Pareto Front Modeling for Sensitivity Analysis in Multi-Objective Bayesian Optimization

Roberto Calandra **Jan Peters**[*]
Intelligent Autonomous Systems Lab
Technische Universität Darmstadt

Marc Peter Deisenroth
Department of Computing
Imperial College London

Abstract

Many real-world applications require the optimization of multiple conflicting criteria. For example, in robot locomotion, we are interested is maximizing speed while minimizing energy consumption. Multi-objective Bayesian optimization (MOBO) methods, such as ParEGO [6], ExI [5] and SMS-EGO [8] make use of models to define the next experiment, i.e., select the next parameters for which the objective function is to be evaluated. However, suggesting the next experiment is typically the only use of models in MOBO. In this paper, we propose to further exploit these models to improve the estimation of the final Pareto front and ultimately provide useful tool to the user for further analysis. We demonstrate that a small philosophical difference leads to substantial advantages in the practicality of most MOBO methods "for free".

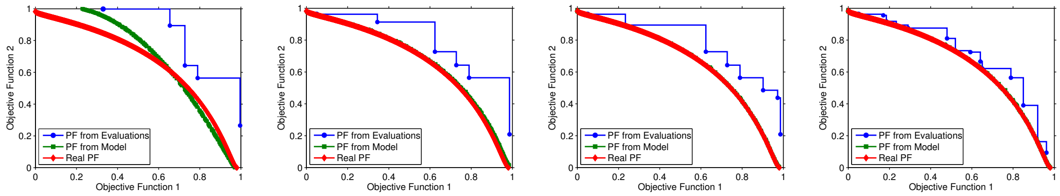
Optimization often requires the definition of a single objective function to be optimized. However, many real-world applications naturally present multiple criteria to be optimized. For example, in complex robotic systems, we must consider performance criteria such as motion accuracy, speed, robustness to noise or energy-efficiency [9]. Typically, it is impossible to optimize all these desiderata at the same time as they may be conflicting. However, it is still desirable to find a trade-off that satisfies as much as possible the different criteria and the necessities of the final user.

For practical purposes, the existence of multiple criteria is often side-stepped by designing a single objective that incorporates all criteria, e.g., by defining a weighted sum of the criteria [3]. Alternatively, the optimization of these different objectives can be formalized as a **Multi-Objective Optimization** (MOO) [2]. In MOO, the goal is to return a Pareto front (PF), which represents the best trade-off possible between the different criteria [7]. From this Pareto front, it is the responsibility of the user to ultimately select the most convenient/promising set of parameters to apply. Intuitively, the goodness of the returned PF can be measured by the accuracy (how close is the proposed PF to the real unknown *optimal* PF), by the size (having a large set of solutions in the PF is desirable) and by the diversity (having solutions that encompass a wide range of trade-offs).

Many model-based MOO methods exist, which extend Bayesian optimization to the multi-objective case, such as ParEGO [6], ExI [5] and SMS-EGO [8]. We here define all these methods as Multi Objective Bayesian Optimization (MOBO) methods. Currently, the main advantage of MOBO methods, compared to model-free MOO methods [4, 12], is to reduce the number of experiments/ evaluations of the objective function. However, the models of the objective functions are used exclusively to select the next parameters to evaluate.

In this paper, we present a new perspective on MOBO and on the use of models in MOO. We demonstrate that it is possible to exploit the learned models, e.g., for better approximations of the Pareto front or for computing additional statistics, such as robustness w.r.t. noise in the parameters. These additional properties can be used by most MOBO methods without significant cost.

[*]Also with Max Planck Institute for Intelligent Systems, Tübingen, Germany



(a) 20 Evaluations (b) 50 Evaluations (c) 100 Evaluations (d) 200 Evaluations

Figure 1: **Pareto front (PF) approximation from response surface.** Each plot shows the PF from the evaluated randomly chosen parameters (blue curve), the PF from the response surface learned using the evaluated parameters (green curve) and the real PF (red curve). (a) Using only a few evaluations leads to a poor approximation of the real PF. Using the response surface learned from the same evaluations already improves the quality of the approximated PF. (b,c,d) With more evaluations, the response surface accurately approximates the real PF. In contrast, the PF from the evaluations is still a poor, sparse and discontinuous approximation.

		Hypervolume		
		from measurements	from learned model	ground truth
Number of evaluations	20	0.123	0.305	0.342
	50	0.183	0.330	
	100	0.202	0.338	
	200	0.273	0.338	
	1000	0.317	0.338	

Table 1: **Pareto front approximation from response surface.** Hypervolume of the MOP2 function using different numbers of random evaluations. Comparison of the hypervolume computed only from the evaluations and from a model learned from these evaluations. The model allows for a better approximation of the real Pareto front and, therefore, consistently increases the hypervolume.

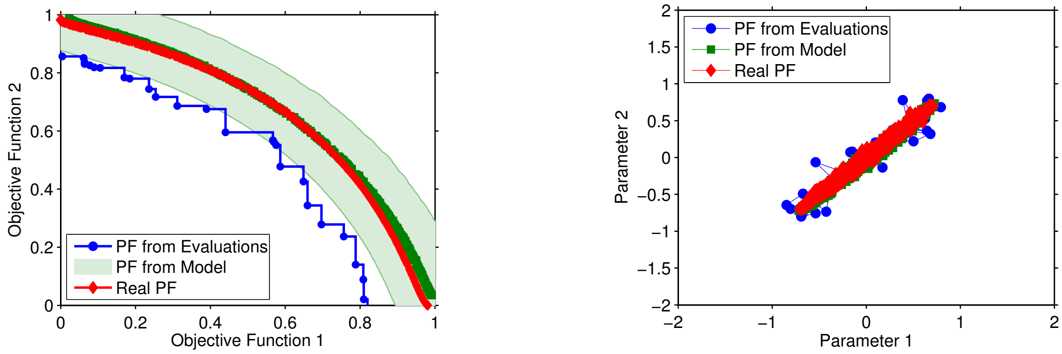
1 Exploiting the Objective Functions Models

In this section, we show how to exploit three different aspects of model-based MOO methods to specifically suit the demands of robotic applications: a) We analyze the use of models to create a dense Pareto front to assist the human decision maker for choosing parameters. b) We show that a model-based approach can substantially reduce the negative effects of measurement noise during the identification of the Pareto front. c) We demonstrate the usefulness of our novel sensitivity measure as a diagnostic tool in the light of the final human decision.

We consider the MOP2 function [10], a standard test function for MOO, and we set the number of parameters $d = 2$, so that both the parameter and objective function spaces can be easily visualized. As performance measure, we use the hypervolume contribution [11] (also called $\mathcal{S}$-metric or Lebesgue measure), a common performance measure for MOO tasks. Finally, to compute the hypervolume of the MOP2 function, we use the standard reference point $r = [1, 1]$.

1.1 Modeling the Pareto Front

We now demonstrate how the use of models (independent of any optimization process) can exploit past evaluations to provide an *a posteriori* dense and smooth Pareto front approximation from previous evaluations/experiments. We sample from a uniform distribution 20, 50, 100 and 200 parameters, respectively, and evaluate them on the MOP2 test function. As shown in Figure 1, from such a small number of evaluations the resulting approximation of the real Pareto front (blue curve) is poor, discontinuous and sparse. However, it is possible to learn a response surface from which to sample a much larger number of "virtual" evaluations (i.e., without performing additional evaluations on the real function). From the "virtual" evaluations of the response surface we compute a dense Pareto front approximation (green curve), which approximates the ground-truth Pareto front (red curve) better. The same conclusion is reached by analyzing the hypervolume measure in Table 1, where the response surface approximates the ground truth of the hypervolume better. Note that the quality of the Pareto front from the response surface depends on the past evaluations. Hence, the approximation is typically inaccurate when only few evaluations are available.



(a) Objective function space

(b) Parameter space

Figure 2: **MOO with measurement noise.** (a) Estimating the real Pareto front (red curve) exclusively from noisy evaluations leads to an over-optimistic PF (blue curve) and, hence sub-optimal parameters (blue dots). With a response surface, we obtain a better approximation of the PF (green curve) and estimate the 95% confidence of the noise (green area). (b) The parameters from the response surface (green dots) closely resemble the parameters of the real PF (red dots).

Hypervolume		
from noisy measurements	from learned model	ground truth
0.502	0.326	0.342

Table 2: **MOO with measurement noise.** Measurement noise leads to poor approximations of the Pareto front and to overestimation of the hypervolume. Using the learned model substantially improves the approximation of the Pareto front and results in a near-optimal hypervolume.

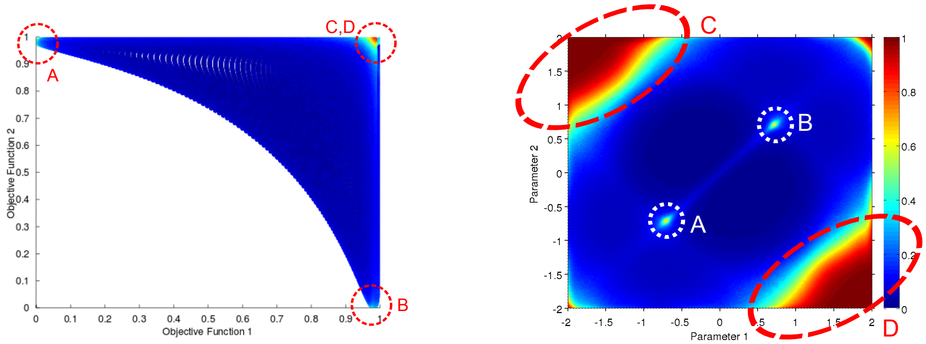
Our approach is similar in spirit to [1]. However, Binois et al. sample random functions from the posterior of the GP models and subsequently use them to approximate the PF (through Vorob'ev expectation), while we do not need to sample random functions to approximate the PF as we directly consider the posterior of the GP models.

1.2 MOO with Measurement Noise

We now study the advantages of model-based MOO in the presence of measurement noise. For this purpose, we consider the MOP2 function with additive measurement noise $\epsilon \sim \mathcal{N}(0, \mathrm{diag}([0.0025, 0.0025]))$. We uniformly randomly sample 50,000 parameters and evaluate their objective, corrupted by noise. As shown in Figure 2, approximating the Pareto front using these measurements leads to an over-optimistic estimation of the Pareto front: Evaluations with highest optimistic noise dominate all other noisy evaluations. However, when creating a response surface from noisy evaluations it is possible to approximate the real Pareto front more accurately. Moreover, the noise level learned by the GP model through hyperparameter optimization is $\mathbf{\Sigma} = \mathrm{diag}([0.0024, 0.0026])$, which is close to the real noise. The same conclusion can be reached when analyzing the hypervolume in Table 2. The value of the hypervolume computed from the measurements is over-estimated. Nevertheless, the use of a response surface allows to accurately approximate the ground truth of the hypervolume.

1.3 MOO with Parameter Uncertainty

The performances of different parameters are not equally influenced by parameter uncertainty. In some cases, perturbations of the parameters might not be very influential. In other cases, even small perturbations lead to substantially different performances. This sensitivity is crucial in fields such as robotics where parameters can correspond to mechanical values, which can be set only within a certain accuracy (e.g., the stiffness of a spring). For the selection of the final parameters by the human decision-maker, information regarding the sensitivity of the objective function with respect to the parameter choice can be invaluable. In particular cases, even solutions outside the Pareto front, but with higher robustness to parameter uncertainty might be preferable. As a proof-of-concept, we



(a) Objective function space (b) Parameter space

Figure 3: **MOO with parameter uncertainty: Sensitivity analysis.** (a) Along the Pareto front only the solutions at the two extremities of the front are robust to parameter uncertainty. This area of stable solutions is visible in the top-right corner, but the values of the objective functions are sub-optimal. (b) The robust solutions along the PF are the areas around [-0.7, -0.7] and [0.7, 0.7]. The large portions of the parameter space at the corners correspond to sub-optimal solutions.

perform a sensitivity analysis of the MOP2 function. We collect 750 randomly sampled parameters and corresponding evaluations of the objective function and learn a response surface from them. From the GP model of the objective, we compute the robustness index r at $\boldsymbol{\theta}$ as

$$r(\boldsymbol{\theta}) = r_s(\boldsymbol{\theta})/r_u(\boldsymbol{\theta})\,,$$
$$r_s(\boldsymbol{\theta}) = \tfrac{1}{n_s}\sum\nolimits_s \mathcal{N}(f(\boldsymbol{\theta}_s)|\boldsymbol{\mu}_f(\boldsymbol{\theta}), \boldsymbol{\sigma}_f(\boldsymbol{\theta}))\,, \qquad r_u(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_f(\boldsymbol{\theta}) + 3\boldsymbol{\sigma}_f(\boldsymbol{\theta})|\boldsymbol{\mu}_f(\boldsymbol{\theta}), \boldsymbol{\sigma}_f(\boldsymbol{\theta}))$$

where $\boldsymbol{\theta}_s$ is one of 2000 samples from a neighborhood around $\boldsymbol{\theta}$. Figure 3b visualizes the resulting intensity map of the robustness measure in the parameter space computed along a 200-by-200 grid. Higher robustness values belong to parameters to whose perturbations the objective function is less sensitive. For the MOP2 test function, we find four areas in the parameter space, which are robust to parameter uncertainty: two at the North-West and South-East corners, and two at the coordinates [-0.7, -0.7] and [0.7, 0.7]. Figure 3a shows that the corresponding Pareto front contains only two of these four areas: at the North-West and South-East corners of the objective function space. These two areas of robust solutions correspond to the parameters at the coordinates [-0.7, -0.7] and [0.7, 0.7]. In contrast, the other two areas of the parameter space correspond to sub-optimal (but robust) solutions far from the Pareto front.

If we only consider Pareto optimality, all solutions on the Pareto front are equally good. However, if we additionally consider the robustness with respect to the sensitivity of the objective function with respect to the parameters this is no longer true. This additional source of information can be extremely valuable in model-based MOO for making informed and robust decisions.

2 Conclusion

Current MOO methods simply return the set of evaluations belonging to the Pareto front as result of the optimization process. Simply returning this Pareto front is often inaccurate, but it is ultimately the only possibility for model-free MOO methods. In the context of multi-objective Bayesian optimization, we proposed to make full use of the models created during the optimization to analyze and and assess the quality of the proposed solutions. We showed that we can (a) compute arbitrarily dense and continuous PFs (b) approximate the real PF better in presence of measurement noise, (c) perform sensitivity analysis with respect to the parameters. These are useful tools to assist the user in their final decision. Moreover, these analyses are independent of the method used during the optimization and as such can be easily integrated into most MOBO methods.

Acknowledgments The research leading to these results has received funding from the EC's Seventh Framework Programme (FP7/2007–2013) under grant agreements #270327 (CompLACS) and #600716 (CoDyCo). MPD was supported by an Imperial College Junior Research Fellowship.

References

[1] M. Binois, D. Ginsbourger, and O. Roustant. Quantifying uncertainty on pareto fronts with gaussian process conditional simulations. In *Proceedings of Learning and Intelligent OptimizatioN Conference (LION8)*, 2014.

[2] J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors. *Multiobjective Optimization - Interactive and Evolutionary Approaches*, volume 5252 of *Lecture Notes in Computer Science*. Springer, 2008.

[3] I. Das and J. Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Structural optimization*, 14(1):63–69, 1997.

[4] 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.

[5] M. Emmerich and J.-W. Deutz, Andr Klinkenberg. The computation of the expected improvement in dominated hypervolume of pareto front approximations. Technical Report LIACS-TR 9-2008, Leiden Institute for Advanced Computer Science, September 2008.

[6] J. Knowles. Parego: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, January 2006.

[7] V. Pareto. *Manuale di economia politica*, volume 13. Societa Editrice, 1906.

[8] W. Ponweiser, T. Wagner, D. Biermann, and M. Vincze. Multiobjective Optimization on a Limited Budget of Evaluations Using Model-Assisted $\mathcal{S}$-Metric Selection. In *Parallel Problem Solving from Nature–PPSN X*, pages 784–794. Springer, 2008.

[9] M. Tesch, J. Schneider, and H. Choset. Expensive multiobjective optimization for robotics. In *International Conference on Robotics and Automation (ICRA)*, pages 973 – 980, 2013.

[10] D. A. Van Veldhuizen and G. B. Lamont. Multiobjective evolutionary algorithm test suites. In *Proceedings of the 1999 ACM symposium on Applied computing*, pages 351–357. ACM, 1999.

[11] E. Zitzler, J. D. Knowles, and L. Thiele. Quality assessment of pareto set approximations. In *Multiobjective Optimization*, pages 373–404, 2008.

[12] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, editors, *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pages 95–100, Athens, Greece, 2001. International Center for Numerical Methods in Engineering.