Solving Optimistic Bilevel Programs by Iteratively Approximating Lower Level Optimal Value Function

Ankur Sinha[1], Pekka Malo[2], and Kalyanmoy Deb[3]

[1]Production and Quantitative Methods
Indian Institute of Management, Ahmedabad 380015 India
asinha@iimahd.ernet.in

[2]Department of Information and Service Economy
Aalto University School of Business, PO Box 21220, 00076 Aalto, Finland
pekka.malo@aalto.fi

[3]Department of Electrical and Computer Engineering
Michigan State University, East Lansing, MI, USA
kdeb@egr.msu.edu

COIN Report: 2016015

Abstract

Bilevel optimization is a nested optimization problem that contains one optimization task as a constraint to another optimization task. Owing to enormous applications that are bilevel in nature, these problems have received attention from mathematical programming as well as evolutionary optimization community. However, most of the available solution methods can either be applied to highly restrictive class of problems, or are highly computationally expensive that they do not scale for large scale bilevel problems. The difficulties in bilevel programming arise primarily from the nested structure of the problem. In this paper, we propose a metamodeling based solution strategy that attempts to iteratively approximate the optimal lower level value function. To the best knowledge of the authors, this kind of a strategy has not been used to solve bilevel optimization problems, particularly in the context of evolutionary computation. The proposed method has been evaluated on a number of test problems from the literature.

1 Introduction

Bilevel optimization is characterized as a mathematical program that involves two levels of optimization. The outer optimization task is commonly referred to as the upper level optimization problem and the inner optimization task is commonly referred to as the lower level optimization problem. Bilevel optimization has two roots: these problems were first realized by Stackelberg [40] in the area of game theory and came to be known as Stackelber games; later these problems were realized in the area of mathematical programming by Bracken and McGill [13] as a constrained optimization task, where the lower level optimization problem acts as a constraint to the upper level optimization problem. These problems are known to be difficult due to its nested structure; therefore most attention has been given to simple cases where the objective functions and constraints are linear [48, 10], quadratic [8, 20, 1] or convex [30].

An interest in bilevel programming has been driven by a number of new applications arising in different fields of optimization. For instance, in the context of homeland security [15, 47, 3], bilevel and even trilevel optimization models are common. In game theoretic settings, bilevel programs have been used in the context of optimal tax policies [26, 38, 37]; model production processes [34]; investigation of strategic behavior in deregulated markets [23] and optimization of retail channel structures [49], among others. Bilevel optimization applications are ubiquitous and airse in many other disciplines, like in transportation [32, 19, 14], management [41, 9], facility location [25, 43, 41], chemical engineering [39, 18], structural optimization [11, 17], and optimal control [33, 2] problems.

Evolutionary computation [6] techniques have been successfully applied to handle mathematical programming problems and applications that do not adhere to regularities like continuity, differentiability or convexities. Due to these properties of evolutionary algorithms, attempts have been made to solve bilevel optimization problems using these methods, as even simple (linear or quadratic) bilevel optimization problems are intrinsically non-convex, non-differentiable and disconnected at times. However, the advantages come with a trade-off, as the evolutionary techniques are computationally intensive and require large number of function evaluations to solve bilevel problems. In this paper, we make an attempt to reduce the computational expense of evolutionary bilevel optimization algorithms by utilizing a metamodeling-based principle that approximates the lower level optimal value function. The principle can be integrated with any evolutionary algorithm to handle bilevel optimization problems. We make comparisons with other algorithms studied in the past to demonstrate the effectiveness of the proposed strategy.

The paper is organized as follows. To begin with we provide a brief literature survey of bilevel optimization using evolutionary algorithms. Thereafter, we provide the general formulation and introduce the single-level reduction principle using lower level optimal value function. Then we incorporate the principle in an evolutionary algorithm and provide comparison results with earlier approachs. Finally, we provide the conclusions.

2 A Survey on Evolutionary Bilevel Optimization

Most of the evolutionary approaches proposed to handle bilevel optimization problems are nested in nature. As the name suggests, these approaches rely on two optimization algorithms, where one algorithm is nested within the other. Based on the complexity of the optimization tasks at each level, researchers have chosen to use either evolutionary algorithms at both levels or evolutionary algorithm at one level and classical optimization algorithm a the the other level. One of the earliest evolutionary agorithms for solving bilevel optimization problems was proposed in the early 1990s by Mathieu et al. [31] who used a nested approach with genetic algorithm at the upper level, and linear programming at the lower level. Later, Yin [51] solved genetic algorithm at the upper level and Frank-Wolfe algorithm (reduced gradient method) at the lower level. In both these approaches a lower level optimization task was executed for every upper level member that emphasizes the nested structure of these approaches. Along similar lines, nested procedures were used in [29, 28, 52]. Approaches with evolutionary algorithms at both levels are also common; for instance, in [4] authors used differential evolution at both levels, and in [5] authors nested differential evolution with an ant colony optimization.

In a number of studies, where lower level problem adhered to certain regularity conditions, researchers have used the KKT conditions for the lower level problem to reduce the bilevel problem into a single-level problem. The reduced single-level problem is then solved with an evolutionary algorithm. For instance, Hejazi et al. [22], reduced the linear bilevel problem to single-level and then used a genetic algorithm, where chromosomes emulate the vertex points, to solve the problem. Wang et al. [45] used KKT conditions to reduce the bilevel problem into single-level, and then utilized a constraint handling scheme to successfully solve a number of standard test problems. A later study by Wang et al. [46] introduced an improved algorithm that performed better than the previous approach [45]. Recently, Jiang et al. [24] reduced the bilevel optimization problem into a non-linear optimization problem with complementarity constraints, which is sequentially smoothed and solved with a PSO algorithm. Other studies using similar ideas are [27, 44].

It is noteworthy that utilization of KKT conditions restricts the algorithm's applicability to only a special class of bilevel problems. To overcome this drawback, researchers are looking into metamodeling based approaches where the lower level optimal reaction set is approximated over generations of the evolutionary algorithm. Studies in this direction are [35, 36]. Along similar lines, in this paper we attempt to metamodel the lower level optimal value function to solve bilevel optimization problems. Approximating the lower level optimal value function may offer a few advantages over approximating the lower level reaction set that are discussed in the later part of the paper.

3 Bilevel Formulation and Single-level Reductions

In this section, we provide a general formulation for bilevel optimization, and different ways people have used to reduce bilevel optimization problems to single-level problems. Bilevel problems contain two levels, upper and lower, where lower level is nested withing the upper level problem. The two levels have their own objectives, constraints and variables. In the context of game theory, the two problems are also referred to as the leader's (upper) and follower's problems (lower). The lower level optimization problem is a parametric optimization problem that is solved with respect to the lower level variables while the upper level variables act as parameters. The difficulty in bilevel optimization arises from the fact that only lower level optimal solutions can be considered as feasible members if they also satisfy the upper level constraints. Below we provide a general bilevel formulation:

Definition 1 *For the upper-level objective function $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and lower-level objective function $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, the bilevel optimization problem is given by*

$$\begin{aligned} &\underset{x_u \in X_U, x_l \in X_L}{\text{“min”}} F(x_u, x_l) \text{ subject to} \\ &\quad x_l \in \underset{x_l \in X_L}{\operatorname{argmin}} \{ f(x_u, x_l) : g_j(x_u, x_l) \le 0, j = 1, \ldots, J \} \\ &\quad G_k(x_u, x_l) \le 0, k = 1, \ldots, K \end{aligned}$$

where $G_k : X_U \times X_L \to \mathbb{R}$, $k = 1, \ldots, K$ denotes the upper level constraints, and $g_j : X_U \times X_L \to \mathbb{R}$ represents the lower level constraints, respectively.

3.1 Optimistic vs Pessimistic

Quotes have been used while specifying the upper level minimization problem in Definition 1 because the problem is ill-posed for cases where the lower level has multiple optimal solutions. There is lack of clarity as to which optimal solution from the lower level should be utilized at the upper level in such cases. It is common to take either of the two positions, i.e., optimistic or pessimistic, to sort out this ambiguity. In optimistic position some form of cooperation is assumed between the leader and the follower. For any given leader's decision vector that has multiple optimal solutions for the follower, the follower is expected to choose that optimal solution that leads to the best objective function value for the leader. On the other hand in a pessimistic position the leader optimizes for the worst case, where the follower may choose that solution from the optimal set which leads to the worst objective function value for the leader. Optimistic position being more tractable is commonly studied in the literature, and we handle the optimistic position in this paper.

3.2 KKT reduction

When the lower level problem in Definition 1 adheres to certain convexity and regularity conditions, it is possible to replace the lower level optimization task with its KKT conditions.

Definition 2 *The KKT conditions appear as Lagrangian and complementarity constraints in the single-level formulation provide below:*

$$\begin{aligned}
&\min_{x_u \in X_U, x_l \in X_L, \lambda} \quad F(x_u, x_l) \\
&\text{subject to} \\
&\qquad G_k(x_u, x_l) \leq 0, k = 1, \ldots, K, \\
&\qquad g_j(x_u, x_l) \leq 0, j = 1, \ldots, J, \\
&\qquad \lambda_j g_j(x_u, x_l) = 0, j = 1, \ldots, J, \\
&\qquad \lambda_j \geq 0, j = 1, \ldots, J, \\
&\qquad \nabla_{x_l} L(x_u, x_l, \lambda) = 0,
\end{aligned}$$

where

$$L(x_u, x_l, \lambda) = f(x_u, x_l) + \sum_{j=1}^{J} \lambda_j g_j(x_u, x_l).$$

The above formulation might not be simple to handle, as the Lagrangian constraints often lead to non-convexities, and the complementarity condition being combinatorial, make the overall problem a mixed integer problem. In case of linear bilevel optimization problems, the Lagrangian constraint is also linear. Therefore, the single-level reduced problem becomes a mixed integer linear program. Approaches based on vertex enumeration [12, 16, 42], as well as branch-and-bound [7, 21] have been proposed to solve these problems.

3.3 Reaction set mapping

An equivalent formulation of the problem given in Definition 1 can be stated in terms of set-valued mappings as follows:

Definition 3 *Let $\Psi : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ be the reaction set mapping,*

$$\Psi(x_u) = \underset{x_l \in X_L}{\operatorname{argmin}} \{ f(x_u, x_l) : g_j(x_u, x_l) \leq 0, j = 1, \ldots, J \},$$

which represents the constraint defined by the lower-level optimization problem, i.e. $\Psi(x_u) \subset X_L$ *for every* $x_u \in X_U$*. Then the bilevel optimization problem can be expressed as a constrained optimization problem as follows:*

$$\begin{aligned} \min_{x_u \in X_U, x_l \in X_L} \quad & F(x_u, x_l) \\ \text{subject to} \quad & \\ & x_l \in \Psi(x_u) \\ & G_k(x_u, x_l) \leq 0, k = 1, \ldots, K \end{aligned}$$

Note that if the Ψ-mapping can somehow be determined, the problem reduces to a single level constrained optimization task. However, that is rarely the case. Evolutionary computation studies that rely on iteratively mapping this set to avoid frequent lower level optimization are [35, 36].

3.4 Lower Level Optimal Value Function

Another equivalent definition of the problem in Definition 1 can be given in terms of the lower level optimal value function that is defined below [50]:

Definition 4 *Let* $\varphi : X_U \to R$ *be the lower level optimal value function mapping,*

$$\varphi(x_u) = \min_{x_l \in X_L} \{f(x_u, x_l) : g_j(x_u, x_l) \leq 0, j = 1, \ldots, J\},$$

which represents the minimum lower level function value corresponding to any upper level decision vector. Then the bilevel optimization problem can be expressed as follows:

$$\begin{aligned} \min_{x_u \in X_U, x_l \in X_L} \quad & F(x_u, x_l) \\ \text{subject to} \quad & \\ & f(x_u, x_l) \leq \varphi(x_u) \\ & g_j(x_u, x_l) \leq 0, j = 1, \ldots, J \\ & G_k(x_u, x_l) \leq 0, k = 1, \ldots, K. \end{aligned}$$

In this paper, we aim to approximate the φ-mapping iteratively during the generations of the evolutionary algorithm, and solve a reduced bilevel problem described in Definition 4. To our best knowledge, there does not exist an evolutionary algorithm that relies on approximating this mapping during the course of solving a bilevel optimization problem.

Approximating the optimal value function mapping might offer an advantage over approximating reaction set mapping, as the optimal value function mapping is not set valued. Moreover, it returns a scalar for any given upper level decition vector. However, in Definition 4 the resulting problem has to be solved with respect to both upper and lower level variables, while in Definition 3, the lower level variables are directly available from the Ψ-mapping. Therefore, there exists a trade-off.

4 Evolutionary Algorithm based on φ-mapping Approximation

In this section, we provide an implementation of the φ-mapping approximation within an evolutionary algorithm. The steps of the algorithm are provided through a flowchart in Figure 1. For

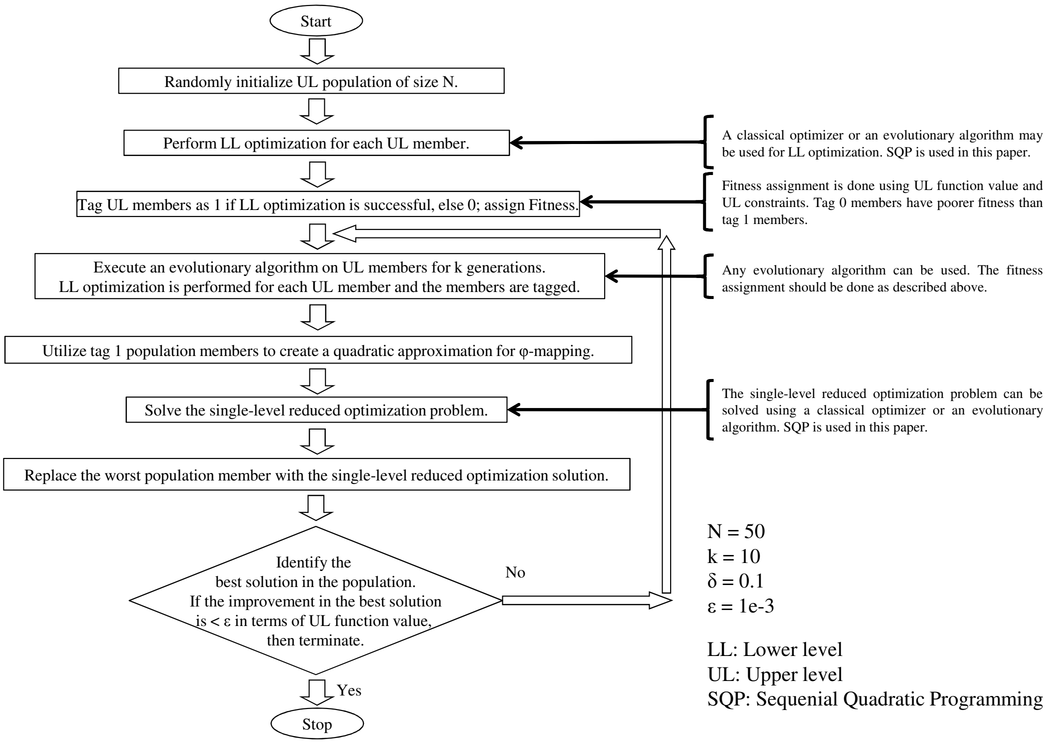

Figure 1: Flowchart for incorporating approximated φ-mapping in an evolutionary algorithm.

brevity, we do not discuss the steps of the evolutionary algorithm, as any scheme can be utilized in the provided framework to handle bilevel optimization problems. For our tests and comparisons, we use the same genetic operators as reported in [35, 36].

5 Results

For comparing φ-approximation approach against Ψ-approximation approach, we simply replace the φ-approximation step in the algorithm flowchart with the Ψ-approximation step. A quadratic approximation is used for both the mappings. Both the ideas were tested on a set of 8 test problems selected from the literature given in Tables 1 and 2. To assess the savings achieved by the two approximation approaches, we compare them against a nested approach where the approximation idea is not incorporated, but the same evolutionary algorithm described in Figure 1 is used at the upper level and a lower level optimization problem is solved for every upper level member. Hereafter, we refer this benchmark as a no-apprximation approach. Whenever lower level optimization is required, we rely on sequential quadratic programming to solve the problem for all cases. Table 3 provides the median function evaluations (31 runs) at the upper and lower level required by each of the three cases, i.e, φ-approximation, Ψ-approximation and no-approximation. Detailed results from multiple runs are presented through Figures 2 and 3. Interestingly, both the approximation ideas

Table 1: Standard test problems TP1-TP5. (Note that $x = x_u$ and $y = x_l$)

Problem	Formulation	Best Known Sol.
TP1 $n = 2$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = (x_1 - 30)^2 + (x_2 - 20)^2 - 20y_1 + 20y_2$, s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (x_1 - y_1)^2 + (x_2 - y_2)^2 \\ 0 \leq y_i \leq 10, \quad i = 1,2 \end{array} \right\}$, $x_1 + 2x_2 \geq 30, x_1 + x_2 \leq 25, x_2 \leq 15$	$F = 225.0$ $f = 100.0$
TP2 $n = 2$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = 2x_1 + 2x_2 - 3y_1 - 3y_2 - 60$, s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (y_1 - x_1 + 20)^2 + (y_2 - x_2 + 20)^2 \\ x_1 - 2y_1 \geq 10, x_2 - 2y_2 \geq 10 \\ -10 \geq y_i \geq 20, \quad i = 1,2 \end{array} \right\}$, $x_1 + x_2 + y_1 - 2y_2 \leq 40$, $0 \leq x_i \leq 50, \quad i = 1,2.$	$F = 0.0$ $f = 100.0$
TP3 $n = 2$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = -(x_1)^2 - 3(x_2)^2 - 4y_1 + (y_2)^2$, s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = 2(x_1)^2 + (y_1)^2 - 5y_2 \\ (x_1)^2 - 2x_1 + (x_2)^2 - 2y_1 + y_2 \geq -3 \\ x_2 + 3y_1 - 4y_2 \geq 4 \\ 0 \leq y_i, \quad i = 1,2 \end{array} \right\}$, $(x_1)^2 + 2x_2 \leq 4$, $0 \leq x_i, \quad i = 1,2$	$F = -18.6787$ $f = -1.0156$
TP4 $n = 2$, $m = 3$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = -8x_1 - 4x_2 + 4y_1 - 40y_2 - 4y_3$, s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = x_1 + 2x_2 + y_1 + y_2 + 2y_3 \\ y_2 + y_3 - y_1 \leq 1 \\ 2x_1 - y_1 + 2y_2 - 0.5y_3 \leq 1 \\ 2x_2 + 2y_1 - y_2 - 0.5y_3 \leq 1 \\ 0 \leq y_i, \quad i = 1,2,3 \end{array} \right\}$, $0 \leq x_i, \quad i = 1,2$	$F = -29.2$ $f = 3.2$
TP5 $n = 2$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = rt(x)x - 3y_1 - 4y_2 + 0.5t(y)y$, s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = 0.5t(y)hy - t(b(x))y \\ -0.333y_1 + y_2 - 2 \leq 0 \\ y_1 - 0.333y_2 - 2 \leq 0 \\ 0 \leq y_i, \quad i = 1,2 \end{array} \right\}$, where $h = \begin{pmatrix} 1 & 3 \\ 3 & 10 \end{pmatrix}, b(x) = \begin{pmatrix} -1 & 2 \\ 3 & -3 \end{pmatrix} x, r = 0.1$ $t(\cdot)$ denotes transpose of a vector	$F = -3.6$ $f = -2.0$

Table 2: Standard test problems TP6-TP8. (Note that $x = x_u$ and $y = x_l$)

Problem	Formulation	Best Known Sol.
TP6 $n = 1$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = (x_1-1)^2 + 2y_1 - 2x_1,$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (2y_1-4)^2 + \\ (2y_2-1)^2 + x_1 y_1 \\ 4x_1 + 5y_1 + 4y_2 \leq 12 \\ 4y_2 - 4x_1 - 5y_1 \leq -4 \\ 4x_1 - 4y_1 + 5y_2 \leq 4 \\ 4y_1 - 4x_1 + 5y_2 \leq 4 \\ 0 \leq y_i, \quad i = 1,2 \end{array} \right\},$ $0 \leq x_1$	$F = -1.2091$ $f = 7.6145$
TP7 $n = 2$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = -\frac{(x_1+y_1)(x_2+y_2)}{1+x_1y_1+x_2y_2},$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = \frac{(x_1+y_1)(x_2+y_2)}{1+x_1y_1+x_2y_2} \\ 0 \leq y_i \leq x_i, \quad i = 1,2 \end{array} \right\},$ $(x_1)^2 + (x_2)^2 \leq 100$ $x_1 - x_2 \leq 0$ $0 \leq x_i, \quad i = 1,2$	$F = -1.96$ $f = 1.96$
TP8 $n = 2$, $m = 2$	$\underset{(x,y)}{\text{Minimize}}\ F(x,y) = \lvert 2x_1 + 2x_2 - 3y_1 - 3y_2 - 60 \rvert,$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (y_1 - x_1 + 20)^2 + \\ (y_2 - x_2 + 20)^2 \\ 2y_1 - x_1 + 10 \leq 0 \\ 2y_2 - x_2 + 10 \leq 0 \\ -10 \leq y_i \leq 20, \quad i = 1,2 \end{array} \right\},$ $x_1 + x_2 + y_1 - 2y_2 \leq 40$ $0 \leq x_i \leq 50, \quad i = 1,2$	$F = 0.0$ $f = 100.0$

Table 3: Median function evaluations for the upper level (UL) and the lower level (LL) from 31 runs of different algorithms. The savings represent the proportion of total function evaluations (LL+UL) saved because of using the approximation when compared with no-approximation approach.

	UL Func. Evals.			LL Func. Evals.			Savings	
	φ-Appx Med	Ψ-Appx Med	No-Appx Med	φ-Appx Med	Ψ-Appx Med	No-Appx Med	φ	Ψ
TP1	134	150	-	1438	2061	-	Large	Large
TP2	148	193	436	1498	2852	5686	73%	50%
TP3	187	137	633	2478	1422	6867	64%	79%
TP4	299	426	1755	3288	6256	19764	83%	69%
TP5	175	270	576	2591	2880	6558	61%	56%
TP6	110	94	144	1489	1155	1984	25%	41%
TP7	166	133	193	2171	1481	2870	24%	47%
TP8	212	343	403	2366	5035	7996	69%	36%

perform significantly well on all the problems as compared to the no-approximation approach. The savings column in the table shows the proportion of function evaluations savings that can be directly attributed to φ and Ψ approximations. There is also a slight difference in performance between the two approximation strategies which could be attributed to the quality of approximation achieved for specific test problems. To provide the readers an idea about the extent of savings in function evaluations obtained from using metamodeling based strategies, we also provide comparisons with earlier evolutionary approaches [45, 46] in Table 4. It is quite clear that the savings achieved are better by order of magnitudes.

It is noteworthy that the Ψ-mapping in a bilevel optimization problem could be a set-valued mapping. In such cases the quadratic approximation idea will not work. To test this hypothesis, we modified all the 8 test problems by adding two additional lower level variables (y_p and y_q) that makes the Ψ-mapping in all the test problems as set-valued for the entire domain of Ψ. The modification does not change the original bilevel solution. This was achieved by modifying the upper and lower level functions for all the test problems as follows:

$$
\begin{aligned}
F^{new}(x, y) &= F(x, y) + y_p^2 + y_q^2 \\
f^{new}(x, y) &= f(x, y) + (y_p - y_q)^2 \\
y_p, y_q &\in [-1, 1]
\end{aligned}
$$

Note that the above modification necessarily makes the lower level problem have multiple optimal solutions corresponding to all x, as the added term gets minimized at $y_p = y_q$ which has infinitely many solutions. Out of the infinitely many lower level optimal solutions, the upper level prefers $y_p = y_q = 0$. With this simple modification, we execute our algorithm with φ-approximation and Ψ-approximation on all test problems, the results for which are presented through Tables 5 and 6. For all the problems, the Ψ-approximation idea fails. The Ψ-approximation idea continues to work effectively as before. The slight increase in function evaluations for the Ψ-approximation approach comes from the fact that there are additional variables in the problem.

Table 4: Mean of the sum of upper level (UL) and lower level (LL) function evaluations for different approaches.

	Mean Func. Evals. (UL+LL)				
	φ-appx.	Ψ-appx.	No-appx.	WJL [45]	WLD [46]
TP1	1595	2381	35896	85499	86067
TP2	1716	3284	5832	256227	171346
TP3	2902	1489	7469	92526	95851
TP4	3773	6806	21745	291817	211937
TP5	2941	3451	7559	77302	69471
TP6	1689	1162	1485	163701	65942
TP7	2126	1597	2389	1074742	944105
TP8	2699	4892	5215	213522	182121

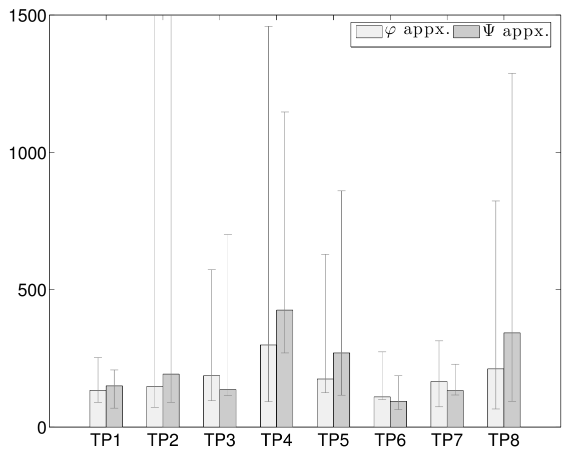

Figure 2: Box plot (31 runs/samples) for the upper level function evaluations required for test problems 1 to 8.

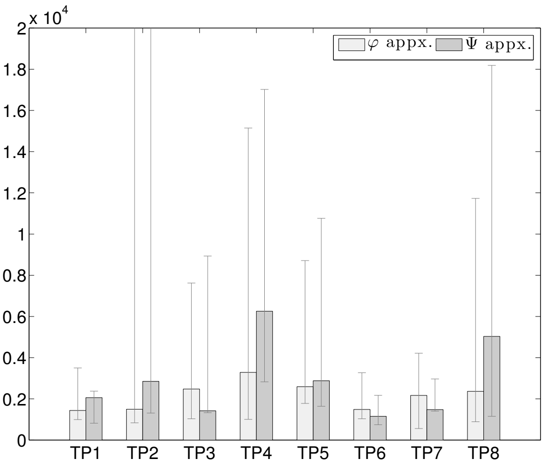

Figure 3: Box plot (31 runs/samples) for the lower level function evaluations required for test problems 1 to 8.

Table 5: Minimum, median and maximum function evaluations at the upper level (UL) from 31 runs of the φ-approximation algorithm on the modifed test problems (m-TP). The other two approaches fail on all the test problems.

	φ-Appx.			Ψ-Appx.	No-Appx.
	Min	Med	Max	Min/Med/Max	Min/Med/Max
m-TP1	130	172	338	-	-
m-TP2	116	217	-	-	-
m-TP3	129	233	787	-	-
m-TP4	198	564	2831	-	-
m-TP5	160	218	953	-	-
m-TP6	167	174	529	-	-
m-TP7	114	214	473	-	-
m-TP8	150	466	2459	-	-

Table 6: Minimum, median and maximum function evaluations at the lower level (LL) from 31 runs of the φ-approximation algorithm on the modifed test problems (m-TP). The other two approaches fail on all the test problems.

	φ-Appx.			Ψ-Appx.	No-Appx.
	Min	Med	Max	Min/Med/Max	Min/Med/Max
m-TP1	2096	2680	8629	-	-
m-TP2	2574	4360	-	-	-
m-TP3	1394	3280	13031	-	-
m-TP4	1978	5792	28687	-	-
m-TP5	3206	4360	17407	-	-
m-TP6	2617	3520	8698	-	-
m-TP7	1514	5590	11811	-	-
m-TP8	2521	6240	35993	-	-

6 Conclusions

In this paper, we have evaluated two single-level reduction techniques for bilevel optimization; i.e., Ψ-mapping approximation approach and φ-mapping approximation approach. To conclude, the Ψ-mapping offers the advantage that if it can be approximated accurately, it readily gives the optimal lower level variables. However, in cases when this mapping is set-valued, approximating Ψ can be very difficult. On the other hand, the φ-mapping is always single-valued, approximating which is much easier, and is therefore more preferred over the Ψ-mapping. The results shown in this paper clearly demonstrate that even a simple modification that makes the lower level problem have multiple-optimal solutions, makes the Ψ-approximation strategy fail because of poor quality of approximation. To our best knowledge, most of the studies utilizing meta-modelling techniques to solve bilevel optimization problems have mostly relied on approximating the Ψ-mapping. Given the ease and reliability offered by the φ-approximation over Ψ-approximation, we believe that the meta-modelling-based techniques should closely look at the benefits of the φ-approximation.

References

[1] F. Al-Khayyal, R. Horst, and P. Pardalos. Global optimization of concave functions subject to quadratic constraints: an application in nonlinear bilevel programming. *Annals of Operations Research*, 34:125–147, 1992.

[2] Sebastian Albrecht, K Ramirez-Amaro, Federico Ruiz-Ugalde, David Weikersdorfer, M Leibold, Michael Ulbrich, and Michael Beetz. Imitating human reaching motions using physically inspired optimization principles. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 602–607. IEEE, 2011.

[3] B. An, F. Ordóñez, M. Tambe, E. Shieh, R. Yang, C. Baldwin, J. DiRenzo III, K. Moretti, B. Maule, and G. Meyer. A Deployed Quantal Response-Based Patrol Planning System for the U.S. Coast Guard. *Interfaces*, 43(5):400–420, 2013.

[4] J. Angelo, E. Krempser, and H. Barbosa. Differential evolution for bilevel programming. In *Proceedings of the 2013 Congress on Evolutionary Computation (CEC-2013)*. IEEE Press, 2013.

[5] Jaqueline S Angelo and Helio JC Barbosa. A study on the use of heuristics to solve a bilevel programming problem. *International Transactions in Operational Research*, 2015.

[6] T. Bäck. *Evolutionary Algorithms in Theory and Practice*. New York: Oxford University Press, 1996.

[7] J. Bard and J. Falk. An explicit solution to the multi-level programming problem. *Computers and Operations Research*, 9:77–100, 1982.

[8] J. Bard and J. Moore. A branch and bound algorithm for the bilevel programming problem. *SIAM Journal on Scientific and Statistical Computing*, 11:281–292, 1990.

[9] J. F. Bard. Coordination of multi-divisional firm through two levels of management. *Omega*, 11(5):457–465, 1983.

[10] O. Ben-Ayed. Bilevel linear programming. *Computers and Operations Research*, 20:485–501, 1993.

[11] M. P. Bendsoe. Optimization of structural topology, shape, and material. Technical report, 1995.

[12] W. Bialas and M. Karwan. Two-level linear programming. *Management Science*, 30:1004–1020, 1984.

[13] J. Bracken and J. McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21:37–44, 1973.

[14] Luce Brotcorne, Martine Labbe, Patrice Marcotte, and Gilles Savard. A bilevel model for toll optimization on a multicommodity transportation network. *Transportation Science*, 35(4):345–358, 2001.

[15] G. Brown, M. Carlyle, D. Diehl, J. Kline, and K. Wood. A Two-Sided Optimization for Theater Ballistic Missile Defense. *Operations Research*, 53(5):745–763, 2005.

[16] Y. Chen and M. Florian. On the geometry structure of linear bilevel programs: a dual approach. Technical Report CRT-867, Centre de Recherche sur les Transports, 1992.

[17] Snorre Christiansen, Michael Patriksson, and Laura Wynter. Stochastic bilevel programming in structural optimization. Technical report, Structural and Multidisciplinary Optimization, 1997.

[18] Peter A Clark and Arthur W Westerberg. Bilevel programming for steady-state chemical process designÑi. fundamentals and algorithms. *Computers & Chemical Engineering*, 14(1):87–97, 1990.

[19] Isabelle Constantin and Michael Florian. Optimizing frequencies in a transit network: a nonlinear bi-level programming approach. *International Transactions in Operational Research*, 2(2):149 – 164, 1995.

[20] T. Edmunds and J. Bard. Algorithms for nonlinear bilevel mathematical programming. *IEEE Transactions on Systems, Man, and Cybernetics*, 21:83–89, 1991.

[21] J. Fortuny-Amat and B. McCarl. A representation and economic interpretation of a two-level programming problem. *Journal of the Operational Research Society*, 32:783–792, 1981.

[22] S Reza Hejazi, Azizollah Memariani, G Jahanshahloo, and Mohammad Mehdi Sepehri. Linear bilevel programming solution by genetic algorithm. *Computers & Operations Research*, 29(13):1913–1925, 2002.

[23] X. Hu and D. Ralph. Using EPECs to Model Bilevel Games in Restructured Electricity Markets with Locational Prices. *Operations Research*, 55(5):809–827, 2007.

[24] Yan Jiang, Xuyong Li, Chongchao Huang, and Xianing Wu. Application of particle swarm optimization based on chks smoothing function for solving nonlinear bilevel programming problem. *Applied Mathematics and Computation*, 219(9):4332–4339, 2013.

[25] QIN Jin and SHI Feng. Bi-level simulated annealing algorithm for facility location. *Systems Engineering*, 2:007, 2007.

[26] M. Labbé, P. Marcotte, and G. Savard. A Bilevel Model of Taxation and Its Application to Optimal Highway Pricing. *Management Science*, 44(12):1608–1622, 1998.

[27] Hecheng Li. A genetic algorithm using a finite search space for solving nonlinear/linear fractional bilevel programming problems. *Annals of Operations Research*, pages 1–16, 2015.

[28] Hecheng Li and Yuping Wang. A hybrid genetic algorithm for solving nonlinear bilevel programming problems based on the simplex method. *International Conference on Natural Computation*, 4:91–95, 2007.

[29] Xiangyong Li, Peng Tian, and Xiaoping Min. A hierarchical particle swarm optimization for solving bilevel programming problems. In Leszek Rutkowski, Ryszard Tadeusiewicz, Lotfi A. Zadeh, and Jacek M. Zurada, editors, *Artificial Intelligence and Soft Computing âĂŞ ICAISC 2006*, volume 4029 of *Lecture Notes in Computer Science*, pages 1169–1178. Springer Berlin Heidelberg, 2006.

[30] Guoshan Liu, Jiye Han, and Shouyang Wang. A trust region algorithm for bilevel programing problems. *Chinese science bulletin*, 43(10):820–824, 1998.

[31] R. Mathieu, L. Pittard, and G. Anandalingam. Genetic algorithm based approach to bi-level linear programming. *Operations Research*, 28(1):1–21, 1994.

[32] Athanasios Migdalas. Bilevel programming in traffic planning: Models, methods and challenge. *Journal of Global Optimization*, 7(4):381–405, 1995.

[33] Katja Mombaur, Anh Truong, and Jean-Paul Laumond. From human to humanoid locomotionÑan inverse optimal control approach. *Autonomous robots*, 28(3):369–383, 2010.

[34] M.G. Nicholls. Aluminium Production Modeling - A Nonlinear Bilevel Programming Approach. *Operations Research*, 43(2):208–218, 1995.

[35] A. Sinha, P. Malo, and K. Deb. Efficient evolutionary algorithm for single-objective bilevel optimization. *arXiv preprint arXiv:1303.3901*, 2013.

[36] A. Sinha, P. Malo, and K. Deb. An improved bilevel evolutionary algorithm based on quadratic approximations. In *2014 IEEE Congress on Evolutionary Computation (CEC-2014)*, pages 1870–1877. IEEE Press, 2014.

[37] A. Sinha, P. Malo, and K. Deb. Transportation policy formulation as a multi-objective bilevel optimization problem. In *2015 IEEE Congress on Evolutionary Computation (CEC-2015)*. IEEE Press, 2015.

[38] A. Sinha, P. Malo, A. Frantsev, and K. 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 (CEC-2013)*. IEEE Press, 2013.

[39] W.R. Smith and R.W. Missen. *Chemical Reaction Equilibrium Analysis: Theory and Algorithms.* John Wiley & Sons, New York, 1982.

[40] Heinrich von Stackelberg. *The theory of the market economy.* Oxford University Press, New York, 1952.

[41] Huijun Sun, Ziyou Gao, and Jianjun Wu. A bi-level programming model and solution algorithm for the location of logistics distribution centers. *Applied Mathematical Modelling*, 32(4):610 – 616, 2008.

[42] H. Tuy, A. Migdalas, and P. Värbrand. A global optimization approach for the linear two-level program. *Journal of Global Optimization*, 3:1–23, 1993.

[43] Takeshi Uno, Hideki Katagiri, and Kosuke Kato. An evolutionary multi-agent based search method for stackelberg solutions of bilevel facility location problems. *International Journal of Innovative Computing, Information and Control*, 4(5):1033–1042, 2008.

[44] Zhongping Wan, Guangmin Wang, and Bin Sun. A hybrid intelligent algorithm by combining particle swarm optimization with chaos searching technique for solving nonlinear bilevel programming problems. *Swarm and Evolutionary Computation*, 8:26–32, 2013.

[45] Y. Wang, Y. C. Jiao, and H. Li. An evolutionary algorithm for solving nonlinear bilevel programming based on a new constraint-handling scheme. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 32(2):221–232, 2005.

[46] Yuping Wang, Hong Li, and Chuangyin Dang. A new evolutionary algorithm for a class of nonlinear bilevel programming problems and its global convergence. *INFORMS Journal on Computing*, 23(4):618–629, 2011.

[47] L. Wein. Homeland Security: From Mathematical Models to Policy Implementation: The 2008 Philip McCord Morse Lecture. *Operations Research*, 57(4):801–811, 2009.

[48] U. Wen and S. Hsu. Linear bi-level programming problems - a review. *Journal of the Operational Research Society*, 42:125–133, 1991.

[49] N. Williams, P.K. Kannan, and S. Azarm. Retail Channel Structure Impact on Strategic Engineering Product Design. *Management Science*, 57(5):897–914, 2011.

[50] Jane J Ye and Daoli Zhu. New necessary optimality conditions for bilevel programs by combining the mpec and value function approaches. *SIAM Journal on Optimization*, 20(4):1885–1905, 2010.

[51] Y. Yin. Genetic algorithm based approach for bilevel programming models. *Journal of Transportation Engineering*, 126(2):115–120, 2000.

[52] Xiaobo Zhu, Qian Yu, and Xianjia Wang. A hybrid differential evolution algorithm for solving nonlinear bilevel programming with linear constraints. In *Cognitive Informatics, 2006. ICCI 2006. 5th IEEE International Conference on*, volume 1, pages 126–131. IEEE, 2006.