CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Bayesian optimization with multi-objective acquisition function for bilevel problem
Authors	Dogan, Vedat;Prestwich, Steven D.
Publication date	2023-02-23
Original Citation	Dogan, V. and Prestwich, S. (2023) 'Bayesian optimization with multi-objective acquisition function for bilevel problems', AICS2022: 30th Irish Conference on Artificial Intelligence and Cognitive Science, in L. Longo and R. O'Reilly (eds) Artificial Intelligence and Cognitive Science. Cham: Springer Nature Switzerland, pp. 409–422. https://doi.org/10.1007/978-3-031-26438-2_32
Type of publication	Conference item;Book chapter
Link to publisher's version	10.1007/978-3-031-26438-2_32
Rights	© 2023 The Author(s). Open Access. This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made - http://creativecommons.org/licenses/by/4.0/
Download date	2026-11-06 06:21:30
Item downloaded from	https://hdl.handle.net/10468/14335

UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Bayesian Optimization with Multi-objective Acquisition Function for Bilevel Problems

Vedat Dogan[1(✉)] and Steven Prestwich[2]

[1] Confirm Centre for Smart Manufacturing, School of Computer Science and Information Technology, University College Cork, Cork, Ireland
vedat.dogan@cs.ucc.ie

[2] Insight Centre for Data Analytics, School of Computer Science and Information Technology, University College Cork, Cork, Ireland
s.prestwich@cs.ucc.ie

Abstract. A bilevel optimization problem consists of an upper-level and a lower-level optimization problem connected to each other hierarchically. Efficient methods exist for special cases, but in general solving these problems is difficult. Bayesian optimization methods are an interesting approach that speed up search using an acquisition function, and this paper proposes a modified Bayesian approach. It treats the upper-level problem as an expensive black-box function, and uses multiple acquisition functions in a multi-objective manner by exploring the Pareto-front. Experiments on popular bilevel benchmark problems show the advantage of the method.

Keywords: Bayesian optimization · Bilevel optimization problems · Multi-objective acquisition · Multi-objective optimization

1 Introduction

Bilevel optimization deals with optimization problems including additional optimization problem within the constraints. Two decision-makers attempt to find his/her optimal solution on these hierarchical nested systems. The upper-level problem is the first problem, and the decision-maker is called the leader. The lower-level problem forms a constraint in the leader problem, and this decision-maker is called the follower. The leader knows the follower's objective and constraints, but the follower may have no knowledge about the leader. The decision-makers objectives are often in conflict though they may also be cooperative. During the optimization process the leader takes his/her action first. The follower takes that decision as a parameter and tries to find the best reaction. However,

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 16/RC/3918.

© The Author(s) 2023
L. Longo and R. O'Reilly (Eds.): AICS 2022, CCIS 1662, pp. 409–422, 2023.
https://doi.org/10.1007/978-3-031-26438-2_32

the follower's reaction affects the leader's decisions because the leader makes choices in the knowledge of how the follower will react.

Bilevel optimization problems occur in many practical applications including transportation, management, environmental economics, engineering and design. [43]. They also occur in machine learning: signal processing, meta-learning, hyperparameter optimization, reinforcement learning and neural architecture search can be modelled as bilevel optimization [21]. However, a lack of efficient solution methods has prevented the uptake of bilevel optimization.

The aim of this paper is to propose a new approach based on Bayesian optimization (BO) using multiple acquisition functions (MACBP) to improve efficiency (defined in terms of function evaluations). BO is a surrogate-based method for solving black-box functions that are expensive to evaluate [16], making it a useful approach to solving bilevel problems. An example is the BOBP algorithm [23]. BOBP used one lower confidence bound (LCB) acquisition function and obtains one decision point at a time. We propose using more than one acquisition function to improve the optimization process by making a wiser choice of acquisition points. Multiple acquisition functions have been used in BO, for example in the MACE algorithm for optimizing analog circuit design [33]. However, to the best of our knowledge no work has been done on this area for solving bilevel optimization problems.

Our contributions are twofold:

- We use multiple acquisition functions, as no single acquisition function is appropriate for every problem [18]. We solve the resulting multi-objective optimization problem with evolutionary techniques, and select new points on the Pareto-front solution set.
- We show empirically how using multiple acquisition functions affects optimization performance.

The rest of the paper is organised as follows. Background is provided in Sect. 2. The preliminaries for general bilevel optimization problems and BO are given in Sect. 3. The proposed method and algorithm details are explained in Sect. 4. In Sect. 5 the experimental setup is described. Finally, Sect. 6 concludes the paper and proposes future work.

2 Background

Bilevel optimization problems are described in two areas. In game theory von Stackelberg [50] proposed descriptive models of decision behaviour and built game-theoretic equilibria. In mathematical programming problems containing nested lower-level optimization problem as a constraint of upper-level optimization problem [8]. The hierarchical structure of bilevel problems might cause difficulties such as non-convexity and having no relation between instances. It is known to be strongly NP-hard [17].

A considerable number of exact approaches have been applied to bilevel problems. *Karush-Kuhn-Tucker* conditions [3] can be used to reformulate a bilevel

problem to a single-level problem. Penalty functions compute the stationary points and local optima. Vertex enumeration has been used with a version of the Simplex method [6]. Gradient information for the follower problem can be extracted for use by the leader objective function. In terms of integer and mixed integer bilevel problems, reformulation [14], branch-and-bound [4] and parametric programming approaches have been applied to solve bilevel problems [27].

Because of the inefficiency of exact methods in complex bilevel problems, several kinds of meta-heuristics have been applied to bilevel problems in the literature. Four existing categories have been published in [53]: the nested sequential approach [25], the single-level transformation approach, the multi-objective approach [41] and the co-evolutionary approach [31]. An algorithm based on a human evolutionary model for non-linear bilevel problems [34], and the Bilevel Evolutionary Algorithm based on Quadratic approximations (BLEAQ) have been proposed [45]. This is another work which attempts to try to reduce the number of follower optimizations. The algorithm approximates the inducible region through the feasible region of the bilevel problem. In [40] they consider single optimization problem at both levels. They propose the Sequential Averaging Method (SAM) algorithm. In different recent works [32,42] they used a truncated backpropagation approach to approximate the (stochastic) gradient of the upper-level problem. Basically, they use a dynamical system to model an optimization algorithm that solves the lower-level problem, and replaces the lower-level optimal solution. In another work [19] they developed a two-timescale stochastic approximation algorithm (TTSA) for solving a bilevel problem assuming the follower problem is unconstrained and strongly convex and the leader is a smooth objective function.

Many practical problems can be modelled and solved as Stackelberg games in the field of economics [46,47] including principal agency problems and policy decisions. Hierarchical decision-making processes in management [2,51] and in engineering and optimal structure design are other practical examples [24,48]. Network design and the toll setting problem are the most popular applications in the field of transportation [9,11,35]. Finding optimal chemical equilibria, planning the preposition of defensive missile interceptors to counter an attacking threat, and interdicting nuclear weapons are other applications [10]. Inverse optimal control problems are modelled as bilevel optimization problems in nature [22,37,52]. There are many applications in robotics, computer vision, communication theory etc. In the machine learning community, bilevel optimization received significant attention recently and became an important framework in applications. Some interesting topics are meta-learning [5,15,39], hyperparameter optimization [13,42], reinforcement learning [19,26] and signal processing [29].

3 Preliminaries

The description of the MACBP algorithm will be divided into three parts. Firstly, we explain bilevel programming problems and their structure. Secondly, we discuss Bayesian optimization (BO) and Gaussian processes (GP). Finally, we propose the MACBP algorithm for solving bilevel optimization problems.

3.1 Bilevel Optimization Problems

For the upper-level objective function $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and lower-level objective function $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, bilevel optimization problem can be defined as

$$\begin{aligned} &\min_{\mathbf{x}_u \in X_u, \mathbf{x}_l \in X_l} F(\mathbf{x}_u, \mathbf{x}_l) \\ &\text{s.t. } \mathbf{x}_l \in \operatorname*{argmin}_{\mathbf{x}_l} \{f(\mathbf{x}_u, \mathbf{x}_l) : g_j(\mathbf{x}_u, \mathbf{x}_l) \leq 0, j = 1, 2, \ldots, J\} \\ &\quad G_k(\mathbf{x}_u, \mathbf{x}_l) \leq 0, \ k = 1, 2, \ldots, K \end{aligned} \tag{1}$$

where $\mathbf{x}_u \in X_U, \mathbf{x}_l \in X_L$ are upper-level and lower-level decision variables and decision spaces, G_k, g_j are constraints.

Because the lower-level decision maker depends on the upper-level variables, for every decision x_u, there is a follower-optimal decision x_l^*. In bilevel optimization, the decision set $\mathbf{x}^* = (\mathbf{x}_u^*, \mathbf{x}_l^*)$ is a feasible member for the upper-level *only if* it satisfies all the upper-level constraints and vector $\mathbf{x}_x^*$ is an optimal solution to the lower-level problem with upper-level decision as *parameter*.

3.2 Bayesian Optimization and Gaussian Process

BO is a method to optimize expensive-to-evaluate black-box functions. The probabilistic surrogate model and acquisition functions is important for BO. Predictions and uncertainties are provided by the surrogate model. It uses commonly GP [49] as a surrogate model, to obtain a posterior distribution $\mathbb{P}(\mathbf{f}|D)$ over the objective function $\mathbf{f}$ given the observed data $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$. An acquisition function uses the posterior distribution to explore the search space. So the surrogate model is assisted by an *acquisition function* to choose the next candidate or a set of candidates $X_{cand} = \{\mathbf{x}_i\}_{i=1}^q$. Though the objective function is expensive to evaluate, the surrogate-based acquisition function is not, so it can be optimized much more easier than the true function to yield X_{cand}.

Let us assume that we have a set of collection points $\{x_1, \ldots, x_n\} \in \mathbb{R}^d$ and an objective function values of these points $\{f(x_1), \ldots, f(x_n)\}$. After we observe n points, the mean vector is obtained by evaluating a *mean function* μ_0 at each point x_i and the covariance matrix by evaluating a *covariance function* or *kernel* Σ_0 at each pair of x_i, x_j. The resulting prior distribution on $\{f(x_1), \ldots, f(x_n)\}$ is defined by

$$f(x_{1:n}) \sim N(\mu_0(x_{1:n}), \Sigma_0(x_{1:n}, x_{1:n})) \tag{2}$$

Let us suppose we wish to find a value of $f(X_{cand})$ at some new candidate point X_{cand}. For this purpose, the prior over $\{f(x_{1:n}), f(X_{cand})\}$ is given by (2). Then we can compute the distribution of $f(X_{cand})$ given the observations

$$f(X_{cand})|f(x_{1:n}) \sim N(\mu_0(X_{cand}), \sigma_0^2(X_{cand})) \tag{3}$$

$$\mu_0(X_{cand}) = \Sigma_0(X_{cand}, x_{1:n})\Sigma_0(x_{1:n}, x_{1:n})^{-1}(f(x_{1:n}) - \mu_0(x_{1:n})) + \mu_0(X_{cand}) \tag{4}$$

$$\sigma_n^2(X_{cand}) = \Sigma_0(X_{cand}, X_{cand}) - \Sigma_0(X_{cand}, x_{1:n})(\Sigma_0(x_{1:n}, x_{1:n})^{-1}\Sigma_0(x_{1:n}, X_{cand}) \tag{5}$$

The distribution is called the *posterior probability distribution* in Bayesian statistics. So it is very important during the Bayesian optimization and Gaussian process to choose the next point to evaluate.

Acquisition functions are used to guide the search to a promising next point during the likelihood optimization, and it balances exploration and exploitation. Several acquisition functions have been developed over the years, such as probability of improvement (PI), expected improvement (EI) and upper confidence bound (UCB).

Probability of Improvement. The PI acquisition function tries to measure the probability that an arbitrary x exceeds the current best. Given the minimum objective function value τ in the data set, the formulation is as follows [30]:

$$PI(x) = \Phi(\lambda) \tag{6}$$

where $\Phi(\lambda)$ is the cumulative distribution function of standard normal distribution and $\lambda = (\tau - \mu(x))/(\sigma(x))$.

Expected Improvement. We can expect that the observation x will not only reach the current best, but also reach the current best value at the highest magnitude. The corresponding formulation can be expressed as [36]:

$$EI(x) = \sigma(x)(\lambda\Phi(\lambda) + \phi(\lambda)) \tag{7}$$

where $\phi(\cdot)$ is probability density function of standard normal distribution and $\lambda = (\tau - \mu(x))/(\sigma(x))$.

Upper Confidence Bound. This is not an improvement-based strategies like EI and PI. It tries to guide the search from an optimistic perspective. The formulation is:

$$UCB(x) = \mu(x) + \beta\sigma(x) \tag{8}$$

where β is a parameter represents exploration-exploitation trade-off. We fix $\beta = 0.1$.

Algorithm 1. The MACBP Algorithm for Upper-level Optimization

Inputs: $\mathbf{F}_u(\mathbf{x}_u, \mathbf{x}_l) : \mathbf{x_u} \in \mathbb{X}_u, \mathbf{x_l} \in \mathbb{X}_l$,
Total epoch N,
Initial decision data set $D = (\mathbf{x}_{u_i}, F_u(\mathbf{x}_{u_i}, \mathbf{x}^*_{l_i}))_{i=1:n}$ with size of n,
1: $\mathbf{x_l^*}$: Find the best lower-level decisions and set as upper-level parameters,
2: Initialize *Gaussian model* with Observations $\{\mathbf{x}_u, F_u(\mathbf{x}_u, \mathbf{x}^*_l)\}$
3: **for** i = 0 : N **do**
4: Construct the acquisition functions according to the Eqs. (8), (7) and (6)
5: Find Pareto-front of multi-objective acquisition problem using the NSGA-II algorithm
6: For the chosen upper-level decision $\mathbf{x}_u$, find optimal $\mathbf{x}^*_l$ by using SLSQP
7: Calculate fitness scores $\mathbf{F}^*_u$ and $\mathbf{f}^*_u$
8: Update the data set $D = (\mathbf{x}_{u_i}, \mathbf{F}_u(\mathbf{x}_{u_i}, \mathbf{x}^*_{l_i}))_{i=1:q}$
9: Update Multi-objective Gaussian Process Model with new observations
10: **end for**
11: **Return** Best F^*_u and corresponding optimum variables $\mathbf{x}^*_u, \mathbf{x}^*_l$

4 Proposed Method

Bilevel problems have two levels of optimization tasks, such that the lower-level problem is a constraint of the upper-level problem. In general bilevel problems, the follower depends on the leader decisions x_u. The leader has no control over the follower decision x_l. For every leader decision there is an optimal follower decision, which can be called the reaction. Because the follower problem is a parametric optimization problem that depends on the leader decision x_u, it is very time-consuming to adopt a nested strategy approach which sequentially solves both levels. In the continuous domain, the computational cost is very high. During the optimization process, it is important to choose wisely the next leader decision x_u according to make the process faster. For this purpose, we will present the proposed algorithm, we call MACBP, for solving bilevel problems by BO via multiple acquisition functions.

Problem Statement. Let us assume that we have a expensive black-box function that takes leader decisions in leader decision space $x_u \in X_u$ and follower decisions coming from the follower decision maker $x_l \in X_l$ as input. The function returns a scalar fitness score:

$$F(x_u, x_l) : X_u \times X_l \to \mathbb{R} \tag{9}$$

Given a budget of N, the leader makes a decision and the follower makes its decisions accordingly. The leader can observe this information during the optimization process, and how follower decision maker reacting to leader decisions in every iteration and chooses the next leader decision to optimize the fitness score.

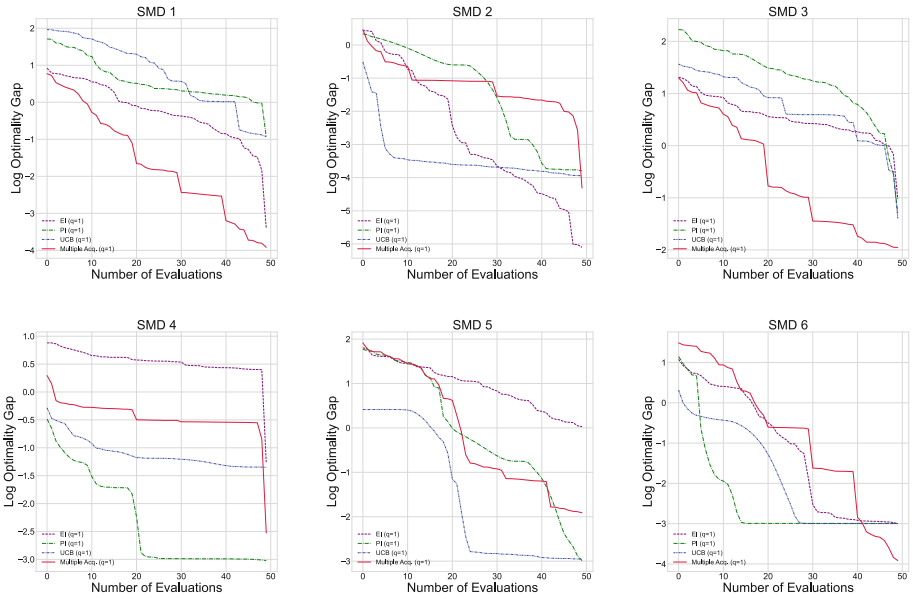


Fig. 1. The log optimality gap for the leader's objective in SMD benchmark problems.

Algorithm Description. First we discuss fitting the decision data to the Gaussian process model. After observing n decision data $\{(x_u^i, y^i)\}_{i=1}^n$ where $y_i = F(x_u^i, x_l^i)$, we fit the data set to the Gaussian process model. After we have the data set let $\hat{X}^n = ((x_u)^1, ..., (x_u)^n)$ and $Y^n = (y^1, ..., y^n)$, then we define the Gaussian process by a prior mean $\mu(x_u)$ and prior covariance function $k((x_u), (x_u^{'}))$. After observing n data points, let $K = k(\hat{X}^n, \hat{X}^n) \in \mathbb{R}^{n \times n}$. So the posterior mean and covariance is given by:

$$\mu(x_u)^n = \mu(x_u) + k(x_u, \hat{X}^n)(K + \sigma_0^2 I)^{-1}(Y^n - \mu(\hat{X}^n)) \tag{10}$$

$$k^n(x_u, x_u^{'})^n = k^n(x_u, x_u^{'}) - k^n(x_u, \hat{X}^n)(K + \sigma_0^2 I)^{-1} k(\hat{X}^n, x_u^{'}) \tag{11}$$

After fitting the data to the model, we choose the next leader decision. After we find the optimal reaction (x_u^{n+1}, x_l^{n+1}) and the fitness score of leader function $F(x_u^{n+1}, x_l^{n+1}) = y^{n+1}$, we update the Gaussian process model with new decision data (x_u^{n+1}, y^{n+1}). We shared the details of the MACBP algorithm on Algorithm 1 for upper-level optimization.

4.1 Multi-objective Optimization

There are multiple objectives to optimize when we consider the multi-objective optimization problems. It is formulated as

$$\underset{\mathbf{x} \in X}{\text{minimize}} \;\; \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_d(\mathbf{x})) \tag{12}$$

Table 1. The summary of SMD benchmark problems

	Dimension		Search domain		Optimal value	
	UL	LL	UL	LL	UL	LL
SMD1	2	2	$[-5, 10] \times [-5, 10]$	$[-5, 10] \times [-\frac{\pi}{2}, \frac{\pi}{2}]$	0.0	0.0
SMD2	2	2	$[-5, 10] \times [-5, 1]$	$[-5, 10] \times (0, \mathrm{e}]$	0.0	0.0
SMD3	2	2	$[-5, 10] \times [-5, 10]$	$[-5, 10] \times [-\frac{\pi}{2}, \frac{\pi}{2}]$	0.0	0.0
SMD4	2	2	$[-5, 10] \times [-1, 1]$	$[-5, 10] \times (0, \mathrm{e}]$	0.0	0.0
SMD5	2	2	$[-5, 10] \times [-5, 10]$	$[-5, 10] \times [-5, 10]$	0.0	0.0
SMD6	2	2	$[-5, 10] \times [-5, 10]$	$[-5, 10] \times [-5, 10]$	0.0	0.0

for a vector-valued function $\mathbf{f}(\mathbf{x}) : \mathbb{R} \rightarrow \mathbb{R}^d$ and $X \in \mathbb{R}$. So it is hard and commonly impossible to find a single optimum solution as there may be conflicts between the objectives. Therefore the main goal for these problems is to approximate the Pareto-front. Let us say that $\mathbf{f}(\mathbf{x})$ *dominates* another solution $\mathbf{f}(\mathbf{x}')$ if $\mathbf{f}^{(i)}(x) \succ \mathbf{f}^{(i)}(x')$ for all $i = 1, 2, \ldots, M$ and there exists $i' \in \{1, 2, \ldots, M\}$ such that $f^{i'}(x) \succ f^{i'}(x')$. So we can express the *Pareto-optimal* by $P^* = \{\mathbf{f}(\mathbf{x}) \; s.t. \; \nexists \mathbf{x}' \in X : \mathbf{f}(\mathbf{x}') \succ \mathbf{f}(\mathbf{x})\}$ and $X^* = \{\mathbf{x} \in \mathbf{X} \; s.t. \; \mathbf{f}(\mathbf{x}) \in P^*\}$. A solution set is Pareto-optimal if it is not dominated by any other point and it dominates at least one point. The Pareto-set the set of all Pareto-optimal points, and a set of Pareto-optimal points is called a Pareto-front. There are many multi-objective optimization algorithms such as non-dominated sorting based genetic algorithm (NSGA-II) [12], multi-objective evolutionary algorithm based on decomposition (MOEA/D) [55] and multi-objective optimization based on differential evolution (DEMO) [54].

4.2 Multi-objective Acquisition Function in Bayesian Optimization

Different acquisition functions have different characteristics according to their structure and point selection strategy. Improvement based strategies rely on the best selection so far at each iteration. For example the PI function value decreases when difference between mean function the best objective value so far below zero, $\mu(x) - F^*(x) < 0$. The EI function value at sampled points would always be worse than the EI values at pending decision points. Uncertainty-based acquisition functions, for instance UCB, increase as $\sigma(x)$ increases.

According to the different selection strategies explained above, we use the multi-objective optimization method NSGA-II in this work, to find the best trade-off between acquisition functions. Then we select the next point of the leader's decision during the bilevel optimization process from the best trade-off between acquisition functions. This is called the Pareto-front of acquisition functions. So in every iteration the multi-objective optimization problem constructed is:

$$\underset{\mathbf{x} \in X}{\text{minimize}} \left\{ -UCB(x), -PI(x), -EI(x) \right\} \tag{13}$$

After we find the Pareto-front from the multi-objective optimization Problem 13 we make the random selection from Pareto-optimal decision set.

Table 2. Upper-level function evaluations for the proposed MACBP algorithm and other known algorithms for SMD1-SMD6

	Upper-Level Function Evaluations					
	MACBP	CGA-BS [1]	BLMA [20]	NBLE [20]	BIDE [20]	BLEAQ [45]
SMD1	9.70×10^1	1.01×10^4	1.19×10^3	1.52×10^3	6.00×10^3	1.19×10^3
SMD2	19.90×10^1	5.00×10^4	1.20×10^3	1.56×10^3	6.00×10^3	1.20×10^3
SMD3	10.30×10^1	1.00×10^4	1.29×10^3	1.56×10^3	6.00×10^3	1.29×10^3
SMD4	26.90×10^1	1.25×10^5	1.31×10^3	1.53×10^3	6.00×10^3	1.31×10^3
SMD5	17.00×10^1	1.00×10^5	2.06×10^3	3.40×10^3	6.00×10^3	2.06×10^3
SMD6	14.70×10^1	1.37×10^5	4.08×10^3	4.06×10^3	6.00×10^3	4.08×10^3

Table 3. Upper-level accuracy for the proposed MACBP algorithm and other known algorithms for SMD1-SMD6.

	Upper-Level Accuracy					
	MACBP	CGA-BS [1]	BLMA [20]	NBLE [20]	BIDE [20]	BLEAQ [45]
SMD1	2.51×10^{-6}	0	1.00×10^{-6}	5.03×10^{-6}	3.41×10^{-6}	1.00×10^{-6}
SMD2	1.33×10^{-6}	2.22×10^{-6}	1.00×10^{-6}	3.17×10^{-6}	1.29×10^{-6}	5.44×10^{-6}
SMD3	3.82×10^{-6}	0	1.00×10^{-6}	1.37×10^{-6}	4.10×10^{-6}	7.55×10^{-6}
SMD4	6.27×10^{-7}	3.41×10^{-11}	1.00×10^{-6}	9.29×10^{-6}	2.30×10^{-6}	1.00×10^{-6}
SMD5	7.44×10^{-7}	1.13×10^{-9}	1.00×10^{-6}	1.00×10^{-6}	1.58×10^{-6}	1.00×10^{-6}
SMD6	1.09×10^{-7}	9.34×10^{-11}	1.00×10^{-6}	1.00×10^{-6}	3.47×10^{-6}	1.00×10^{-6}

5 Experiments

We evaluate the MACBP algorithm using two experiments. First, we run the experiments by choosing a single point at each iteration for the setting of $N_{iter} = 50$. We set the number of initial random sampling to $N_{init} = 20$. Then, we compare the results with those for the three single acquisition functions EI, PI and UCB performances. Second, we run the experiment with stopping criteria of $d < 10^{-5}$ where d represents the difference between the results and the optimum value of functions. We compare the performance of our proposed method in terms of function evaluations in Table 2 and in terms of accuracy in Table 3. We run the algorithm in sequential mode and the Matern52 kernel is used for GP for both experiments. The parameters for acquisition functions are declared in Sect. 3.2. For the first experiment, the experiments are repeated 31 times to average the random fluctuations and the optimality gap in the log scale presented in Fig. 1.

The optimization is completed in a single core of 1.4 Ghz Quad Core i5, 8 Gb 2133 MHz LPDDR3 RAM. Bayesian optimization is implemented in the Python language and uses BoTorch [38], the SLSQP algorithm [28] is used for lower-level optimization, and the NSGA-II algorithm for multi-objective optimization by using PyMOO [7] library.

5.1 SMD Problems

We evaluated the MACBP algorithm on six standard benchmark problems proposed in [44]. It is called SMD test problems and the problems are unconstrained and high-dimensional with controllable complexities. They are scalable in terms of the number of decision variables. Each problem in the benchmark represents a different difficulty level in terms of convergence, the complexity of interaction and lower-level multi-modality as declared in [44]. Table 1 provides details on the problems. For all functions we used 2D decision variables. The total function evaluations for the leader's objective can be calculated by $N_{iter} + N_{init}$.

5.2 Results

Although bilevel optimization problems deal with the leader's and follower's optimization problems, we shall consider only the leader's performance as it is the only one we model as an expensive black-box function. The optimality gap plots between true optimal points and approximated points in 50 iterations *in log scale* are given in Fig. 1. As can be seen in Fig. 1, the proposed algorithm for bilevel optimization is competitive with the sequential Bayesian method at upper-level optimization with the UCB, EI and PI acquisition functions. We fixed the iteration number for the first experiment to see how using multiple acquisitions effect the performance when we compare it with single acquisition functions. As we can see in Fig. 1, the multi-objective acquisition approach gave better results than EI, PI and UCB alone for SMD1, SMD3 and SMD6. We can see that at the end of optimization by reaching the closer point to the optimal value for these problems. The proposed algorithm gave better performance than UCB and PI for the SMD2 problem but EI reached closer point to the optimal at the end. PI reached the best point at the end of iterations for SMD 4 and they are so close as it is the second best one.

In the second experiment, we can see from Table 3 the MACBP algorithm reached better results for SMD4, SMD5 and SMD7 than compared algorithms. For SMD1, we get closer to the optimal solution than NBLE and BIDE algorithms. We reached the better results for SMD2 when we compare the results with NBLE and BLEAQ algorithms. Comparing with BIDE and BLEAQ, the proposed algorithm get better results for SMD3. In terms of function evaluations, our MACBP algorithm decreased significantly the function evaluations as we can see at the Table 2 when we compare the other state-of-art algorithm in the literature.

6 Conclusion

In this paper, we proposed the MACBP algorithm, a Bayesian approach via multi-objective acquisition functions for bilevel optimization problems. We approached the leader's objective as an expensive black-box function. We used multiple acquisition functions during the bilevel optimization process, and made

our selection from a Pareto-front solution set in each iteration. We selected six popular SMD benchmark problems for the experiments. We compared our experimental results with a classic sequential setting of Bayesian optimization with each acquisition function performance individually. We also compare our results in terms of required function evaluations at the upper-level. It is shown that the proposed MACBP algorithm is competitive with existing well-known algorithms compared in the paper for solving bilevel optimization problems.

Acknowledgement. This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 16/RC/3918 which is co-funded under the European Regional Development Fund. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

References

1. Abo-Elnaga, Y., Nasr, S.: Modified evolutionary algorithm and chaotic search for bilevel programming problems. Symmetry **12** (2020). https://doi.org/10.3390/SYM12050767
2. Bard, J.F.: Coordination of a multidivisional organization through two levels of management. Omega **11**(5), 457–468 (1983)
3. Bard, J.F., Falk, J.E.: An explicit solution to the multi-level programming problem. Comput. Oper. Res. **9**(1), 77–100 (1982). https://doi.org/10.1016/0305-0548(82)90007-7
4. Bard, J.F., Moore, J.T.: A branch and bound algorithm for the bilevel programming problem. SIAM J. Sci. Stat. Comput. **11**(2), 281–292 (1990). https://doi.org/10.1137/0911017
5. Bertinetto, L., Henriques, J.F., Torr, P., Vedaldi, A.: Meta-learning with differentiable closed-form solvers. In: International Conference on Learning Representations (2019). https://openreview.net/forum?id=HyxnZh0ct7
6. Bialas, W., Karwan, M.: On two-level optimization. IEEE Trans. Autom. Control **27**(1), 211–214 (1982). https://doi.org/10.1109/TAC.1982.1102880
7. Blank, J., Deb, K.: Pymoo: multi-objective optimization in python. CoRR abs/2002.04504 (2020). https://arxiv.org/abs/2002.04504
8. Bracken, J., McGill, J.T.: Mathematical programs with optimization problems in the constraints. Oper. Res. **21**(1), 37–44 (1973). https://www.jstor.org/stable/169087
9. Brotcorne, L., Labbé, M., Marcotte, P., Savard, G.: A bilevel model for toll optimization on a multicommodity transportation network. Transp. Sci. **35**, 345–358 (2001). https://doi.org/10.1287/trsc.35.4.345.10433
10. Brown, G., Carlyle, M., Diehl, D., Kline, J., Wood, R.: A two-sided optimization for theater ballistic missile defense. Oper. Res. **53**, 745–763 (2005). https://doi.org/10.1287/opre.1050.0231
11. Constantin, I., Florian, M.: Optimizing frequencies in a transit network: a nonlinear bi-level programming approach. Int. Trans. Oper. Res. **2**(2), 149–164 (1995)
12. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002). https://doi.org/10.1109/4235.996017

13. Feurer, M., Hutter, F.: Hyperparameter optimization. In: Hutter, F., Kotthoff, L., Vanschoren, J. (eds.) Automated Machine Learning. TSSCML, pp. 3–33. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05318-5_1
14. Fontaine, P., Minner, S.: Benders decomposition for discrete-continuous linear bilevel problems with application to traffic network design. Transp. Res. Part B: Methodol. **70**(C), 163–172 (2014). https://doi.org/10.1016/J.TRB.2014.09.007. https://ideas.repec.org/a/eee/transb/v70y2014icp163-172.html
15. Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., Pontil, M.: Bilevel programming for hyperparameter optimization and meta-learning. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 1568–1577. PMLR (2018). https://proceedings.mlr.press/v80/franceschi18a.html
16. Frazier, P.: A tutorial on Bayesian optimization. arXiv abs/1807.02811 (2018)
17. Hansen, P., Jaumard, B., Savard, G.: New branch-and-bound rules for bilevel linear programming. SIAM J. Sci. Stat. Comput. **13**, 273 (1992). https://doi.org/10.1137/0913069
18. Hoffman, M., Brochu, E., de Freitas, N.: Portfolio allocation for Bayesian optimization. In: Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence, UAI 2011, Arlington, Virginia, USA, pp. 327–336. AUAI Press (2011)
19. Hong, M., Wai, H.T., Wang, Z., Yang, Z.: A two-timescale framework for bilevel optimization: complexity analysis and application to actor-critic. arXiv abs/2007.05170 (2020)
20. Islam, M.M., Singh, H.K., Ray, T., Sinha, A.: An enhanced memetic algorithm for single-objective bilevel optimization problems. Evol. Comput. **25**, 607–642 (2017). https://doi.org/10.1162/EVCOa00198
21. Ji, K., Yang, J., Liang, Y.: Bilevel optimization: convergence analysis and enhanced design. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 4882–4892. PMLR (2021). https://proceedings.mlr.press/v139/ji21c.html
22. Johnson, M., Aghasadeghi, N., Bretl, T.: Inverse optimal control for deterministic continuous-time nonlinear systems. In: 52nd IEEE Conference on Decision and Control, pp. 2906–2913 (2013). https://doi.org/10.1109/CDC.2013.6760325
23. Kieffer, E., Danoy, G., Bouvry, P., Nagih, A.: Bayesian optimization approach of general bi-level problems. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO 2017, pp. 1614–1621. Association for Computing Machinery, New York (2017). https://doi.org/10.1145/3067695.3082537
24. Kirjner-Neto, C., Polak, E., Kiureghian, A.D.: An outer approximation approach to reliability-based optimal design of structures. J. Optim. Theory Appl. **98**(1), 1–16 (1998)
25. Koh, A.: Solving transportation bi-level programs with differential evolution. In: 2007 IEEE Congress on Evolutionary Computation, pp. 2243–2250 (2007). https://doi.org/10.1109/CEC.2007.4424750
26. Konda, V., Tsitsiklis, J.: Actor-critic algorithms. In: Solla, S., Leen, T., Müller, K. (eds.) Advances in Neural Information Processing Systems, vol. 12. MIT Press (1999). https://proceedings.neurips.cc/paper/1999/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf
27. Koppe, M., Queyranne, M., Ryan, C.T.: Parametric integer programming algorithm for bilevel mixed integer programs. J. Optim. Theory Appl. **146**(1), 137–150 (2010). https://doi.org/10.1007/S10957-010-9668-3

28. Kraft, D.: A software package for sequential quadratic programming. Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt Koln: Forschungsbericht, Wiss. Berichtswesen d. DFVLR (1988). https://books.google.ie/books?id=4rKaGwAACAAJ
29. Kunapuli, G., Bennett, K., Hu, J., Pang, J.S.: Classification model selection via bilevel programming. Optim. Methods Softw. **23**(4), 475–489 (2008)
30. Kushner, H.J.: A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J. Basic Eng. **86**, 97–106 (1963)
31. Legillon, F., Liefooghe, A., Talbi, E.G.: Cobra: a cooperative coevolutionary algorithm for bi-level optimization. In: 2012 IEEE Congress on Evolutionary Computation, pp. 1–8 (2012). https://doi.org/10.1109/CEC.2012.6256620
32. Likhosherstov, V., Song, X., Choromanski, K., Davis, J., Weller, A.: UFO-BLO: unbiased first-order bilevel optimization. arXiv abs/2006.03631 (2020)
33. Lyu, W., Yang, F., Yan, C., Zhou, D., Zeng, X.: Batch Bayesian optimization via multi-objective acquisition ensemble for automated analog circuit design. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 3306–3314. PMLR (2018). https://proceedings.mlr.press/v80/lyu18a.html
34. Ma, L., Wang, G.: A solving algorithm for nonlinear bilevel programing problems based on human evolutionary model. Algorithms **13**(10) (2020). https://www.mdpi.com/1999-4893/13/10/260
35. Migdalas, A.: Bilevel programming in traffic planning: models, methods and challenge. J. Glob. Optim. **7**, 381–405 (1995). https://doi.org/10.1007/BF01099649
36. Močkus, J.: On Bayesian methods for seeking the extremum. In: Marchuk, G.I. (ed.) Optimization Techniques 1974: Optimization Techniques IFIP Technical Conference Novosibirsk. LNCS, vol. 27, pp. 400–404. Springer, Heidelberg (1975). https://doi.org/10.1007/3-540-07165-2_55
37. Mombaur, K., Truong, A., Laumond, J.P.: From human to humanoid locomotion-an inverse optimal control approach. Auton. Robots **28**, 369–383 (2010). https://doi.org/10.1007/s10514-009-9170-7
38. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library. CoRR abs/1912.01703 (2019). https://arxiv.org/abs/1912.01703
39. Rajeswaran, A., Finn, C., Kakade, S.M., Levine, S.: Meta-learning with implicit gradients. CoRR abs/1909.04630 (2019). https://arxiv.org/abs/1909.04630
40. Sabach, S., Shtern, S.: A first order method for solving convex bi-level optimization problems (2017). https://doi.org/10.48550/ARXIV.1702.03999
41. Sahin, K., Ciric, A.R.: A dual temperature simulated annealing approach for solving bilevel programming problems. Comput. Chem. Eng. **23**, 11–25 (1998)
42. Shaban, A., Cheng, C.A., Hatch, N., Boots, B.: Truncated back-propagation for bilevel optimization. CoRR abs/1810.10667 (2018). https://arxiv.org/abs/1810.10667
43. Sinha, A., Malo, P., Deb, K.: A review on bilevel optimization: from classical to evolutionary approaches and applications. IEEE Trans. Evol. Comput. (2017). https://doi.org/10.1109/TEVC.2017.2712906
44. Sinha, A., Malo, P., Deb, K.: Unconstrained scalable test problems for single-objective bilevel optimization. In: 2012 IEEE Congress on Evolutionary Computation, pp. 1–8 (2012). https://doi.org/10.1109/CEC.2012.6256557
45. Sinha, A., Malo, P., Deb, K.: An improved bilevel evolutionary algorithm based on quadratic approximations. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 1870–1877 (2014). https://doi.org/10.1109/CEC.2014.6900391

46. Sinha, A., Malo, P., Frantsev, A., Deb, K.: Multi-objective stackelberg game between a regulating authority and a mining company: a case study in environmental economics. In: 2013 IEEE Congress on Evolutionary Computation, pp. 478–485 (2013). https://doi.org/10.1109/CEC.2013.6557607
47. Sinha, A., Malo, P., Frantsev, A., Deb, K.: Finding optimal strategies in a multi-period multi-leader-follower stackelberg game using an evolutionary algorithm. Comput. Oper. Res. **41**, 374–385 (2014)
48. Smith, W.R., Missen, R.W.: Chemical reaction equilibrium analysis: theory and algorithms. In: Chemical Reaction Equilibrium Analysis: Theory and Algorithms (1982)
49. Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.W.: Information-theoretic regret bounds for gaussian process optimization in the bandit setting. IEEE Trans. Inf. Theory **58**(5), 3250–3265 (2012). https://doi.org/10.1109/tit.2011.2182033
50. von Stackelberg, H.: The Theory of the Market Economy. William Hodge (1952). https://books.google.ie/books?id=fjIAtQEACAAJ
51. Sun, H., Gao, Z., Wu, J.: A bi-level programming model and solution algorithm for the location of logistics distribution centers. Appl. Math. Model. **32**(4), 610–616 (2008)
52. Suryan, V., Sinha, A., Malo, P., Deb, K.: Handling inverse optimal control problems using evolutionary bilevel optimization. In: 2016 IEEE Congress on Evolutionary Computation (CEC), pp. 1893–1900 (2016). https://doi.org/10.1109/CEC.2016.7744019
53. Talbi, E.G.: A taxonomy of metaheuristics for bi-level optimization. In: Talbi, E.G. (ed.) Metaheuristics for Bi-level Optimization. SCI, vol. 482, pp. 1–39. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37838-6_1
54. Tusar, T., Filipic, B.: Demo: differential evolution for multiobjective optimization. In: Proceedings of the 3rd International Conference on Evolutionary Multi-Criterion Optimization, Guanajuato, Mexico, pp. 520–533 (2005). https://doi.org/10.1007/978-3-540-31880-4-36
55. Zhang, Q., Li, H.: MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007). https://doi.org/10.1109/TEVC.2007.892759

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

