2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

# An Improved Bilevel Evolutionary Algorithm based on Quadratic Approximations

Ankur Sinha, Pekka Malo, and Kalyanmoy Deb*
Department of Information and Service Economy
Aalto University School of Business
PO Box 21220, 00076 Aalto, Helsinki, Finland
{Firstname.Lastname}@aalto.fi

***Abstract*— In this paper, we provide an improved evolutionary algorithm for bilevel optimization. It is an extension of a recently proposed Bilevel Evolutionary Algorithm based on Quadratic Approximations (BLEAQ). Bilevel optimization problems are known to be difficult and computationally demanding. The recently proposed BLEAQ approach has been able to bring down the computational expense significantly as compared to the contemporary approaches. The strategy proposed in this paper further improves the algorithm by incorporating archiving and local search. Archiving is used to store the feasible members produced during the course of the algorithm that provide a larger pool of members for better quadratic approximations of optimal lower level solutions. Frequent local searches at upper level supported by the quadratic approximations help in faster convergence of the algorithm. The improved results have been demonstrated on two different sets of test problems, and comparison results against the contemporary approaches are also provided.**

***Index Terms*— Bilevel optimization, evolutionary algorithms, quadratic approximations, quadratic programming, local search.**

## I. Introduction

Bilevel optimization is a complex optimization problem with two levels of optimization tasks. The two optimization tasks are commonly referred to as upper and lower level tasks with the lower level nested within the upper level. The lower level optimization task is a constraint to the upper level optimization task such that a solution can be considered feasible at the upper level only if it is an optimal solution at the lower level and also satisfies the upper level equality and inequality constraints. This requirement makes bilevel problems difficult to solve. Along with two levels of optimization, a bilevel problem also contains two kinds of variables corresponding to each level. The variables are also commonly referred to as upper and lower level variables. A number of classical and evolutionary approaches have been proposed to solve bilevel optimization problems. However, on one hand most of the classical approaches are too restrictive that they are applicable only to a small class of problems, and on the other hand the evolutionary approaches are computationally expensive that they do not scale for problems with larger number of variables. Therefore, there is a need for efficient bilevel procedures that can handle complex bilevel problems efficiently.

Recently, bilevel evolutionary algorithm based on quadratic approximations (BLEAQ) has been proposed [42] by the authors that has been shown to efficiently solve a variety of bilevel optimization problems. In this paper, we further improve upon the approach by incorporating archiving and local search ability into the algorithm. Since the BLEAQ approach is based on quadratic approximations of the optimal lower level variables corresponding to upper level variables, the idea of archiving supports the quadratic approximations by providing a larger pool of members for approximations, and the idea of local search helps in utilizing the quadratic approximations for faster convergence. Integration of archiving and local search into BLEAQ improves the performance of the algorithm substantially on all the test problems chosen in the paper.

The sections in the paper are organized as follows. In the next section, we provide a generic formulation for bilevel optimization problems that is followed by the relevance of these problems in practice. Thereafter, we provide a brief literature review on bilevel optimization using evolutionary algorithms. This is followed by a detailed description of the improved methodology where we incorporate the archiving and local search idea into BLEAQ. After this we evaluate the improved algorithm on a number of test problems. Firstly, the improved method is evaluated on a set of standard test problems chosen from the literature [39], [2], [3], [9], [36], [35], [31], [53]. For these test problems, we provide a comparison against the algorithms proposed in [52], [53] and the original BLEAQ algorithm. Secondly, we evaluate the improved method on the SMD test-suite [41], [43] and provide a performance comparison against the original BLEAQ algorithm and a nested bilevel evolutionary algorithm (NBLEA) [45], [43].

## II. A Brief Description of Bilevel Optimization

Bilevel optimization involves an optimization task within the constraints of another optimization task. Such problems contain two kinds of variables: the upper level variables $x_u$, and the lower level variables $x_l$. At the upper level the optimization is expected to be performed with respect to $x_u$ as well as $x_l$ in the presence of constraints that includes the nested optimization task. Lower level optimization is

*Also Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA (Kdeb@egr.msu.edu).

performed with respect to the lower level variables $x_l$, while the upper level variables $x_u$ act as parameters. We provide two equivalent formulations for a general bilevel optimization problem below:

*Definition 1 (Bilevel Optimization Problem):* Let $X = X_U \times X_L$ denote the product of the upper-level decision space $X_U$ and the lower-level decision space $X_L$, i.e. $x = (x_u, x_l) \in X$, if $x_u \in X_U$ and $x_l \in X_L$. For upper-level objective function $F : X \to \mathbb{R}$ and lower-level objective function $f : X \to \mathbb{R}$, a general bilevel optimization problem is given by

$$\begin{aligned} \underset{x \in X}{\text{Min}} \quad & F(x), \\ \text{s.t.} \quad & x_l \in \underset{x_l \in X_L}{\text{argmin}} \left\{ f(x) \;\middle|\; g_i(x) \geq 0, i \in I \right\}, \\ & G_j(x) \geq 0, j \in J. \end{aligned} \tag{1}$$

where the functions $g_i : X \to \mathbb{R}$, $i \in I$, represent lower-level constraints and $G_j : X \to \mathbb{R}$, $j \in J$, represent upper-level constraints.

The structure of the problem introduces a difficulty that only the optimal solutions to the lower level optimization task may be acceptable as possible feasible candidates to the upper level optimization task. For instance, a member $x^{(0)} = (x_u^{(0)}, x_l^{(0)})$ can be considered feasible at the upper level only if $x^{(0)}$ satisfies the upper level constraints, and $x_l^{(0)}$ is an optimal solution to the lower level problem corresponding to $x_u^{(0)}$. Next, we provide an equivalent formulation of the bilevel optimization problem by replacing the lower-level optimization problem with a set valued function that maps a given upper level vector to corresponding optimal lower-level vector(s).

*Definition 2 (Alternative formulation):* Let the set-valued function $\Psi : X_U \rightrightarrows X_L$ denote the optimal-solution mapping of the lower level problem, i.e.

$$\Psi(x_u) = \underset{x_l \in X_L}{\text{argmin}} \left\{ f(x) \;\middle|\; g_i(x) \geq 0, i \in I \right\}.$$

A general bilevel optimization problem is then given by

$$\begin{aligned} \underset{x \in X}{\text{Min}} \quad & F(x), \\ \text{s.t.} \quad & x_l \in \Psi(x_u), \\ & G_j(x) \geq 0, j \in J. \end{aligned} \tag{2}$$

where the function $\Psi$ may be a single-vector valued or a multi-vector valued function depending on whether the lower level function has multiple global optimal solutions or not.

*Example 3:* The above formulations have been explained below with the help of a simple quadratic unconstrained bilevel problem with a single upper level and a single lower level variable.

$$\begin{aligned} \underset{(x_u, x_l)}{\text{Min}} \quad & F(x_u, x_l) = (x_u - a)^2 + (x_l - b)^2, \\ \text{s.t.} \quad & x_l \in \underset{x_l}{\text{argmin}} \left\{ f(x_u, x_l) = (x_u - x_l)^2 - x_u^2 \right\}, \\ & -10(a+b) \geq x_u, x_l \geq 10(a+b). \end{aligned} \tag{3}$$

where $a$ and $b$ are constants. In the example, the lower level problem is a parameterized quadratic optimization problem. Figure 1 shows different lower level optimization problems corresponding to a few upper level points. The lower level optimal solutions corresponding to those upper level points are shown by small circles that are feasible solutions to the upper level problem. The path traced by the optimal solutions is shown by a broken curve that represents all the feasible solutions to the upper level problem. A solution lying on this broken line has a property such that the value of optimal $x_l$ is equal to $x_u$. This gets further clarified when one looks at the $x_l$ vs $x_u$ plot in Figure 2. In the figure the straight line represents the $\Psi$-mapping for the problem which is single-valued in this case. The contours for the upper level objective function are circular and the bilevel minimum corresponds to the contour that is tangent to the straight line. The bilevel optimum is given by $(\frac{a+b}{2}, \frac{a+b}{2})$.

## III. Practical Applications of Bilevel Optimization

Bilevel optimization commonly appears in many practical problems. They are often encountered in the fields of economics [23], [50], [45], [44] in the context of Stackelberg games, principal-agency problems and policy decisions. In the domain of transportation [34], [18], [13] they commonly arise in network design and toll-setting problems. In management [48], [8] bilevel optimization appears in facility location problems and hierarchical decision making and optimization within firms. The field of engineering [26], [46] also involves a number of bilevel scenarios for example in optimal structure design, optimal chemical equilibria etc. For additional bilevel optimization problems found in practice researchers may refer to [21], [6]. Next, we discuss a few examples from practice to give a further insight to the readers on the relevance of these problems.

1) Toll Setting Problem: In this problem, there is an authority that wants to optimize the tolls for a network of roads. The authority acts as an upper level decision maker and the network users act as lower level decision makers. For any given toll the network users optimize their own problem of cost and time minimization. The structure of the problem is such that the authority can optimize its revenues earned by tolls only by taking the network users' problem into account [13].
2) Stackelberg Games: A Stackelberg game is a strategic competition involving a leader and a follower. In this model, the leader makes the first move and has all relevant information about the possible actions the follower might take in response to the leader's actions. The follower is expected to react optimally to the actions of the leader. In order to determine its optimal actions, the leader has to take the possible follower actions into account introducing a hierarchy into the problem [23], [47].
3) Environmental Economics: Bilevel optimization commonly appears in environmental economics. For exam-

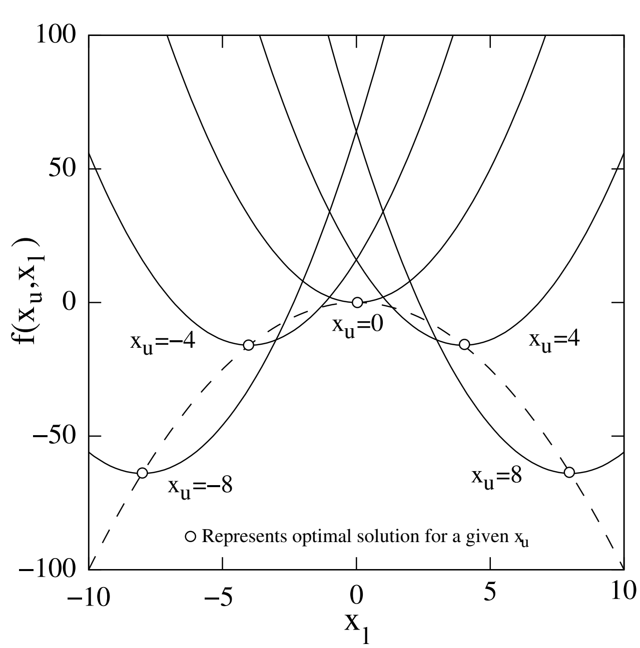


Fig. 1. Lower level optimization problems corresponding to different values of $x_u$.

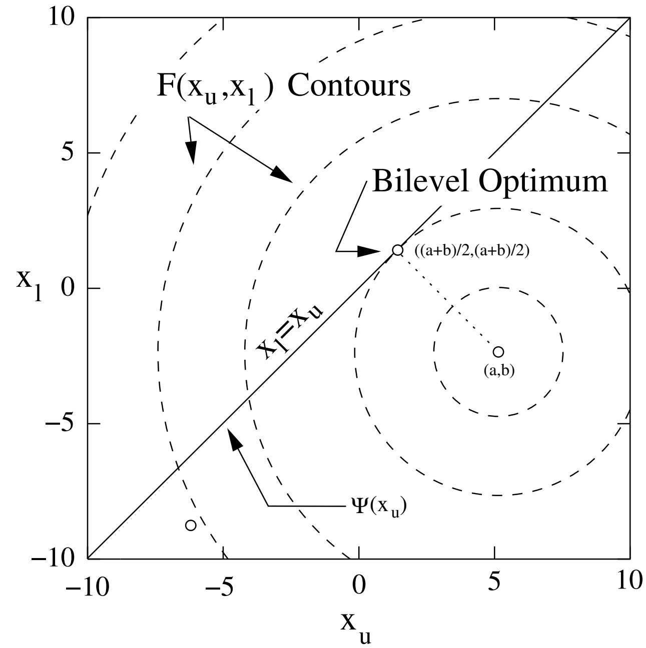


Fig. 2. Upper level objective contours with respect to $x_u$ and $x_l$ along with the $\Psi$-mapping.

ple, consider a mining activity [44] that leads to pollution as well as revenue generation. The mining activity is performed by a mining company and regulated by an authority. The regulating authority acts as a leader and the mining company as a follower. The regulating authority usually has two objectives of maximizing its revenues by taxation and minimizing the environmental damage. The mining company on the other hand may have a sole objective of maximizing profits. Under this situation, the authority solves a bilevel optimization problem with two objectives at the upper level and a single objective at the lower level in order to determine an optimal taxation policy.

4) Chemical Industry: In chemical industries, the chemists often face a bilevel optimization problem when they have to decide upon the quantities of reactants in order to optimally create one or more products. In a chemical reaction products are produced when an equilibrium is established between the reactants and products. Therefore, in order to optimize the quantity of products, one needs to ensure equilibrium which itself is an optimization task. In this problem, the upper level objective is to maximize the quantity of a particular product, while the lower level problem involves optimizing the entropy functional to ensure that an equilibria is established [46].
5) Structural optimization: Structural optimization problems also involve a bilevel optimization task [10], [16]. In structural optimization problems minimization of the weight or cost of a structure appears as an upper level objective. One needs to search for the most suitable design variables in order to achieve the objective. The upper level decision variables usually are shape of the structure, choice of materials, amount of material etc. The constraints at the upper level involve bounds on displacements, stresses and contact forces. The displacements, stresses and contact forces are lower level variables whose values are determined by minimizing the potential energy of the system that appears as a lower level optimization task.
6) Defense applications: Bilevel optimization has a number of applications in the defense sector, for example attacker-defender Stackelberg game [4], planning the pre-positioning of defensive missile interceptors to counter an attack threat [14], interdicting nuclear weapons project [15], and homeland security applications [54], [32]. Other applications include strategic bomber force structure, and allocation of tactical aircraft to missions [20]. The bilevel problem in such cases may involve maximizing the damage caused to the opponent by taking into account the optimal reactions of the opponent. Conversely, minimizing the maximum damage that an attacker can cause is also a bilevel optimization task.
7) Others: Bilevel optimization problems are also realized in principal-agent problems, optimal pricing, network facility location, optimal algorithm configuration, machine learning etc.

## IV. A Review of Bilevel Algorithms

In this section, we highlight some of the studies in classical and evolutionary optimization literature on bilevel optimization. Bilevel programming was introduced in the domain of mathematical programming by Bracken and Mcgill [12] in the early seventies. Since then a number of studies have been conducted on bilevel programming [17], [49], [22]. A number of solution methodologies have also been developed by researchers in classical optimization with simplifying assumptions like smoothness, linearity or convexity. These assumptions limit the application of the methodologies only to a smaller class of bilevel problems. Researchers have attempted to solve bilevel programming problems using Karush-Kuhn-Tucker approach [11], [25], branch-and-bound techniques [7], and the use of penalty functions [1]. However, one often needs to resort to other approaches like evolutionary techniques when bilevel problems get complex. A number of evolutionary algorithms for bilevel optimization have also been proposed with a number of them being nested strategies [33], [55], [29], [45], [5] that solve the lower level optimization problem completely for any given upper level decision. In [33], [55], [29] the authors handle the upper level optimization task using an evolutionary algorithm and the lower level is handled using a classical approach. In [45], [5] the authors handle both levels using an evolutionary technique. Researchers in the evolutionary community have also attempted to replace the lower level optimization task with the Karush-Kuhn-Tucker conditions [51], [28], [30] to convert the bilevel optimization into a single level constrained optimization task. Co-evolutionary approaches to handle bilevel optimization problems have been proposed in [35], [27]. Wang et al. [52] developed an evolutionary algorithm based on a constraint handling scheme, where they successfully solve a number of standard test problems. Later on, they provided an improved algorithm [53] that was able to handle non-differentiable upper level objective function and non-convex lower level problem. There has also been an interest in multi-objective bilevel optimization using evolutionary algorithms. Some of the studies in the direction of solving multi-objective bilevel optimization problems are [24], [38], [19], [40], [37], [56].

## V. Algorithm Description

Recently, a bilevel evolutionary algorithm based on quadratic approximations (BLEAQ) [42] has been proposed that is shown to perform better than some of the recently proposed techniques. In this section, we further improve upon that method by incorporating archiving and local search in the BLEAQ approach. For brevity, we focus on the algorithm description at the upper level, where archiving and local search is incorporated. Other parts of the algorithm like lower level optimization, constraint handling, crossover, mutation and termination are kept the same as in the original algorithm. We recommend the readers to refer to [42] for further details.

S. 1 Initialize a random population of upper level variables of size $N$. Execute the lower level optimization problem to identify optimal lower level variables. Assign fitness based on upper level function value and constraints. Initialize generation number as $gen \leftarrow 0$.

S. 2 Tag all upper level members that have undergone a successful lower level optimization run as 1, and others as 0. Copy the tag 1 members to an archive.

S. 3 Increment generation number as $gen \leftarrow gen + 1$. Choose the best tag 1 member as one of the parents (index parent) from the population[1]. Randomly choose $2(\mu - 1)$ members from the population and perform a tournament selection based on upper level fitness to choose remaining $\mu - 1$ parents.

S. 4 Create $\lambda$ offspring from the chosen $\mu$ parents, using crossover and polynomial mutation operators.

S. 5 If the number of tag 1 members in the population is greater than $\frac{N}{2}$ and archive size is greater than $\frac{(dim(x_u)+1)(dim(x_u)+2)}{2} + dim(x_u)$, then select $\frac{(dim(x_u)+1)(dim(x_u)+2)}{2} + dim(x_u)$ closest archive members[2] from the index parent. Construct quadratic functions $Q_t, t \in \{1, \ldots, dim(x_l)\}$ to represent lower level optimal variables as a function of upper level variables.

S. 6 If a quadratic approximation was performed in the previous step, find the lower level optimum for the offspring using the quadratic functions ($Q_t$). If the mean squared error $e_{mse}$ is less than $e_0$(1e-3), the quadratic approximation is considered good and the offspring are tagged as 1, otherwise they are tagged as 0. If a quadratic approximation was not performed in the previous step, execute lower level optimization runs for each offspring. Tag the offspring as 1 for which a successful lower level optimization is performed.

S. 7 Copy the tag 1 offspring from the previous step (if any) to the archive. After finding the lower level variables for the offspring, choose $r$ members from the parent population. A pool of chosen $r$ members and $\lambda$ offspring is formed. The best $r$ members from the pool replace the chosen $r$ members from the population.

S. 8 If $gen$ is divisible by $g_l$ and quadratic functions were generated at Step 5, then reduce the bilevel problem to a single level problem, such that optimal lower level solutions are given by the quadratic functions $Q_t$. Solve the single level optimization problem using local search (Refer to Subsection V-A) around the best population member.

S. 9 If a local search is performed, test the upper level solution produced from the previous step by performing a lower level optimization at that point and evaluating the upper level fitness. If the newly produced point is better than the population best then replace the population best by the newly

[1]The choice of best tag 1 member as a parent makes the algorithm faster. However, for better exploration at upper level some other strategy may also be used.

[2]Please note that a quadratic fit in $d$ dimensions requires at least $\frac{(d+1)(d+2)}{2}$ points. However, to avoid overfitting we use at least $\frac{(d+1)(d+2)}{2} + d$ points.

generated point.

S. 10 Perform a termination check. If the termination check is false, the algorithm moves to the next generation (Step 3).

### A. Local Search

To perform local search, we reduce the bilevel optimization problem to a single level optimization problem using the quadratic functions $Q_t$ that approximate the lower level optimal solution for any given upper level vector. The auxiliary problem can be written as follows,

$$\begin{aligned} \underset{x \in X}{\text{Min}} \quad & F(x_u, x_l), \\ \text{s.t.} \quad & x_{l,t} = Q_t(x_u), \forall\, t \in \{1, \ldots, dim(x_l)\} \\ & G_j(x_u, x_l) \geq 0, j \in J. \end{aligned} \tag{4}$$

where $Q_t$ is a quadratic function of the $x_u$ variables. The above single level problem can be solved using sequential quadratic programming, if the functions are differentiable. The best upper level member in the population is used as a starting solution. If the functions $F(x_u, x_l)$ and $G_j(x_u, x_l)$ are non-differentiable, we approximate them with quadratic functions by sampling points around the best member in the population and then use sequential quadratic programming. Please note that $Q_t$ represents a single-valued function between $x_u$ and optimal $x_l$. It is not necessary that the mapping between $x_u$ and optimal $x_l$ is single-valued, rather there can be multiple optimal $x_l$ for a given $x_u$. Therefore, the auxiliary problem should not be considered as a local approximation of the bilevel problem. The benefit in solving such a single level problem is that it is able to direct the BLEAQ approach into better regions in the search space.

### B. Parameter Setting

The parameters in the algorithm are fixed as $\mu = 3$, $\lambda = 2$ and $r = 2$ at both levels. Parameter $g_l$ that determines the frequency of local search is fixed as 50, such that a local search is performed after every 50 generations. The probability for crossover is fixed at $0.9$ and the probability for mutation is fixed at $0.1$. The population sizes at both levels are fixed at 50 for all the problems. The termination strategy and the termination parameters are kept the same as proposed in [42].

## VI. Results

In this section, we present the results obtained from the modified version of the BLEAQ approach. A comparative study has been performed against the previous version of the BLEAQ approach, nested bilevel evolutionary algorithm (NBLEA) [41], [45], and two other approaches [52], [53]. We choose two sets of test problems for comparison. The first set contains 10 standard bilevel test problems (referred to as TP [42]) collected from the literature [39], [2], [3], [9], [36], [35], [31], [53]. The second set consists of 6 recently proposed scalable unconstrained bilevel problems (referred to as SMD [41]). These test problems evaluate the performance of the approaches on different kinds of complexities that may arise in realistic bilevel problems. Next, we provide the results on the two sets in separate subsections.

### A. Standard test problems

Table I defines the set of 10 standard test problems. The table contains the dimensions of the upper and lower level variables in the first column, the problem formulation in the second column, and the best known solution in the third column. We evaluate five different approaches on this test set. The first approach is the modified BLEAQ approach, the second approach is the original BLEAQ approach, the third (WJL) and fourth (WLD) approaches are the ones proposed by Wang et al. [52], [53] in 2005 and 2011 respectively, and the fifth approach is a nested strategy (NBLEA). Table II shows the best, mean and worst function evaluations at the two levels required by the modified BLEAQ approach. The mean accuracy at the upper and the lower level is also reported. Table III provides a comparison between modified BLEAQ and other approaches. In the table, the mean function evaluations (MFE) represents the average of the sum of upper and lower level function evaluations required by an approach on a particular test problem. Mean function evaluations ratio (MFER) represents the ratio of MFE for two different approaches. Original BLEAQ, WJL, WLD and NBLEA approaches have been compared against the modified BLEAQ algorithm in terms of MFER. It can be observed that the modified BLEAQ approach performs consistently better than original BLEAQ approach and significantly better than WJL, WLD and NBLEA.

### B. SMD test problems

For brevity, we do not present the SMD test problems in this paper rather refer the readers to [41] for the description of the test problems. We evaluate the performance of the modified BLEAQ approach, original BLEAQ approach and the nested bilevel evolutionary algorithm approach on this test set. The results of the study are presented in Table IV that provides the mean lower and upper level function evaluations along with the mean accuracy at the two levels. The MFER has been reported for original BLEAQ and NBLEA when compared against modified BLEAQ. For SMD problems again we observe a consistently better performance of the modified BLEAQ idea over the other two strategies.

## VII. Conclusions

The original BLEAQ approach [42] has been shown to outperform a number of contemporary strategies for bilevel optimization. In this paper, we further improve upon the original idea by incorporating archiving and local search in the methodology. It has been shown how the idea of quadratic approximations can be used to reduce the bilevel problem to a single level optimization task for local search at the upper level. Though the auxiliary problem may not be an excellent approximation of the bilevel problem, it is usually sufficiently informative to direct the search of an algorithm in better regions. Frequent local searches based on this auxiliary problem lead to a faster convergence. The archive maintains a large pool of members for performing quadratic estimations with higher accuracy. Incorporation

TABLE I

DESCRIPTION OF THE SELECTED STANDARD TEST PROBLEMS (TP1-TP10) WITH $x$ AS UPPER LEVEL VECTOR AND $y$ AS LOWER LEVEL VECTOR.

| 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. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (x_1 - y_1)^2 + (x_2 - y_2)^2 \\ 0 \le y_i \le 10, \quad i = 1,2 \end{array} \right\},$ $x_1 + 2x_2 \ge 30,\ x_1 + x_2 \le 25,\ x_2 \le 15$ | $F = 225.0$ $f = 100.0$ |
| TP2 $n = 2$ $m = 2$ | $\underset{(x,y)}{\text{Minimize}}\ F(x,y) = 2x_1 + 2x_2 - 3y_1 - 3y_2 - 60,$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (y_1 - x_1 + 20)^2 + (y_2 - x_2 + 20)^2 \\ x_1 - 2y_1 \ge 10,\ x_2 - 2y_2 \ge 10 \\ -10 \ge y_i \ge 20, \quad i = 1,2 \end{array} \right\},$ $x_1 + x_2 + y_1 - 2y_2 \le 40,$ $0 \le x_i \le 50, \quad i = 1,2.$ | $F = 0.0$ $f = 100.0$ |
| 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. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = 2(x_1)^2 + (y_1)^2 - 5y_2 \\ (x_1)^2 - 2x_1 + (x_2)^2 - 2y_1 + y_2 \ge -3 \\ x_2 + 3y_1 - 4y_2 \ge 4 \\ 0 \le y_i, \quad i = 1,2 \end{array} \right\},$ $(x_1)^2 + 2x_2 \le 4,$ $0 \le x_i, \quad i = 1,2$ | $F = -18.6787$ $f = -1.0156$ |
| 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. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = x_1 + 2x_2 + y_1 + y_2 + 2y_3 \\ y_2 + y_3 - y_1 \le 1 \\ 2x_1 - y_1 + 2y_2 - 0.5y_3 \le 1 \\ 2x_2 + 2y_1 - y_2 - 0.5y_3 \le 1 \\ 0 \le y_i, \quad i = 1,2,3 \end{array} \right\},$ $0 \le x_i, \quad i = 1,2$ | $F = -29.2$ $f = 3.2$ |
| 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. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = 0.5t(y)hy - t(b(x))y - 0.333y_1 + y_2 - 2 \le 0 \\ y_1 - 0.333y_2 - 2 \le 0 \\ 0 \le y_i, \quad i = 1,2 \end{array} \right\},$ 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 | $F = -3.6$ $f = -2.0$ |
| TP6 $n = 1$ $m = 2$ | $\underset{(x,y)}{\text{Minimize}}\ F(x,y) = (x_1 - 1)^2 + 2y_1 - 2x_1,$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (2y_1 - 4)^2 + (2y_2 - 1)^2 + x_1 y_1 \\ 4x_1 + 5y_1 + 4y_2 \le 12 \\ 4y_2 - 4x_1 - 5y_1 \le -4 \\ 4x_1 - 4y_1 + 5y_2 \le 4 \\ 4y_1 - 4x_1 + 5y_2 \le 4 \\ 0 \le y_i, \quad i = 1,2 \end{array} \right\},$ $0 \le x_1$ | $F = -1.2091$ $f = 7.6145$ |
| 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. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = \frac{(x_1+y_1)(x_2+y_2)}{1+x_1y_1+x_2y_2} \\ 0 \le y_i \le x_i, \quad i = 1,2 \end{array} \right\},$ $(x_1)^2 + (x_2)^2 \le 100$ $x_1 - x_2 \le 0$ $0 \le x_i, \quad i = 1,2$ | $F = -1.96$ $f = 1.96$ |
| 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. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = (y_1 - x_1 + 20)^2 + (y_2 - x_2 + 20)^2 \\ 2y_1 - x_1 + 10 \le 0 \\ 2y_2 - x_2 + 10 \le 0 \\ -10 \le y_i \le 20, \quad i = 1,2 \end{array} \right\},$ $x_1 + x_2 + y_1 - 2y_2 \le 40$ $0 \le x_i \le 50, \quad i = 1,2$ | $F = 0.0$ $f = 100.0$ |
| TP9 $n = 10$ $m = 10$ | $\underset{(x,y)}{\text{Minimize}}\ F(x,y) = \sum_{i=1}^{10} \left( \lvert x_i - 1 \rvert + \lvert y_i \rvert \right),$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = e^{\left(1 + \frac{1}{4000} \sum_{i=1}^{10} (y_i)^2 - \prod_{i=1}^{10} \cos(\frac{y_i}{\sqrt{i}})\right) \sum_{i=1}^{10} (x_i)^2} \\ -\pi \le y_i \le \pi, \quad i = 1,2\ldots,10 \end{array} \right\},$ | $F = 0.0$ $f = 1.0$ |
| TP10 $n = 10$ $m = 10$ | $\underset{(x,y)}{\text{Minimize}}\ F(x,y) = \sum_{i=1}^{10} \left( \lvert x_i - 1 \rvert + \lvert y_i \rvert \right),$ s.t. $y \in \underset{(y)}{\text{argmin}} \left\{ \begin{array}{l} f(x,y) = e^{\left(1 + \frac{1}{4000} \sum_{i=1}^{10} (y_i x_i)^2 - \prod_{i=1}^{10} \cos(\frac{y_i x_i}{\sqrt{i}})\right)} \\ -\pi \le y_i \le \pi, \quad i = 1,2\ldots,10 \end{array} \right\},$ | $F = 0.0$ $f = 1.0$ |

TABLE II

PERFORMANCE OF MODIFIED-BLEAQ ON STANDARD TEST PROBLEMS (TP). UPPER LEVEL FUNCTION EVALUATIONS ARE DENOTED AS ULFE AND LOWER LEVEL FUNCTION EVALUATIONS ARE DENOTED AS LLFE.

| | Best (LLFE) | Best (ULFE) | Mean (LLFE) | Mean (ULFE) | Worst (LLFE) | Worst (ULFE) | UL Mean. Acc. | LL Mean. Acc. |
|---|---|---|---|---|---|---|---|---|
| TP1 | 12408 | 715 | 12273 | 721 | 19868 | 1239 | 0.000000 | 0.000000 |
| TP2 | 10175 | 1298 | 11112 | 1426 | 13782 | 2270 | 0.012535 | 0.000130 |
| TP3 | 3796 | 323 | 3403 | 318 | 5689 | 476 | 0.000000 | 0.000000 |
| TP4 | 12091 | 215 | 11523 | 254 | 12149 | 320 | 0.040435 | 0.007425 |
| TP5 | 8178 | 438 | 14106 | 1203 | 12202 | 1378 | 0.008348 | 0.038291 |
| TP6 | 12846 | 215 | 14686 | 265 | 16539 | 327 | 0.000101 | 0.000333 |
| TP7 | 186032 | 3106 | 218169 | 3616 | 257227 | 4371 | 0.089834 | 0.089556 |
| TP8 | 8274 | 1231 | 10069 | 1398 | 15900 | 1967 | 0.001758 | 0.000066 |
| TP9 | 76553 | 464 | 73972 | 627 | 93016 | 736 | 0.000012 | 0.000000 |
| TP10 | 73586 | 498 | 87882 | 549 | 92252 | 611 | 0.000106 | 0.000000 |

TABLE III

COMPARISON OF MODIFIED-BLEAQ ($BLEAQ^{AR+LS}$) AGAINST THE RESULTS ACHIEVED BY ORIGINAL-BLEAQ ($BLEAQ$), WJL, WLD AND NBLEA APPROACH ON STANDARD TEST PROBLEMS (TP).

| | MFE=Mean(ULFE + LLFE) | MFER= $\frac{BLEAQ}{BLEAQ^{AR+LS}}$ | $\frac{WJL}{BLEAQ^{AR+LS}}$ | $\frac{WLD}{BLEAQ^{AR+LS}}$ | $\frac{NBLEA}{BLEAQ^{AR+LS}}$ |
|---|---|---|---|---|---|
| TP1 | 14142 | 1.13 | 6.15 | 6.09 | 11.40 |
| TP2 | 14394 | 1.15 | 17.20 | 11.90 | 16.86 |
| TP3 | 3741 | 1.26 | 24.63 | 25.62 | 32.27 |
| TP4 | 13954 | 1.15 | 20.71 | 15.19 | 19.55 |
| TP5 | 14540 | 1.17 | 5.52 | 4.78 | 10.19 |
| TP6 | 14642 | 1.22 | 11.28 | 4.50 | 12.38 |
| TP7 | 213497 | 1.23 | 5.13 | 4.42 | 4.05 |
| TP8 | 11933 | 1.16 | 17.79 | 15.26 | 26.70 |
| TP9 | 87259 | 1.14 | - | 4.04 | 7.62 |
| TP10 | 78372 | 1.28 | - | 5.92 | 7.65 |

TABLE IV

PERFORMANCE OF MODIFIED-BLEAQ ($BLEAQ^{AR+LS}$) AGAINST THE RESULTS ACHIEVED BY ORIGINAL-BLEAQ ($BLEAQ$) AND NBLEA APPROACH ON SMD1-SMD6 TEST PROBLEMS.

| | LL Mean Acc. | UL Mean Acc. | Mean LLFE | Mean ULFE | MFER= $\frac{BLEAQ}{BLEAQ^{AR+LS}}$ | $\frac{NBLEA}{BLEAQ^{AR+LS}}$ |
|---|---|---|---|---|---|---|
| SMD1 | 0.003244 | 0.006351 | 92234 | 675 | 1.23 | 19.26 |
| SMD2 | 0.003093 | 0.003035 | 77329 | 506 | 1.18 | 19.12 |
| SMD3 | 0.004325 | 0.009754 | 111955 | 750 | 1.17 | 11.21 |
| SMD4 | 0.002864 | 0.006728 | 68359 | 683 | 1.11 | 15.05 |
| SMD5 | 0.003584 | 0.004232 | 113607 | 579 | 1.16 | 16.21 |
| SMD6 | 0.000007 | 0.000012 | 103729 | 880 | 1.23 | 22.85 |

of archiving and local search in BLEAQ has led to an improved performance on all the test problems chosen in this paper. Though the modified BLEAQ and original BLEAQ approaches are based on quadratic approximations of the lower level optimal variables with respect to upper level variables, they are still applicable to generic bilevel problems without the restrictions of differentiability, convexity etc. A proper choice of the crossover and mutation operators will make the procedures applicable for discrete bilevel problems as well. Integrating concepts from classical optimization into evolutionary algorithms have already been found to be of immense significance. We hope that similar hybridizations in the field of evolutionary bilevel optimization could lead to algorithms that are efficient and non-restrictive.

## REFERENCES


[1] E. Aiyoshi and K. Shimizu. Hierarchical decentralized systems and its new solution by a barrier method. *IEEE Transactions on Systems, Man, and Cybernetics*, 11:444–449, 1981.

[2] E. Aiyoshi and K. Shimuzu. A solution method for the static constrained stackelberg problem via penalty method. *IEEE Transactions on Automatic Control*, AC-29(12):1112–1114, 1984.

[3] M. A. Amouzegar. A global optimization method for nonlinear bilevel programming problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 29(6):771–777, 1999.

[4] 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.

[5] 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.

[6] J. Bard. *Practical Bilevel Optimization: Algorithms and Applications*. The Netherlands: Kluwer, 1998.

[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. F. Bard. Coordination of multi-divisional firm through two levels of management. *Omega*, 11(5):457–465, 1983.
[9] J. F. Bard. *Practical Bilevel Optimization: Algorithms and Applications*. Kluwer Academic Publishers, 1998.
[10] M. P. Bendsoe. Optimization of structural topology, shape, and material. Technical report, 1995.
[11] L. Bianco, M. Caramia, and S. Giordani. A bilevel flow model for hazmat transportation network design. *Transportation Research Part C: Emerging technologies*, 17(2):175–196, 2009.
[12] J. Bracken and J. McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21:37–44, 1973.
[13] L. Brotcorne, M. Labbe, P. Marcotte, and G. Savard. A bilevel model for toll optimization on a multicommodity transportation network. *Transportation Science*, 35(4):345–358, 2001.
[14] 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.
[15] G. Brown, M. Carlyle, R. C. Harney, E. Skroch, and K. Wood. Interdicting a Nuclear-Weapons Project. *Operations Research*, 57(4):866–877, 2009.
[16] S. Christiansen, M. Patriksson, and L. Wynter. Stochastic bilevel programming in structural optimization. Technical report, Structural and Multidisciplinary Optimization, 1997.
[17] B. Colson, P. Marcotte, and G. Savard. An overview of bilevel optimization. *Annals of Operational Research*, 153:235–256, 2007.
[18] I. Constantin and M. Florian. Optimizing frequencies in a transit network: a nonlinear bi-level programming approach. *International Transactions in Operational Research*, 2(2):149 – 164, 1995.
[19] K. Deb and A. Sinha. An efficient and accurate solution methodology for bilevel multi-objective programming problems using a hybrid evolutionary-local-search algorithm. *Evolutionary Computation Journal*, 18(3):403–449, 2010.
[20] S. Dempe. *Foundations of Bilevel Programming*. Kluwer Academic Publishers, Secaucus, NJ, USA, 2002.
[21] S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52(3):339–359, 2003.
[22] S. Dempe, J. Dutta, and S. Lohse. Optimality conditions for bilevel programming problems. *Optimization*, 55(5âĂŞ6):505âĂŞ–524, 2006.
[23] D. Fudenberg and J. Tirole. *Game theory*. MIT Press, 1993.
[24] W. Halter and S. Mostaghim. Bilevel optimization of multi-component chemical systems using particle swarm optimization. In *Proceedings of World Congress on Computational Intelligence (WCCI-2006)*, pages 1240–1247, 2006.
[25] J. Herskovits, A. Leontiev, G. Dias, and G. Santos. Contact shape optimization: A bilevel programming approach. *Struct Multidisc Optimization*, 20:214–221, 2000.
[26] C. Kirjner-Neto, E. Polak, and A. Kiureghian. An outer approximations approach to reliability-based optimal design of structures. *Journal of Optimization Theory and Applications*, 98(1):1–16, 1998.
[27] F. Legillon, A. Liefooghe, and E.-G. Talbi. Cobra: A cooperative coevolutionary algorithm for bi-level optimization. In *2012 IEEE Congress on Evolutionary Computation (CEC-2012)*. IEEE Press, 2012.
[28] H. Li and Y. Wang. A genetic algorithm for solving a special class of nonlinear bilevel programming problems. In Y. Shi, G. Albada, J. Dongarra, and P. Sloot, editors, *Computational Scienceï£¡ AI ICCS 2007*, volume 4490 of *Lecture Notes in Computer Science*, pages 1159–1162. Springer Berlin Heidelberg, 2007.
[29] H. Li and Y. 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.
[30] H. Li and Y. Wang. An evolutionary algorithm with local search for convex quadratic bilevel programming problems. *Applied Mathematics and Information Sciences*, 5(2):139–146, 2011.
[31] B. D. Liu. Stackelbergï£¡nash equilibrium for multi-level programming with multiple followers using genetic algorithms. *Computers and Mathematics with Applications*, 36(7):79–89, 1998.
[32] J. Lowe. Homeland Security: Operations Research Initiatives and Applications. *Interfaces*, 36(6):483–485, 2006.
[33] R. Mathieu, L. Pittard, and G. Anandalingam. Genetic algorithm based approach to bi-level linear programming. *Operations Research*, 28(1):1–21, 1994.
[34] A. Migdalas. Bilevel programming in traffic planning: Models, methods and challenge. *Journal of Global Optimization*, 7(4):381–405, 1995.
[35] V. Oduguwa and R. Roy. Bi-level optimization using genetic algorithm. In *Proceedings of the 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002)*, pages 322–327, 2002.
[36] J. V. Outrata. On the numerical solution of a class of stackelberg problems. *Zeitschrift Fur Operation Research*, AC-34(1):255ï£¡278, 1990.
[37] S. Ruuska and K. Miettinen. Constructing evolutionary algorithms for bilevel multiobjective optimization. In *Evolutionary Computation (CEC), 2012 IEEE Congress on*, pages 1–7, june 2012.
[38] X. Shi. and H. S. Xia. Model and interactive algorithm of bi-level multi-objective decision-making with multiple interconnected decision makers. *Journal of Multi-Criteria Decision Analysis*, 10(1):27–34, 2001.
[39] K. Shimizu and E. Aiyoshi. A new computational method for stackelberg and min-max problems by use of a penalty method. *IEEE Transactions on Automatic Control*, AC-26(2):460–466, 1981.
[40] A. Sinha and K. Deb. Towards understanding evolutionary bilevel multi-objective optimization algorithm. In *IFAC Workshop on Control Applications of Optimization (IFAC-2009)*, volume 7. Elsevier, 2009.
[41] A. Sinha, P. Malo, and K. Deb. Unconstrained scalable test problems for single-objective bilevel optimization. In *2012 IEEE Congress on Evolutionary Computation (CEC-2012)*. IEEE Press, 2012.
[42] A. Sinha, P. Malo, and K. Deb. Efficient evolutionary algorithm for single-objective bilevel optimization. *CoRR*, abs/1303.3901, 2013.
[43] A. Sinha, P. Malo, and K. Deb. Test problem construction for single-objective bilevel optimization. *Evolutionary Computation Journal*, 2014 (In Press).
[44] 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.
[45] A. Sinha, P. Malo, A. Frantsev, and K. Deb. Finding optimal strategies in a multi-period multi-leader-follower stackelberg game using an evolutionary algorithm. *Computers & Operations Research*, 41:374–385, 2014.
[46] W. Smith and R. Missen. *Chemical Reaction Equilibrium Analysis: Theory and Algorithms*. John Wiley & Sons, New York, 1982.
[47] H. v. Stackelberg. *The theory of the market economy.* Oxford University Press, New York, 1952.
[48] H. Sun, Z. Gao, and J. Wu. A bi-level programming model and solution algorithm for the location of logistics distribution centers. *Applied Mathematical Modelling*, 32(4):610 – 616, 2008.
[49] L. N. Vicente and P. H. Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization*, 5(3):291–306, 2004.
[50] F. J. Wang and J. Periaux. Multi-point optimization using gas and Nash/Stackelberg games for high lift multi-airfoil design in aerodynamics. In *Proceedings of the 2001 Congress on Evolutionary Computation (CEC-2001)*, pages 552–559, 2001.
[51] G. Wang, Z. Wan, X. Wang, and Y. Lv. Genetic algorithm based on simplex method for solving linear-quadratic bilevel programming problem. *Comput Math Appl*, 56(10):2550–2555, 2008.
[52] 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.
[53] Y. Wang, H. Li, and C. 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.
[54] L. Wein. Homeland Security: From Mathematical Models to Policy Implementation: The 2008 Philip McCord Morse Lecture. *Operations Research*, 57(4):801–811, 2009.
[55] Y. Yin. Genetic algorithm based approach for bilevel programming models. *Journal of Transportation Engineering*, 126(2):115–120, 2000.
[56] T. Zhang, T. Hu, Y. Zheng, and X. Guo. An improved particle swarm optimization for solving bilevel multiobjective programming problem. *Journal of Applied Mathematics*, 2012.