Multi-objective Bilevel Decision Making with Noisy Objectives: A Batch Bayesian Approach

Vedat Dogan[a,b;*] **and Steven Prestwich**[a,b]

[a]Insight Centre for Data Analytics
[b]Confirm Centre for Smart Manufacturing
School of Computer Science and Information Technology,
University College Cork, Ireland
ORCiD ID: Vedat Dogan https://orcid.org/0000-0002-3807-5425,
Steven Prestwich https://orcid.org/0000-0002-6218-9158

Abstract. Bilevel multi-objective optimization is a field of mathematical programming representing a nested hierarchical decision-making process, with one or more decision-makers at each level. These problems appear in many practical applications, solving tasks such as optimal control, process optimization, development of government and game strategy, and transportation. Uncertainty cannot be ignored in these practical problems. We present a hybrid algorithm called BAMBINO, based on the batch Bayesian approach via expected hypervolume improvement, that can handle uncertainty at the upper level. Three popular modified benchmark problems with multiple dimensions, and one real-world example from the field of environmental economics are used to evaluate the performance. The real-world example is a decision-making problem between a governmental authority and a gold mining company. The experimental results under the objective noise compared with two popular algorithms in the literature. The results show that BAMBINO is computationally efficient and capable of handling upper-level objective uncertainty while approximating the Pareto-optimal front. We also evaluate the effect of batch size on performance.

1 Introduction

Hierarchical decision-making has an extensive history, in Game Theory as first realized by von Stackelberg [23] and in the subfield of mathematical programming called bilevel optimization [3]. A bilevel optimization problem contains a nested *inner* optimization problem which is a constraint of an *outer* optimization problem. The outer optimization task is referred to as the *upper level* or *leader* while the inner optimization problem is referred to as the *lower level* or *follower*. Existing bilevel research has focused mainly on single-objective leaders and followers. *Multi-objective* bilevel optimization is relatively neglected, but there is work in the fields of classical optimization [11] and evolutionary computation [15].

Our work focuses on the special case of multi-objective bilevel problems in which the leader has noisy objectives. We assume that the follower is free to choose any feasible solution from a Pareto-optimal set. We use batch Bayesian optimization to improve efficiency, approximating the leader's Pareto-optimal front using fewer function evaluations than existing works. We also present a black-box approach to the noisy leader's objectives for handling the uncertainty during decision-making.

Hierarchical decision-making under uncertainty with noisy objectives becomes more interesting in a bilevel structure. The follower can observe the leader's decisions, but the leader may have no idea how the follower is going to respond. Previously observed decisions are therefore important. Uncertainty in the objective also complicates the leader's decision-making, and our algorithm uses a specifically designed acquisition function called qNEHVI to maximize expected hypervolume improvement under noisy objectives. We call our algorithm BAMBINO (**B**ayesian **A**pproach for **M**ultiobjective **Bi**level Problems with **N**oisy **O**bjectives). To evaluate its performance, we consider three multi-dimensional test problems from two different suites of multi-objective bilevel optimization problems. Also, one environmental economics problem which is a decision-making problem between an authority and a gold mining company. All examples illustrate the importance of taking uncertainty into account.

Most studies in the multi-objective bilevel optimization literature focus on solving the optimization problem without addressing the impact of uncertainty. In practical problems, noise in the leader's objectives might represent environmental uncertainty, for example, in a meta-learning regime [1] that can be mathematically formulated as bilevel programming [13]. As another example, a government might need to prevent terrorist attacks using information from unreliable sources. Yet another example occurs in computing optimal recovery policies for financial markets [18], using bilevel optimization with objective uncertainties caused by several uncontrollable parameters. Bilevel optimization problems are computationally expensive to solve because of their nested structure, and they become even more complex when there are multiple objectives and uncertainty (possibly at both levels). The main purpose of our work is to improve the efficiency of solving multi-objective bilevel optimization while handling leader objective uncertainties.

2 Background

We now provide some necessary background.

* Corresponding Author. Email: vedat.dogan@cs.ucc.ie.

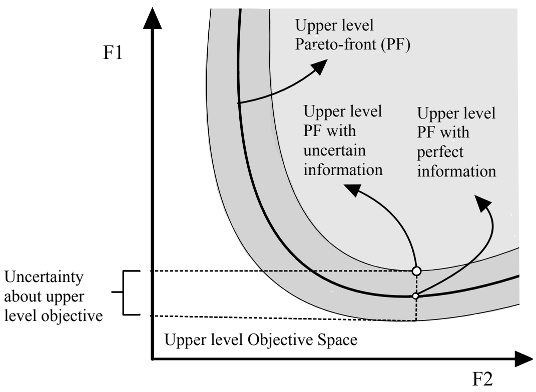

Figure 1. Decision making under upper-level uncertainty.

Algorithm 1 BAMBINO

Inputs: $\mathbf{F}_u(\mathbf{x}_u, \mathbf{x}_l) : \mathbf{x_u} \in X_u, \mathbf{x_l} \in X_l$,
Batch points in each iteration Q,
The number of iterations for BO: N,
Reference point

- $\mathbf{x_l}$: Find the Best Lower Level response as parameters with NSGA-II algorithm,
- Initial decision data set with the objective noise $D = \mathbf{x}_{u_i}, \mathbf{F}_u(\mathbf{x}_{u_i}, \mathbf{x}_{l_i}), (\Sigma_i)_{i=1}^n$ with size of n,
- Initialize the $\mathcal{GP}$ model with the observations and the objective noise
- **for** i = 0 : N **do**
  - Suggest new q-batch points by optimizing *qNEHVI*
  - **for** j = 0 : q **do**
    - For each upper-level decision $\mathbf{x}_u$, find optimal $\mathbf{x}_l^*$ by applying the NSGA-II
    - Calculate fitness scores with noise $\mathbf{F}_u(\mathbf{x}_u, \mathbf{x}_l^*) + \xi$
  - **end for**
  - Update the data set D with new observations
- **end for**
- **Return** Pareto-optimal front $\mathbf{F}_u^*$ and $(\mathbf{x}_u, \mathbf{x}_l^*)$

2.1 *Bilevel Multi-objective Optimization Problems (BMOP).*

Because of the nature of multi-objective optimization problems, only Pareto-optimal solutions at the lower level can be considered as feasible solutions for the upper-level problem. We denote the decision variables at the upper level by $x_u \in X_u \subset \mathbb{R}^n$ and the decision variables at the lower level by $x_l \in X_l \subset \mathbb{R}^m$. The lower-level problem is solved with respect to x_l while the upper-level problem is solved with respect to both decisions $x = (x_u, x_l)$. Each x_u corresponds to a different lower-level optimization problem with a different Pareto-optimal front decision set. The lower level Pareto-optimal front is defined as $P^* = \{f(\mathbf{x}_u, \mathbf{x}_l) : \mathbf{x}_l \in \mathbf{X}_l, \nexists \mathbf{x}'_l \in X_l\}$ s.t. $f(x') \succ f(x)$ where $f(x') \succ f(x)$ denotes $f(x')$ *dominates* $f(x)$. The Pareto-optimal decision set is $X_l^* = \{\mathbf{x}_l^* : f(\mathbf{x}_u, \mathbf{x}_l^*) \in P^*\}$. The definition of a bilevel multi-objective problem with vector-valued decision variables $\mathbf{x_u}$ and $\mathbf{x_l}$ is given by

$$\begin{aligned} &\underset{\mathbf{x}_u, \mathbf{x}_l}{\text{minimize}} \quad F_1(\mathbf{x}_u, \mathbf{x}_l), ..., F_p(\mathbf{x}_u, \mathbf{x}_l) \\ &\text{subject to} \\ &\mathbf{x}_l \in \underset{\mathbf{x}_l}{\text{argmin}} \left\{ f_1(\mathbf{x}_u, \mathbf{x}_l), ..., f_q(\mathbf{x}_u, \mathbf{x}_l); \right. \\ &\qquad\qquad \left. g_j(\mathbf{x}_u, \mathbf{x}_l) \le 0, \ j = 1, 2, \dots, J \right\} \\ &G_k(\mathbf{x}_u, \mathbf{x}_l) \le 0, \ k = 1, 2, \dots, K \end{aligned} \tag{1}$$

where $\mathbf{F} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p$ represents the upper level function and $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^q$ represents the lower level function of the bilevel problem. Upper level and lower level constraints are defined by $G_k : X_u \times X_l \rightarrow \mathbb{R}$ and $g_j : X_u \times X_l \rightarrow \mathbb{R}$ for $k = 1, \dots, K$ and $j = 1, \dots, J$.

2.2 *Bayesian Optimization (BO).*

BO is a sample-efficient approach that has demonstrated great potential in approximating a global optimum with a relatively small number of function evaluations. It uses a probabilistic surrogate model to make decisions by balancing exploration and exploitation [20]. Gaussian process ($\mathcal{GP}$) is a common surrogate model with a flexible and non-parametric form. $\mathcal{GP}$ provides a posterior distribution for a decision point x in the search space by capturing the prior belief about the performance of the unknown objective function, using a mean function $\mu(\mathbf{x})$ and a kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$. BO uses an acquisition function to decide which point to choose next. The acquisition function specifies the value of the next point by using the surrogate's predictive distribution at the current point. We assume that the black-box function f is expensive to evaluate, but that optimizing the acquisition function is relatively cheap and fast.

Multi-objective Bayesian optimization (MOBO) combines the Bayesian surrogate model and an acquisition function specifically designed for multi-objective optimization problems such as qNEHVI [6]. This is a hypervolume improvement-based acquisition function that works well for noisy multi-objective optimization problems.

3 Method

We consider a case that is crucial in practice, in which the leader must make decisions under uncertainty based on noisy observations $F_i = \mathbf{f}(\mathbf{x}_{u_i}, \mathbf{x}_{l_i}) + \xi_i$ where $\xi_i \sim \mathcal{N}(0, \Sigma_i)$ and Σ_i is the noise covariance and $\mathbf{x}_u, \mathbf{x}_l$ are upper and lower decision variables respectively. We reformulate the leader's objective with noisy observations as

$$\underset{\mathbf{x}_u, \mathbf{x}_l}{\text{minimize}} \{F_1(\mathbf{x}_u, \mathbf{x}_l) + \xi_1, ..., F_i(\mathbf{x}_u, \mathbf{x}_l) + \xi_i\} \tag{2}$$

where $\xi_i \sim \mathcal{N}(0, \Sigma_i)$. The hypervolume indicator measures the volume of space between the non-dominated front and a reference point, which we assume is known by the upper-level decision-maker. The selection of reference points is tricky. In this work, it is chosen to be an extreme point of the Pareto front, because reference points should be dominated by all Pareto-optimal solutions.

Hypervolume improvement of a set of points $\mathcal{P}'$ is defined as $HVI(\mathcal{P}'|\mathcal{P}, \mathbf{r}) = HV(\mathcal{P} \cup \mathcal{P}'|\mathbf{r}) - HV(\mathcal{P}|\mathbf{r})$ where $\mathcal{P}$ represents the Pareto front and $\mathbf{r}$ the reference point. Given observations of the upper-level decision-making process, the $\mathcal{GP}$ surrogate model provides us with a posterior distribution over the upper-level function values for each observation. These values can be used to compute the expected hypervolume improvement acquisition function defined by

$$\alpha_{ehvi}(\mathbf{x}_u|\mathcal{P}) = \mathbb{E}[HVI(\mathbf{F}_u|\mathcal{P})] \tag{3}$$

So the expected hypervolume improvement iterates over the posterior distribution, an approach that worked well in [6].

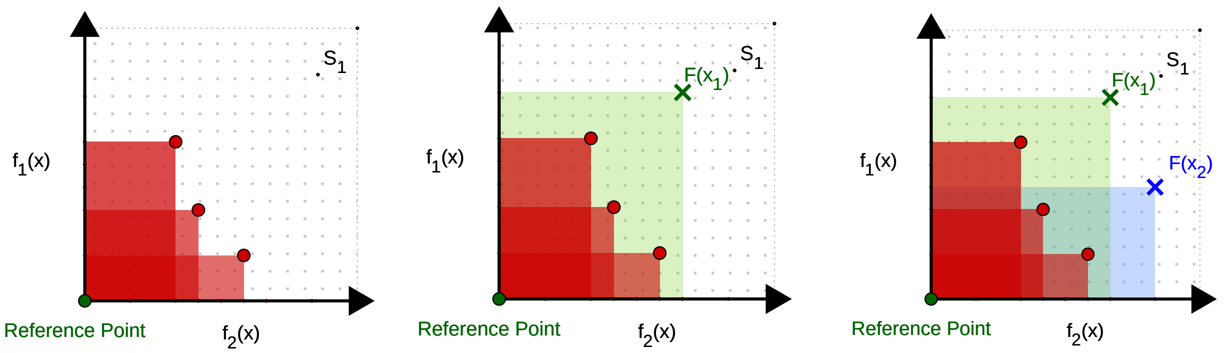

Figure 2. An illustration of the dominated (red) and non-dominated (white) space. The green and blue area on the graphs represents the hypervolume improvement of the new points.

After n observations of the leader's decisions and the follower's response, the posterior distribution can be defined by the conditional probability $p(\mathbf{F}(\mathbf{x}_{u_n}, \mathbf{x}_{l_n})|\mathcal{D}_n)$ of the leader's objective values given decision variables $(\mathbf{x}_{u_n}, \mathbf{x}_{l_n})$ based on noisy observations $\mathcal{D}_n = \mathbf{x}_{u_i}, \mathbf{F}_i(\mathbf{x}_{u_i}, \mathbf{x}_{l_i}), (\Sigma_i)_{i=1}^n$. NEHVI is defined as

$$\alpha_{NEHVI}(\mathbf{x}_u) = \int \alpha_{ehvi}(\mathbf{x}_u|P_n)p(\mathbf{F}|\mathcal{D}_n)d\mathbf{F} \tag{4}$$

where P_n denotes the Pareto-optimal front optimal decision set over the leader's objectives $\mathbf{F}_n$. The aim is to improve the efficiency of the optimization, and the handling of noise in the leader's objective, by using the approach above and reformulating the bilevel multi-objective optimization problem. The algorithm details can be found in Algorithm 1.

4 Experiments

The test problems are selected from the literature [10], with the aim of testing scalability in terms of decision variable dimensionality. The results are compared with state-of-art evolutionary algorithms m-BLEAQ [21] and H-BLEMO [8]. The Pareto-optimal front is independent of the parameters. Also, we use a real-world problem from environmental economics literature which considers a hierarchical decision-making problem between an authority and a gold mining company [22].

Performance Metrics. We compare our results in terms of upper-level function evaluations (FE) to determine the efficiency of the algorithm as Bayesian optimization aims to minimize the function evaluations while optimizing the expensive black-box functions. Hypervolume improvement (HV) [12] and inverted generational distance (IGD) [5] is also used to evaluate the success of approximation to Pareto-optimal fronts, in terms of convergence and diversity. HV measures the volume of the space between the non-dominated front obtained and a reference point. IGD calculates the sum of the distances from each point of the true Pareto-optimal front to the nearest point of the non-dominated set found by the algorithm. Therefore, a smaller IGD value means approximated points are closer to the Pareto-optimal front of the problem.

Parameters. We fixed the number of Bayesian optimization iterations to $N = 50$ and repeated our experiments 21 times to obtain median results for making the comparison fair. We use the independent $\mathcal{GP}$ model with *Matern52* kernel and fit the $\mathcal{GP}$ by maximizing the marginal log-likelihood. The method is initialized with $2 \times (d+1)$ Sobol points where d represents the dimension of the problem to construct the initial $\mathcal{GP}$ model. All experiments are conducted using the BoTorch [2] library. We solved the follower's problem with the popular *non-dominated sorted genetic algorithm* (NSGA-II) [7] and choose the population size 100 and number of generations 200. We choose the follower's decisions from the obtained Pareto-optimal front at random, as all solutions in the Pareto-optimal front are feasible.

4.1 Test Problems

Example 1. The first example is a bi-objective problem that is scalable in terms of the number of follower decision variables. We choose $K = 14$ and $K = 19$, giving 15 and 20 follower variables respectively, with 1 leader decision variable. We choose the reference point required to measure hypervolume improvement to be $(1.0, 0.5)$. The Pareto-optimal decision sets for this specific bilevel decision-making problem can be found in [21].

Example 2. The second test problem is the modified test problem with 10 and 20 variable instances. We choose the required reference point to be $(1.1, 1.1)$. The Pareto-optimal front for a given leader is defined as a circle of radius $(1+r)$ with centre $((1+r),(1+r))$. We choose $K = 5$ for our experiments with parameters $r = 0.1, \tau = 1$ and $\alpha = 1$, following [21] so that our results can be compared with those for m-BLEAQ and H-BLEMO.

Example 3. The third test problem is the modified test problem with 10 and 20 variable instances. We choose the required reference point $(0.8, 0.0)$ for measuring the hypervolume improvement during the optimization. Details on the Pareto-optimal solutions are given in [9].

4.1.1 Results of Test Problems

The performance of BAMBINO is compared with that of m-BLEAQ and H-BLEMO in Table 2, showing computational expense and convergence. The FE is calculated by $N_{initial} + (N_{batch} \times N_{iter} \times N_{restarts})$ for the leader problem,

Table 1. Selected test problem from literature for multi-objective bilevel optimization.

Problem	Formulation
Example 1 n = 1 m = K	$\underset{(x_u,\mathbf{x_l})}{\text{Min}}\ \mathbf{F}(x_u,\mathbf{x_l}) = \begin{pmatrix} (x_{l_1}-1)^2 + \sum_{i=2}^{K} x_{l_i} + (x_u)^2 + \xi \\ (x_{l_1}-1)^2 + \sum_{i=2}^{K} x_{l_i} + (x_u-1)^2 + \xi \end{pmatrix}$ subject to $\mathbf{x}_l \in \underset{\mathbf{x_l}}{\text{argmin}}\ \mathbf{f}(x_u,\mathbf{x_l}) = \begin{pmatrix} x_{l_1} + \sum_{i=2}^{K} x_{l_i}^2 \\ x_{l_1} - x_u + \sum_{i=2}^{K} x_{l_i}^2 \end{pmatrix}$ $-1 \leq (x_u, x_{l_1}, x_{l_2}, ..., x_{l_K}) \leq 2$ $\xi \sim \mathcal{N}(0,\Sigma_\xi),\ \Sigma_\xi = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$
Example 2 n = K m = K	$\underset{(\mathbf{x_u},\mathbf{x_l})}{\text{Min}}\ \mathbf{F}(\mathbf{x_u},\mathbf{x_l}) = \begin{pmatrix} (1+r-\cos(\alpha\pi x_{u_1})) + \sum_{j=2}^{K}(x_{u_j} - \frac{j-1}{2})^2 + \tau\sum_{i=2}^{K}(x_{l_i}-x_{u_i})^2 - r\cos(\gamma\frac{\pi}{2}\frac{x_{l_1}}{x_{u_1}}) + \xi \\ (1+r-\sin(\alpha\pi x_{u_1})) + \sum_{j=2}^{K}(x_{u_j} - \frac{j-1}{2})^2 + \tau\sum_{i=2} K(x_{l_i}-x_{u_i})^2 - r\sin(\gamma\frac{\pi}{2}\frac{x_{l_1}}{x_{u_1}} + \xi) \end{pmatrix}$ subject to $\mathbf{x}_l \in \underset{\mathbf{x_l}}{\text{argmin}}\ \mathbf{f}(\mathbf{x_u},\mathbf{x_l}) = \begin{pmatrix} x_{l_1}^2 + \sum_{i=2}^{K}(x_{l_i}-x_{u_i})^2 + \sum_{i=2}^{K} 10(1-\cos(\frac{\pi}{K}(x_{l_i}-x_{u_i})) \\ \sum_{i=2}^{K}(x_{l_i}-x_{u_i})^2 + \sum_{i=2}^{K} 10\lvert\sin(\frac{\pi}{K}(x_{l_i}-x_{u_i}))\rvert \end{pmatrix}$ $x_{l_i} \in [-K,K], i = 1,\dots,K$ $x_{u_1} \in [1,4], x_{u_j} \in [-K,K], j = 2,\dots,K$ $\xi \sim \mathcal{N}(0,\Sigma_\xi),\ \Sigma_\xi = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.16 \end{bmatrix}$
Example 3 n = K m = K	$\underset{(\mathbf{x_u},\mathbf{x_l})}{\text{Min}}\ \mathbf{F}(\mathbf{x_u},\mathbf{x_l}) = \begin{pmatrix} v_1(x_{u_1}) + \sum_{j=2}^{K}(x_{u_j}^2 + 10(1-\cos(\frac{\pi}{K}x_{u_i})) + \tau\sum_{i=2}^{K}(x_{l_i}-x_{u_i})^2 - r\cos(\gamma\frac{\pi}{2}\frac{x_{l_1}}{x_{u_1}}) + \xi \\ v_2(x_{u_1}) + \sum_{j=2}^{K}(x_{u_j}^2 + 10(1-\cos(\frac{\pi}{K}x_{u_i})) + \tau\sum_{i=2}^{K}(x_{l_i}-x_{u_i})^2 - r\sin(\gamma\frac{\pi}{2}\frac{x_{l_1}}{x_{u_1}}) + \xi \end{pmatrix}$ where $v_1(x_{u_1}) = \begin{cases} \cos(0.2\pi)x_{u_1} + \sin(0.2\pi)\sqrt{\lvert 0.02\sin(5\pi x_{u_1})\rvert}, & \text{for } 0 \leq x_{u_1} \leq 1 \\ x_{u_1} - (1-\cos(0.2\pi)), & \text{for } x_{u_1} \geq 1 \end{cases}$ $v_2(x_{u_1}) = \begin{cases} -\sin(0.2\pi)x_{u_1} + \cos(0.2\pi)\sqrt{\lvert 0.02\sin(5\pi x_{u_1})\rvert}, & \text{for } 0 \leq x_{u_1} \leq 1 \\ 0.01(x_{u_1}-1) - \sin(0.2\pi), & \text{for } x_{u_1} \geq 1. \end{cases}$ subject to $\mathbf{x}_l \in \underset{\mathbf{x_l}}{\text{argmin}}\mathbf{f}(\mathbf{x_u},\mathbf{x_l}) = \begin{pmatrix} x_{l_1}^2 + \sum_{i=2}^{K}(x_{l_i}-x_{u_i})^2 \\ \sum_{i=2}^{K} i(x_{l_i}-x_{u_i})^2 \end{pmatrix}$ $x_{l_i} \in [-K,K], i = 1,\dots,K$ $x_{u_1} \in [0.001,K], x_{u_j} \in [-K,K], j = 2,\dots,K$ $\xi \sim \mathcal{N}(0,\Sigma_\xi),\ \Sigma_\xi = \begin{bmatrix} 0.09 & 0 \\ 0 & 0.09 \end{bmatrix}$

where $N_{initial}$ is the number of initial decisions for starting the algorithm and $N_{restarts}$ is the parameter for Gaussian process declares the number of restart to avoid to stuck at local optima. We choose it to be $2 \times (d+1)$ where d is the dimensions of the decision variable. We run the experiment for different batch numbers $q = 1, q = 2, q = 4, q = 8$ to test the effect on performance. The HV difference is shown in Figure 3 for 15 and 20 variables, and 10 variables for Examples 1 and 2 respectively. While increasing the batch size, decreasing the HV difference as presented in Figure 2 shows the convergence of the proposed algorithm. Because of lack of information in the reference paper, we could not obtain the FE results for Example 1 with 20 variables.

Example 1. We can see from Table 2 that the required upper-level FE is significantly lower, while the algorithm approximates successfully to the Pareto-optimal front while handling the uncertainty at the leader's objective. For 15 variables BAMBINO achieves $\approx$ %38 improvement in terms of FE compared to m-BLEAQ and $\approx$ %89 compared to H-BLEMO. The IGD values in Table 2 for 15 and 20 variables show that BAMBINO successfully approximates the Pareto-optimal front of the problem while handling the uncertainty of the leader's objective for both. We show the HV difference between Pareto-optimal front solutions and approximated BAMBINO decisions algorithm in Figure 3. Again we tried different batch sizes for the experiment, and it can be seen that a batch number of 8 is best for this specific example at both dimensions. We could not compare the 20 dimensional version of the problem with the selected algorithms because of the lack of information in [21].

Example 2. Table 2 shows that BAMBINO obtains the best IGD results compared to the other algorithms. In terms of FE, it significantly improves state of the art, with $\approx$ %81 improvement for 10 variables and $\approx$ %88 improvement for 20 variables compared to m-BLEAQ. We also show the HV difference in Figure 3 and we can observe that, for this specific example, the batch number of 8 is the best selection for both 10 and 20 dimensional versions.

Example 3. Table 2 shows that BAMBINO obtains the best IGD value compared to m-BLEAQ and H-BLEMO while improving efficiency in terms of FE: $\approx 84\%$ and $\approx 97\%$ with 10 variables, and $\approx 89\%$ and $\approx 98\%$ with 20 variables. Figure 3shows that batch size $q = 8$ gives the best results.

In summary, BAMBINO is successful on the selected test problems while handling noisy objectives with less computational cost. Noise

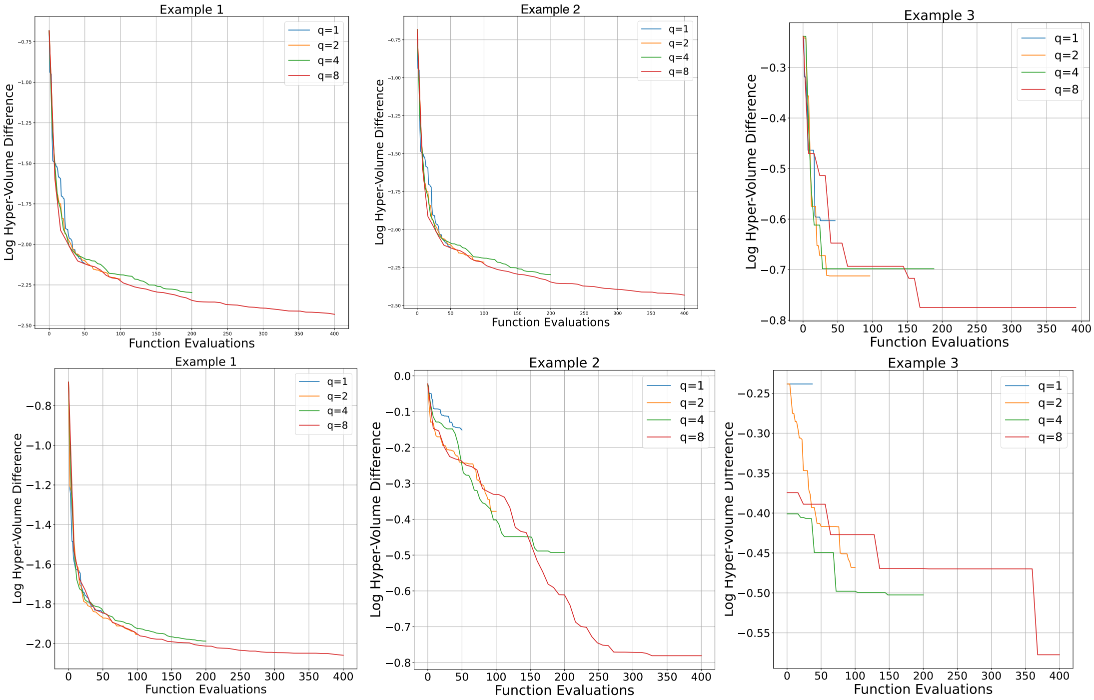

Figure 3. Hypervolume difference graph (log scale) with different batch sizes ($q = 1, q = 2, q = 4, q = 8$) for Example 1 with 15 (*top-left*) and 20 (*bottom-left*) dimensions, for Example 2 with 10 (*top-middle*) and 20 (*bottom-middle*) dimensions, for Example 3 with 10 (*top-right*) and 20 (*bottom-right*) dimensions.

Table 2. FE and IGD values for the examples with the number of variable dimensions for the batch size $q = 8$.

	Number of Variables	BAMBINO		m-BLEAQ		H-BLEMO	
		IGD	FE	IGD	FE	IGD	FE
Example 1	15	0.0044	4032	0.0013	6,464	0.0046	39,818
Example 2	10	0.0051	4022	0.0069	22,223	0.0134	106,003
Example 3	10	0.0076	4022	0.0079	25,364	0.0134	132,907
Example 1	20	0.0105	4042	-	-	-	-
Example 2	20	0.1032	4042	0.0435	34,110	0.1106	191,357
Example 3	20	0.0924	4042	0.0623	36,439	0.1321	216,083

in the leader's objective makes the problem harder to solve but more realistic for modelling practical problems, because of real-world uncertainty. We show the proposed BAMBINO algorithm works well on these test benchmark problems.

4.2 Practical Case: Gold Mining in Kuusamo

Kuusamo region is a popular tourist destination known for its natural beauty. There is a lot of interest in this region cause of containing a huge amount of gold deposits, considered to be a "highly prospective Paleoproterozoic Kuusamo Schist Belt" [4]. The expected gold amount in the ore is around 4.9 g per ton according to an Australia-based gold mining company. Even though there is a big potential in terms of providing lots of jobs in the region and leading to a great amount of gold resources, there are some concerns about harming the environment. The first of them is that mining operations may cause pollution of the river water in the region. It causes fear for environmentalists. Second, the ore in the region contains uranium and if it is mined, it might decrease the reputation of the tourist resorts around. Another one is the open pit mines around the area called Ruka will cause a turn-off for skiing resorts and hiking routes around and it will decrease the tourist interest.

The regulating authority, which is government, acts as a leader and the mining company is the follower which reacts rationally to the decisions of the leader in order to maximize its own profit. The leader should find an optimal strategy assuming that he holds the necessary information about the follower. In the situation explained above, the government has a decision-making problem which is whether to allow mining and to what extent. In the problem above, the leader has two objectives while the follower has one. The first objective is maximization of revenues coming from the project and the second objective is to minimize the environmental harm which is a result of the mining. The mining company also aims to maximize its own profit. While the government is optimizing its own taxation strategy, it needs to model how the mining company reacts to any given tax structure. Therefore, the authority makes an environmental regulatory decision instead of solving the problem to optimality. Clearly, the objectives are conflicting such as large profits may affect the environment by increasing the damage which follows with a bad public image. The mathematical formulation of this hierarchical decision-making problem can be found in Table 3. More details can be found about the bilevel modelling of the problem in [22] and [21]. Figure 4 presents the Pareto-optimal frontier of the given problem for the government according to the formulation in Table 3. We can observe

Table 3. Gold Mining in Kuusamo

Category	Level	Formulation
Variables	Upper level	τ [per unit tax imposed on the mine]
	Lower level	q [amount of metal extracted by the mine]
Objectives	Upper level	$F_1(\tau, q) = E(\tau, q) = \tau q$ [tax revenue]
		$F_2(\tau, q) = -D(q) = -kq$ [environmental damage]
	Lower level	$f_1(\tau, q) = \pi(\tau, q) = (\alpha - \beta q)q - (\delta q^2 + \gamma q + \phi) - \tau q$ [profit]
Constraints	Upper level	$q \geq 0, \tau \geq 0$
	Lower level	$\pi(\tau, q) \geq 0$
Uncertainty	Upper level	$\xi \sim N(\mu_\xi, \Sigma_\xi), \mu_\xi = (1, 1), \Sigma_\xi = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix}$
Parameters		$k = 1, \eta = 1$
		$\alpha = 100, \beta = 1, \delta = 1, \gamma = 1, \phi = 0$

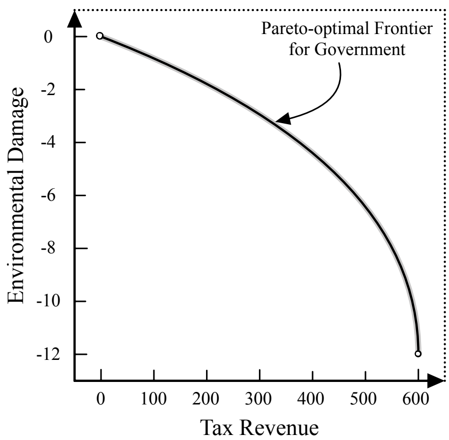

Figure 4. Pareto-optimal frontier for the government representing the trade-off between tax revenues and environmental pollution.

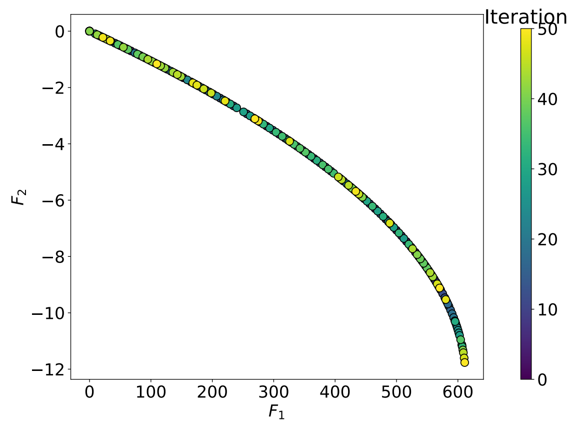

Figure 5. Pareto-optimal frontier for the government representing the trade-off between tax revenues and environmental pollution.

that increasing the tax revenue decreases environmental damage.

4.2.1 Results of Practical Case

In this section, we present the results obtained using the BAMBINO algorithm on the analytical model of the problem proposed in Table 3. Figure 5 shows the Pareto-optimal front obtained using the BAMBINO approach. The plot gives the idea to the authority how to consider the trade-off between its own objectives. We used 50 iterations for upper-level optimization using a batch Bayesian approach with a batch size of 4. For lower-level optimization, the NSGA-II algorithm is implemented with the same parameters specified in Section 4. We can observe that with the proposed method, the obtained results are distributed around the true Pareto-optimal frontier and are approximated to it successfully. We also run the experiments for the batch size $q = 1$,$q = 2$ and $q = 4$ for testing if the algorithm converges to the Pareto-optimal front. The IGD value is 0.0494 for $q = 1$, 0.0033 for $q = 2$ and 0.0021 for $q = 4$. We can see from the decreasing IGD values, as we increase the batch number, the selected points are more approximated to the true Pareto-optimal front. The increasing batch size provides us with parallel evaluations during upper-level optimization which decreases the needed execution time. It is also good to handle environmental uncertainty, represented by ξ in Table 3, which may be an uncontrollable parameter such as inflation during the time period of taxation or unexpected environmental damage during the mining process. We can observe that the proposed method is successfully approximated even handling the uncertainty of objectives.

We believe that BAMBINO can be applied to several practical bilevel problems successfully applied in the machine learning community, such as image classification [19], deep learning [14], neural networks [16], and hyperparameter optimization [17].

5 Conclusions

In this paper, we discussed bilevel multi-objective optimization under upper-level uncertainty and presented a hybrid algorithm called BAMBINO, based on batch Bayesian optimization with hypervolume improvement. We ran experiments using three benchmark problems and one real-world problem from environmental economics literature which is a decision-making problem between an authority and a gold mining company. BAMBINO performed very competitively in terms of computational efficiency and convergence. We also showed how batch size selection affects performance in terms of hypervolume improvement.

Limitations and Future Work. Multi-objective bilevel problems are expensive to evaluate and time-consuming most of the time. Even with the uncertainty at the upper-level, it might cause a lack of performance during the acquisition optimization process of the proposed algorithm. Also, the Pareto-optimal solution set does not always provide the optimal solution for all objectives. However, still approximating the Pareto-front with a black-box approach improves the performance compared with various existing methods as we can see from Section 4.1.1. In future work, we shall explore the decision-making from the Pareto-optimal front and how preference learning might affect the process. Moreover, we will look to gain some information by analyzing the relationship between acquisition selection and optimization results.

Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 12/RC/2289-P2 and 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] Maruan Al-Shedivat, Trapit Bansal, Yuri Burda, Ilya Sutskever, Igor Mordatch, and Pieter Abbeel, 'Continuous adaptation via meta-learning in nonstationary and competitive environments', *CoRR*, **abs/1710.03641**, (2017).

[2] Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy, 'Botorch: Programmable bayesian optimization in pytorch', *CoRR*, **abs/1910.06403**, (2019).

[3] Jerome Bracken and James T. McGill, 'Mathematical programs with optimization problems in the constraints', *Operations Research*, **21**(1), 37–44, (1973).

[4] Latitude Cobalt. Latitude 66 cobalt oy reports a new cobalt-gold discovery in kuusamo, finland highlights, 2023.

[5] Carlos A Coello Coello and Margarita Reyes Sierra, 'Lnai 2972 - a study of the parallelization of a coevolutionary multi-objective evolutionary algorithm', *Mexican International Conference on Artificial Intelligence*, (2004).

[6] Sam Daulton, Maximilian Balandat, and Eytan Bakshy, 'Parallel bayesian optimization of multiple noisy objectives with expected hypervolume improvement', in *NeurIPS*, (2021).

[7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, 'A fast and elitist multiobjective genetic algorithm: Nsga-ii', *IEEE Transactions on Evolutionary Computation*, **6**(2), 182–197, (2002).

[8] Kalyan Deb and Ankur Sinha, 'An efficient and accurate solution methodology for bilevel multi-objective programming problems using a hybrid evolutionary-local-search algorithm', *Evolutionary computation*, **18**, 403–49, (09 2010).

[9] Kalyanmoy Deb and Ankur Sinha, 'Constructing test problems for bilevel evolutionary multi-objective optimization', in *2009 IEEE Congress on Evolutionary Computation*, pp. 1153–1160, (2009).

[10] Kalyanmoy Deb and Ankur Sinha, 'Solving bilevel multi-objective optimization problems using evolutionary algorithms', in *EMO*, (2009).

[11] Gabriele Eichfelder, 'Multiobjective bilevel optimization', *Math. Program.*, **123**, 419–449, (06 2010).

[12] C.M. Fonseca, L. Paquete, and M. Lopez-Ibanez, 'An improved dimension-sweep algorithm for the hypervolume indicator', in *2006 IEEE International Conference on Evolutionary Computation*, pp. 1157–1163, (2006).

[13] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil, 'Bilevel programming for hyperparameter optimization and meta-learning', in *Proceedings of the 35th International Conference on Machine Learning*, eds., Jennifer Dy and Andreas Krause, volume 80 of *Proceedings of Machine Learning Research*, pp. 1568–1577. PMLR, (10–15 Jul 2018).

[14] Ya-nan Han, Jianwei Liu, Bing-biao Xiao, Xin-Tan Wang, and Xionglin Luo, 'Bilevel online deep learning in non-stationary environment', *CoRR*, **abs/2201.10236**, (2022).

[15] Md Monjurul Islam, Hemant Kumar Singh, and Tapabrata Ray, 'A nested differential evolution based algorithm for solving multi-objective bilevel optimization problems', in *ACALCI*, (2016).

[16] Hong Li and Li Zhang, 'A bilevel learning model and algorithm for self-organizing feed-forward neural networks for pattern classification', *IEEE Transactions on Neural Networks and Learning Systems*, **32**, 4901–4915, (2021).

[17] Risheng Liu, Jiaxin Gao, Jin Zhang, Deyu Meng, and Zhouchen Lin, 'Investigating bi-level optimization for learning and vision from a unified perspective: A survey and beyond', *CoRR*, **abs/2101.11517**, (2021).

[18] Carlo Mannino, Fred Bernt, and Geir Dahl, 'Computing optimal recovery policies for financial markets', *Operations Research*, **60**, (12 2012).

[19] Saypraseuth Mounsaveng, Issam H. Laradji, Ismail Ben Ayed, David Vázquez, and Marco Pedersoli, 'Learning data augmentation with online bilevel optimization for image classification', *CoRR*, **abs/2006.14699**, (2020).

[20] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando de Freitas, 'Taking the human out of the loop: A review of bayesian optimization', *Proceedings of the IEEE*, **104**(1), 148–175, (2016).

[21] Ankur Sinha, Pekka Malo, Kalyanmoy Deb, Pekka Korhonen, and Jyrki Wallenius, 'Solving bilevel multicriterion optimization problems with lower level decision uncertainty', *IEEE Transactions on Evolutionary Computation*, **20**(2), 199–217, (2016).

[22] Ankur Sinha, Pekka Malo, Anton Frantsev, and Kalyanmoy Deb, '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).

[23] Heinrich von. Stackelberg, *The theory of the market economy*, William Hodge, London, 1952.