of the objectives next. Random simply picks a point in $\mathcal{X}$ to evaluate all the objectives at next, uniformly at random. Our experiments assume the input space, $\mathcal{X}$, is a convex subset of $\mathbb{R}^D$.

In line with Snoek et al. (2012), we choose to use ARD Matérn 5/2 kernels over the input space, defined as

$$k_{M52}(\boldsymbol{x}, \boldsymbol{x}') = \theta_0^2 \left(1 + \sqrt{5r^2} + \frac{5}{3}r^2\right) \exp\left(-\sqrt{5r^2}\right)$$

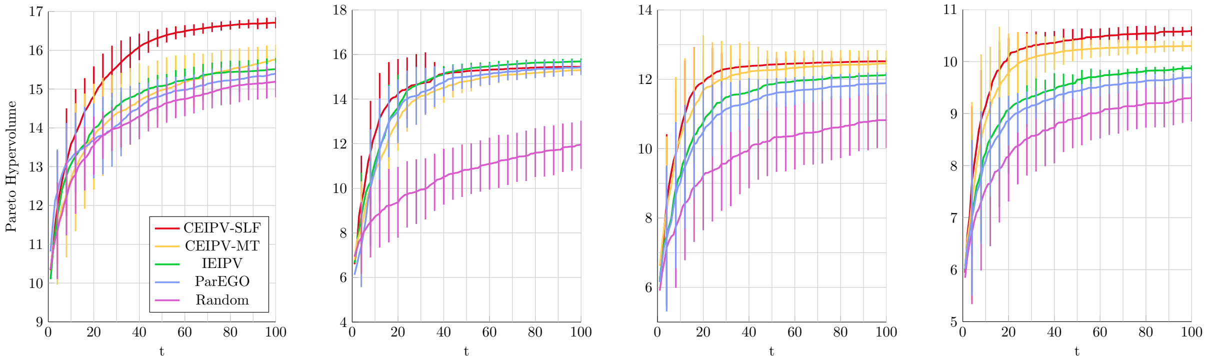
$$r^2 = \sum_{d=1}^{D} (x_d - x'_d)^2 / \theta_d^2.$$

For the CEIPV algorithms, the amplitude hyperparameter, θ_0, is set to 1 to avoid over parameterization.

In many applications, observed values are corrupted with noise. In this work, we assume each objective is observed with its own form of Gaussian noise, such that $y_l(\boldsymbol{x}) = f_l(\boldsymbol{x}) + \epsilon_l(\boldsymbol{x})$, where $\epsilon_l(\boldsymbol{x}) \sim \mathcal{N}(0, \sigma_l^2)$ independently. All of the previous derivations remain possible with the assumption of additive Gaussian noise, because a sum of Gaussians is also Gaussian distributed.

To perform a fully Bayesian treatment of the hyperparameters, we place priors over and sample them from their joint posterior given observed data using slice sampling (Neal, 2003). Independent log-Gaussian priors are placed over $\boldsymbol{\theta}$ and $\boldsymbol{\sigma}$. In the case of CEIPV-MT, we parameterize the $L \times L$ inter-task covariance matrix $\mathbf{K}$ as $\mathbf{A}\mathbf{A}^\top$, where $\mathbf{A}$ is lower triangular. A lower triangular matrix is also used in CEIPV-SLF. For both algorithms, we place Gaussian priors on the lower triangular entries of $\mathbf{A}$. Our first experiment shows the benefit of modelling cross objective correlations, and that the CEIPV method is able to capture these correlations. For illustrative purposes, we limit the input space to $[0, 1]$, and generate two negatively correlated objective functions, f_1 and f_2 which we jointly wish to maximize. Given noise free observations of both objectives at 7 input locations, we compared $\mathrm{CEIPV\text{-}SLF}(x|\mathcal{D})$, $\mathrm{IEIPV}(x|\mathcal{D})$ and the actual increase in Pareto volume from a new evaluation at x given the data and full knowledge of the objective functions, $\mathrm{IPV}(x|\mathcal{D}, f_1, f_2)$. See Figure 2. Notice CEIPV does a much better job of modelling IPV than IEIPV does. Whilst the independent GP method is fooled by the high function values for $x \in [0.8, 1]$, the correlated GP model learns the strong negative correlation, and uses this to recommend that the next evaluation be in the interval $x \in [0.2, 0.3]$.

Once convinced that modelling correlations amongst objective functions is beneficial, we wished to assess the quality of the CEIPV approximation to the true integral, EIPV, under the the SLF model. We again consider 2 objective functions, $f_1, f_2 : [0, 1]^2 \to \mathbb{R}$ with 10 noiseless



(a) $\rho = -0.5$ (b) $\rho = 0$ (c) $\rho = +0.25$ (d) $\rho = +0.75$

Figure 4. Performance of various algorithms in maximizing Pareto hypervolume on 2 objectives on $\mathcal{X} = [0, 1]^3$, synthetically generated from the Semiparameteric Latent Factor GPs with correlations (a) $\rho = -0.5$, (b) $\rho = 0$, (c) $\rho = +0.25$ and (d) $\rho = +0.75$. Experiments for each algorithm are repeated 50 times, the mean perfomances plus-minus one standard deviation are plotted.

function observations (Figure 3(a)). Under a SLF GP model, we compute an estimate of EIPV (equation (3)) using numerical integration, and CEIPV-SLF, at new input locations $\boldsymbol{x} \in [0, 1]^2$ (Figures 3(b),(c)). Notice that the contours of CEIPV-SLF are on the whole very similar to those of EIPV, suggesting that the CEIPV approximation is a decent one. There is a small amount of discrepancy is in the region $[0, 0.2] \times [0.8, 1]$, where CEIPV is slightly more inclined to explore than IEPV, something we noticed in further experiments. The approximation quality appears best in regions close to observed input locations.

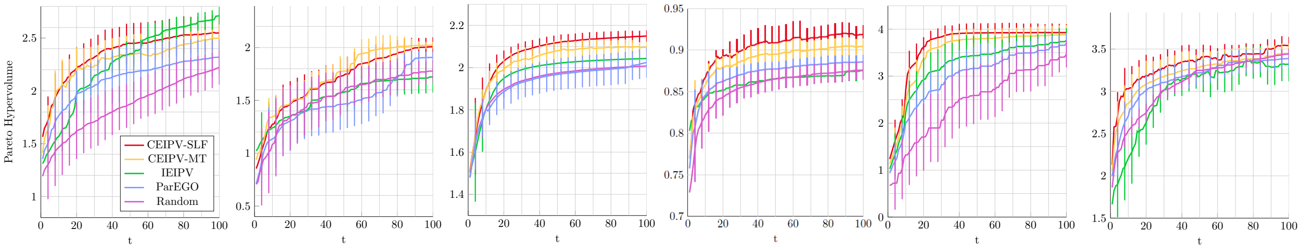
For the remaining experiments, we report $\mathrm{Vol}_{\boldsymbol{v}_{\mathrm{ref}}}\left(\mathcal{P}(\tilde{\mathcal{Y}}_t)\right)$ at every iteration, t. Each experiment is initialized with function evaluations at 5 input locations sampled independently and uniformly at random over the input space. To assess performance with different initial samples, we repeat each experiment 50 times with different initial input points. At each iteration, we average the acquisition function over 10 hyperparameter samples.

Next, we assess the performance of various algorithms on problems with different correlation levels. Setting the input space to $[0, 1]^3$, we generate pairs of objectives, f_1, f_2, with means 2 and -2 respectively. Objectives are drawn using the SLF model with two Matérn kernels with lengthscales $\boldsymbol{\theta}^{(1)} = [0.7, 0.4, 1]$ and $\boldsymbol{\theta}^{(2)} = [0.34, 0.9, 0.5]$. The matrix $\mathbf{A}$ is set such that $\mathrm{A}_{1,1} = 1, \mathrm{A}_{2,1} = \rho$ and $\mathrm{A}_{2,2} = \sqrt{1 - \rho^2}$. We generate pairs of objectives for (a) $\rho = -0.5$, (b) $\rho = 0$, (c) $\rho = +0.25$ and (d) $\rho = +0.75$. The various algorithms are run attempting to find Pareto efficient frontiers, with the results displayed in Figure 4. The higher the absolute value of correlation, the more the CEIPV methods outperform other methods. For $\rho = 0$ there is so statistical difference between between IEIPV and CEIPV models as we would expect. ParEGO also performs just as well for this level of correlation. Each of the 4 model based methods outperforms Random. These experiments provide strong evidence that the CEIPV are able to account for correlations amongst objectives, and that this can lead to superior performance.

While Gaussian process based multi-objective optimization has not been thoroughly explored, there is a long history of, typically model-free based, approaches to this problem, including evolutionary and genetic algorithms (Coello et al., 2002; Zitzler and Thiele, 1999). Several benchmark functions have been constructed for testing the efficacy of optimization algorithms. We choose three such functions for experimentation: `oka2` (Okabe et al., 2004), `vlmop3` (Veldhuizen and Lamont, 1999) and `dtlz1a` (Deb et al., 2001). We define these functions in the supplementary material.

Results for the experiments on these functions are shown in Figure 5(a),(b),(c). The objectives being experimented on have complicated correlations between them, which depend on the location of the input space. Nonetheless, the CEIPV methods perform consistently strongly in all experiments. IEIPV does well on the `oka2` task despite a slow start. We believe this was due to f_1 being simple to model, where a correlated multi-task GP may be prone to consider more complicated explanations than necessary.

Finally we consider three real-world multi objective Pareto frontier optimization problems. The first problem, `boston`, involves training a 2 hidden layer neural network on a random train/test split of the Boston Housing dataset (Bache and Lichman, 2013). The function takes as input the weight-decay parameter, number of training iterations and size of the hidden layers. The outputs are the negative prediction error on the test set, and the negative product of the layer size and number of training iterations. The idea is



(a) `oka2` (b) `vlmop3` (c) `dtlk1a` (d) `boston` (e) `rocket` (f) `llvm`

Figure 5. Performance of multi-objective optimization algorithms on synthetic and real functions. Experiments for each algorithm are repeated 50 times, the mean perfomances plus-minus one standard deviation are plotted.

to explore the trade off between (i) accuracy of prediction, and (ii) a combined measure of memory consumption and training time of the neural network. Such a trade off would be useful to explore, for example in the case of storing neural network models on mobile devices with limited memory and computational power. A manufacturer of such devices would be interested in knowing the form of the Pareto curve as this could influence final design choices.

Next we consider `rocket`, a simulation of a rocket (Hasbun, 2008) being launched from the Earth's surface. The mass of fuel used, launch height and launch angle relative to the ground are inputs to the simulation. The outputs are the time taken to return to the Earth's surface, the angular distance travelled with respect to the centre of the Earth, and the absolute difference between the launch angle and the radius at the point of launch. Simulations are often used in engineering to explore what outcomes of various design choices may be for example in aerospace and automobile engineering. Nevertheless, running simulations can take days and require vast computational resources, making intelligent experiment choice crucial.

Thirdly we consider the problem of optimizing compiler settings for the LLVM compiler, using the SW-LLVM data set of Siegmund et al. (2012). The design space consists of 1023 different compiler settings, determined by 10 binary flags. The objectives are memory footprint and performance on a given set of software programs, compiled with the particular compiler settings. The data was very costly to obtain; evaluating the objectives on a particular compiler setting takes several hours. We denote the problem `llvm`, and set the input space to $[0, 1]^{10}$, using a rounding function to determine the binary indicators.

Results on each of the three real world tasks are shown in Figure 5(d),(e),(f). As before, the CEIPV algorithms which incorporate correlations between objectives tend to perform best, by a statistically significant margin in most cases. Most interestingly, the IEIPV model performs marginally worse than random selection on the `llvm` task, whilst the CEIPV methods do significantly better than random. On the `boston` problem, the CEIPV methods appear to achieve results in 10 iterations, which takes ParEGO about 100 iterations, suggesting that our approximation and correlation modelling significantly boosts the rate at which we approach Pareto optimality.

We also ran a popular evolutionary approach to multi-objective Pareto optimization, NSGA-II (Deb et al., 2002), on each of these tasks. On each and every task, the performance of NSGA-II was marginally below that of IEIPV at every iteration. In order to avoid further clutter, we decided not to plot the performance in Figure 5. All EIPV methods outperform NSGA-II, verifying the findings of Couckuyt et al. (2014).

5. Discussion

In this paper, we argue that modelling correlations amongst objectives in multi-objective Pareto optimization problems is important for success. To overcome the problem of intractable integrals, we devise a novel approximation which leads to an analytic and differentiable approximation to the expected increase in Pareto hypervolume acquisition function. Two forms of correlated output GP models are implemented on a variety of multi-objective problems, and seem to consistently outperform competing models which model objectives as being independent.

There are several directions in which this work may be extended further. Further theoretical analysis is required to assess the nature of the approximation we have made. In order to reduce computational burden under the SLF model, one may consider implementing a sparse approximation to the $NL \times NL$ covariance matrix which leads to faster computation. Next, we could work on how to select a batch of points where we can evaluate next in parallel, in a multi-objective setting. Another interesting avenue would be to experiment with alternative correlated output models, such as a deep neural network.

References

K. Bache and M. Lichman. UCI Machine Learning Repository, 2013.

E. V. Bonilla, K. M. A. Chai, and C. K. I. Williams. Multi-task Gaussian Process Prediction. *NIPS*, 2008.

E. Brochu, M. Cora, and N. de Freitas. A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Applications to Active User Modeling and Hierarchical Reinforcement Learning. Technical Report TR-2009-23, University of British Columbia, 2009.

C. A. C. Coello, D. A. Van Veldhuizen, and G. B. Lamont. *Evolutionary algorithms for solving multi-objective problems*, volume 242. Springer, 2002.

E. Contal, D. Buffoni, D. Robicquet, and N. Vayatis. Parallel Gaussian Process Optimization with Upper Confidence Bound and Pure Exploration. In *Machine Learning and Knowledge Discovery in Databases*, pages 225–240. Springer Berlin Heidelberg, 2013.

I. Couckuyt, D. Deschrijver, and T. Dhaene. Fast calculation of multiobjective probability of improvement and expected improvement criteria for pareto optimization. *Journal of Global Optimization*, 60(3):575–594, 2014.

D. D. Cox and S. John. A statistical method for global optimization. *Proc. IEEE Conference on Systems*, 1992.

K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable Test Problems for Evolutionary Multi-Objective Optimization. Technical Report 112, TIK lab, ETH, 2001.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197, 2002.

M. Emmerich. *Single- and Multiobjective Evolutionary Design Optimization Using Gaussian Random Field Metamodels*. PhD thesis, FB Informatik, University of Dortmund, 2005.

M. Emmerich, A. Deutz, and J. W. Klinkenberg. The computation of the expected improvement in dominated hypervolume of Pareto front. Technical Report LIACS TR-4-2008, Leiden University, 2008.

J. E. Hasbun. In *Classical Mechanics with MATLAB Applications*. Jones & Bartlett Learning, 2008.

P. Hennig and C. J. Schuler. Entropy Search for Information-Efficient Global Optimization. *JMLR*, 2012.

D. Hernández-Lobato, J. M. Hernández-Lobato, A. Shah, and R. P. Adams. Predictive Entropy Search for Multi-objective Bayesian Optimization. *ICML*, 2016.

J. Knowles. ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Tran. Evol. Comput.*, 2006.

H. J. Kushner. A new method of locating the maximum of arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering*, 1964.

D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans. Automatic Gait Optimization with Gaussian Process Regression. *IJCAI*, pages 944–949, 2007.

T. P. Minka. *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, Masachusetts Institute of Technology, 2001.

J. Mockus. *Bayesian Approach to Global Optimization: Theory and Applications*. Kluwer, 1989.

J. Mockus, V. Tiesis, and A. Zilinskas. The Application of Bayesian Method for Seeking the Extremum. In *Toward Global Optimization*, pages 117–128. Elsevier, 1978.

R. M. Neal. Slice sampling. *Annals of Statistics*, 2003.

D. M. Negoescu, P. I. Frazier, and W. B. Powell. The Knowledge-Gradient Algorithm for Sequencing Experiments in Drug Discovery. *INFORMS Journal on Computing*, 23(3):346–363, 2011.

T. Okabe, Y. Jin, M. Olhofer, and B. Sendhoff. On test functions for evolutionary multiobjecive optimization. *Parallel Problem Solving from Nature*, 2004.

M. A. Osborne, R. Garnett, and S. J. Roberts. Gaussian Processes for Global Optimization. *Int'l Conf on Learning and Intelligent Optimization*, 2009.

V. Picheny. Multiobjective optimization using gaussian process emulators via stepwise uncertainty reduction. *Statistics and Computing*, 2014.

C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

A. Shah and Z. Ghahramani. Parallel Predictive Entropy Search for Batch Global Optimization of Expensive Objective Functions. *NIPS*, 2015.

A. Shah, A. G. Wilson, and Z. Ghahramani. Student-t Processes as Alternatives to Gaussian Processes. *AISTATS*, 2014.

N. Siegmund, S. S. Kolesnikov, C. Kastner, S. Apel, D. Batory, M. Rosenmuller, and G. Saake. Predicting Performance via Automated Feature-Interaction Detection. *Int'l Conf. on Software Engineering*, 2012.

J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian Optimization of Machine Learning Algorithms. *NIPS*, 2012.

J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. Patwary, Mr Prabat, and R. P. Adams. Scalable Bayesian Optimization Using Deep Neural Networks. *ICML*, 2015.

K. Swersky, J. Snoek, and R. P. Adams. Multi-task Bayesian Optimization. 2013.

Y. W. Teh, M. Seeger, and M. I. Jordan. Semiparametric Latent Factor Models. *NIPS*, 2004.

D. A. V. Veldhuizen and G. B. Lamont. Multiobjective Evolutionary Algorithm Test Suites. *ACM Symposium on Applied Computing*, 1999.

E. Zitzler. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, ETH Zurich, 1999.

E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE trans. Evol. Comput.*, 3(4):257–271, 1999.