Evolutionary Bilevel Optimization Using KKT Proximity Measure

Ankur Sinha[1], Tharo Soun[2] and Kalyanmoy Deb[2]
[1]Production and Quantitative Methods
Indian Institute of Management, Ahmedabad 380015 India
asinha@iima.ac.in
[2]Department of Electrical and Computer Engineering
Michigan State University, East Lansing, MI, USA
sounthar@msu.edu,kdeb@egr.msu.edu

COIN Report 2017002

Abstract— **Bilevel optimization problems are often reduced to single level using Karush-Kuhn-Tucker (KKT) conditions; however, there are some inherent difficulties when it comes to satisfying the KKT constraints strictly. In this paper, we discuss single level reduction of a bilevel problem using approximate KKT conditions which have been recently found to be more useful than the original and strict KKT conditions. We embed the recently proposed KKT proximity measure idea within an evolutionary algorithm to solve bilevel optimization problems. The idea is tested on a number of test problems and comparison results have been provided against a recently proposed evolutionary algorithm for bilevel optimization. The proposed idea leads to significant savings in lower level function evaluations and shows promise in further use of KKT proximity measures in bilevel optimization algorithm development.**

I. Introduction

There has been growing interest in bilevel optimization as a number of new applications problems have been encountered recently that involve one optimization problem as a constraint to another optimization problem in a nested manner. These problems can be found in a wide variety of applications, for instance, in the area of defense [5], [30], transportation [16], [4], investigation of strategic behavior in deregulated markets [12], model production processes [17], chemical engineering [27], [7], and optimal tax policies [14], [26], [21], among others.

In the domain of mathematical optimization a significant amount of literature exists on bilevel optimization [2], [8]. However, most of the algorithmic progress is limited to only simple instances of bilevel problems with objective functions and constraints being linear, quadratic and/or convex. It is noteworthy that even such simple instances of bilevel programs can be quite challenging to solve. Since the 1990s researchers have started to solve bilevel optimization problems using evolutionary techniques [15]. However, most of these algorithms are nested in nature and are computationally very expensive. Recently, there have been a number of attempts to exploit the structure and properties of bilevel problems to reduce the nested requirement and solve the problem more efficiently using evolutionary algorithms [24], [22], [25], [23]. This study is an attempt in a similar direction, where we attempt to reduce bilevel problems to single level using a recently proposed KKT proximity measure [9]. Substituting the lower level optimization problem using its respective KKT conditions have been a common approach to solve bilevel problems, both in classical [1], [3], [28] and evolutionary literature [11], [29], [13]. However, KKT conditions introduce a number of strict equality constraints that can be difficult to satisfy. Also KKT conditions involve a constraint qualification (regularity) requirement which is difficult to pass on the upper level. Although the second requirement is difficult to handle, to counter the first difficulty we take an approximate KKT approach and reduce the bilevel problem to single level.

The paper is organized as follows. To begin with, we introduce the bilevel formulation, following which we discuss various ways to apply the approximate KKT measure to bilevel problems. This is followed by embedding the idea in an evolutionary algorithm and results on a wide variety of bilevel problems. A comparative study has also been performed against a recently proposed evolutionary algorithm.

II. Bilevel Formulation

While solving bilevel optimization problems, there are two extreme positions that are commonly analyzed in the literature; the *optimistic* and the *pessimistic* positions. In the optimistic position it is assumed that the lower level (follower) chooses an optimal solution, which leads to the best objective value for the leader. In the pessimistic position it is assumed that the lower level may choose an optimal solution, which leads to the worst objective value for the leader. In the optimistic case some form of cooperation between the two players is assumed, while in the pessimistic case no such cooperation is assumed and the target is to find the worst case solution. Optimistic bilevel optimization is much more tractable as compared to pessimistic bilevel optimization; therefore, in this paper we handle the optimistic problem. Bilevel optimization contains two levels such that one level of optimization acts as a constraint to an outer level of optimization. The outer or the upper level has its own objective, constraints and variables, and similarly the inner or the lower level has its own objective, constraints and variables. However, both kinds of variables can be present at either of the levels. The lower level optimization is a parametric optimization task for which the upper level variables act as parameters and the problem has to be solved with respect to the lower level variables. Therefore, the lower level optimal solutions are dependent on the upper level variables. The upper level has to be optimized with respect to upper level variables and optimal lower level variables. A general formulation for optimistic bilevel optimization has been provided below.

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

$$\begin{aligned} &\min_{x,y} F(x,y) \\ &\text{subject to} \\ &\quad y \in \underset{y}{\operatorname{argmin}}\{f(x,y) : g_j(x,y) \leq 0, j = 1, \ldots, J\} \\ &\quad G_k(x,y) \leq 0, k = 1, \ldots, K \end{aligned}$$

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

In bilevel optimization, it is possible to replace the lower level problem in Definition 1 with its KKT conditions, when it adheres to certain convexity and regularity conditions. In such a case, the bilevel optimization problem can be reduced to single level and can be written as follows.

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

$$\begin{aligned} &\min_{x,y,\lambda} \quad F(x,y) \\ &\text{subject to} \\ &\quad G_k(x,y) \leq 0, k = 1, \ldots, K, \\ &\quad g_j(x,y) \leq 0, j = 1, \ldots, J, \\ &\quad \nabla_y L(x,y,\lambda) = 0, \\ &\quad \lambda_j g_j(x,y) = 0, j = 1, \ldots, J, \\ &\quad \lambda_j \geq 0, j = 1, \ldots, J, \\ &\text{where} \end{aligned}$$

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

The above formulation can be difficult to handle, as the Lagrangian constraint ($\nabla_y L(x,y,\lambda) = 0$) may lead to non-convexities and the complementarity slackness conditions ($\lambda_j g_j(x,y) = 0$) introduce combinatorial variables (in case of linearization). In this paper, we will further develop on KKT based single level reduction to solve the bilevel optimization problem. For linear bilevel optimization problems, the Lagrangian constraint is also linear; therefore, the single-level reduced problem is a mixed-integer linear program. Approaches based on vertex enumeration [3], [6], [28], and branch-and-bound [1], [10] have been proposed to solve these problems. Evolutionary optimization studies that replace the lower level problem with its KKT conditions and solve the single-level problem include [11], [29], [13]

III. KKT based Proximity Measure

The KKT conditions are necessary conditions in mathematical programming for a solution to be optimal. The conditions are known to be satisfied exactly at any given local or global optimal provided that a suitable constraint qualification condition holds. In a recent study, Dutta et al. (2013) [9] studied the regularity of the KKT conditions in the neighborhood of the KKT point. The aim of their study was to investigate as to what extent the KKT conditions are violated if one approaches the KKT point through a series of iterates. Using the KKT conditions, the authors have derived a simple KKT-proximity metric that provides an idea about the vicinity of an iterate from the optimum. Using this notion, a point in the vicinity of a KKT point is referred as ϵ-KKT point. The definition for the approximate KKT measure has been shown to be valid for smooth as well as non-smooth cases. The intention of the authors for proposing an approximate KKT measure was to provide a convergence metric that can act as a termination criteria for smooth as well as non-smooth optimization problems.

To begin with, we introduce the idea of using KKT proximity measure in the context of smooth single-level optimization problems and then extend its usage for bilevel optimization problems. Consider the following general optimization problem:

$$\begin{aligned} &\min_{y} \quad f(y) \\ &\text{subject to} \quad g_j(y) \leq 0, j = 1, \ldots, J, \end{aligned}$$

With some assumptions on the properties of the above formulation it is possible rewrite the above problem using its first-order conditions and the KKT proximity measure ϵ.

$$\begin{aligned} &\min_{y} \quad \epsilon \\ &\text{subject to} \\ &\quad g_j(y) \leq 0, j = 1, \ldots, J, \\ &\quad ||\nabla_y L(y,\lambda)||^2 \leq \epsilon, \\ &\quad \sum_{j=1}^{J} \lambda_j g_j(y) \geq -\epsilon \\ &\quad \lambda_j \geq 0, j = 1, \ldots, J, \\ &\text{where} \qquad L(y,\lambda) = f(y) + \sum_{j=1}^{J} \lambda_j g_j(y). \end{aligned}$$

Solving the above optimization problem provides us the proximity measure along with the Lagrange multipliers. The closer the proximity measure is to zero, it gives us an idea about the closeness of the solution to the optimum. For the extension of the above measure to non-smooth problems, the readers may refer to [9]. There can be a number of ways in which ϵ-KKT idea can be beneficial for bilevel optimization problems; below we describe two ideas that will be later used to develop a bilevel optimization algorithm.

A. *Using KKT Proximity Measure as a Termination Condition for the Lower Level Optimization Calls*

While applying evolutionary algorithms to bilevel problems, one often requires to solve the lower level optimization problem multiple times. One of the important classes of solution procedures for bilevel optimization are nested methods that solve the lower level optimization problem for every upper level vector generated during the upper level search process. Even approaches that are not nested require a substantial number of calls to the lower level optimization problem. Termination of the lower level optimization task for any given upper level decision is often based on expected improvement over generations of the evolutionary algorithm, i.e. variance of the population members becoming small, no improvement in a pre-specified number of generations etc. Such termination methods do not provide sufficient confidence about the proximity of the best obtained solution to the optimum. Moreover, having a very strict termination criteria based on above methods often leads to wastage of computational resources. Errors made in identifying the lower level optimal solution may lead to issues in converging toward the bilevel optimum. Allowing the lower level to run until a pre-specified ϵ value of KKT proximity measure is achieved would provide sufficient confidence and is less likely to induce errors.

B. *Using KKT Proximity Measure as a Constraint Parameter*

In the previous section, we provided a brief discussion about the reduction of bilevel optimization problem to a single-level optimization problem using the KKT conditions. However, there are a number of difficulties associated with solving such a reduced problem, for instance, handling of the stationarity and the complementary slackness conditions is not straightforward, all of which appear as an equality. The ϵ-KKT approach provides us a single parameter ϵ to control the violations in these constraints that limits the errors emanating from the approximate lower level optimum. Given a fixed parameter ϵ, the bilevel optimization problem in this case can be reduced to a single level as follows:

$$
\begin{aligned}
&\min_{x,y,\lambda} \quad F(x,y) \\
&\text{subject to} \\
&\quad G_k(x,y) \le 0, k = 1,\ldots,K, \\
&\quad g_j(x,y) \le 0, j = 1,\ldots,J, \\
&\quad ||\nabla_y L(x,y,\lambda)||^2 \le \epsilon, \\
&\quad \sum_{j=1}^{J} \lambda_j g_j(x,y) \ge -\epsilon \\
&\quad \lambda_j \ge 0, j = 1,\ldots,J, \\
&\quad \epsilon \le \epsilon_0
\end{aligned}
$$

where

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

In this paper, we will utilize the above two strategies, i.e. using ϵ-KKT as a termination criteria at the lower level and reducing the bilevel problem to single level for local search during an evolutionary bilevel optimization procedure.

IV. Bilevel Evolutionary Algorithm based on KKT Proximity Measure as a Constraint Parameter

In this section, we provide the bilevel evolutionary algorithm that utilizes ϵ-KKT formulation to solve bilevel optimization problems. A stepwise procedure for the implementation of the algorithm is provided in Table I. Below we provide a detailed explanation for some of the important steps in the algorithm.

A. *Initialization*

During the initialization stage, some of the population members at the upper level are created in the relaxed feasible region by solving the following optimization problem.

$$
\begin{aligned}
&\min_{x,y} \; 0 \\
&\text{subject to} \\
&\quad g_j(x,y) \le 0, j = 1,\ldots,J, \\
&\quad G_k(x,y) \le 0, k = 1,\ldots,K.
\end{aligned}
$$

The optimization is performed using sequential quadratic programming (SQP) that terminates as soon as a feasible solution is found. The above problem is solved using random starting points leading to different points in the relaxed feasible space.

B. *Local Approximation of Functions and Constraints*

Note that the ϵ-KKT method requires gradients for the functions and constraints. In our algorithm we do not compute the gradients, rather create a sample of points around a point (x, y) and find a quadratic approximation for the objective functions and linear approximation of the constraints, i.e. we create a bilevel optimization problem that locally approximates the original bilevel problem. Let the following bilevel problem be

TABLE I

STEP-BY-STEP PROCEDURE FOR ϵ-KKT BASED BILEVEL ALGORITHM

<table>
<tr><th>Step</th><th>Description</th></tr>
<tr><td>1</td><td>Initialization:
Randomly generate upper level (x) and lower level members (y) in the population $\mathcal{P}$. At least half of the members are created in the relaxed feasible region. Initialize the generation counter $g \leftarrow 0$ (see Section IV-A).

(a) For each $x^{(i)} \in \mathcal{P}$, find the optimal lower level member. The optimal lower level member replaces the random $y^{(i)}$ created in the previous step.
(b) Create a random sample of points $(x^{(is)}, y^{(is)}) : s \in \{1, \ldots, S\}$ around each point $(x^{(i)}, y^{(i)})$. Compute their upper and lower level function values and constraints, and add all the points to A_0.
(c) For each member for which lower level is solved and optimization is successful, tag the vector $(x^{(i)}, y^{(i)})$ as 1; otherwise, tag it as 0. Add the tag 0 members to tag 0 archive $\mathcal{A}_0$ and the tag 1 members to tag 1 archive $\mathcal{A}_1$.
(d) Assign fitness to all the population members based on upper level function and constraints (see Section IV-E).
(e) For each vector $(x^{(i)}, y^{(i)}) \in \mathcal{P}$, create local quadratic-linear approximations of objectives and constraints. (see Section IV-B)</td></tr>
<tr><td>2</td><td>Reproduction:
Increment the generation counter by one: $g \leftarrow g + 1$.

(a) Parent selection: Randomly pick 2μ members from $\mathcal{P}$. Perform tournament selection from the picked members based on their tags and fitness. This results in μ parents for crossover. Tag 1 member is given priority over tag 0 member during comparison. If the tags of the members being compared are the same then their fitness values are compared.
(b) Offspring generation: Create λ offspring from the μ parents using genetic operations (see Section IV-E). Note that genetic operations at the upper level are performed only with upper level variables.</td></tr>
<tr><td>3</td><td>Update Offsprings:
For each offspring, two methods are used to find its lower level counterpart and update the population $\mathcal{P}$.

(a) Lower Level Optimization: There is a probability γ that the lower level optimization is performed for the offspring to find the lower level counterpart. If the lower level optimization is successful, the solution is compared using fitness with the worst tag 1 member of the population $\mathcal{P}$, and it replaces that member if it is better. If the lower level optimization is unsuccessful follow a similar process with the worst Tag 0 member in $\mathcal{P}$. The offspring is also added to the archive A_0 or A_1 based on its success flag.
(b) Approximations: This step is performed if Step 3a is not executed. For a given offspring, using the closest members in A_0 compute local approximations for the lower level objective (quadratic approximation) and constraints (linear approximation). The locally approximated quadratic-linear lower level problem is solved using SQP to get the lower level counterpart for the offspring. Fitness is assigned and the offspring is added to $\mathcal{A}_0$. For each offspring, randomly pick a tag 0 member from $\mathcal{P}$ and compare the fitness. If the offspring is better, then it replaces the chosen tag 0 member in the population.</td></tr>
<tr><td>4</td><td>ϵ-KKT based Local Search
The following search is performed every α generations: $g \bmod \alpha = 0$.

(a) Randomly choose 2 members from the population $\mathcal{P}$. Perform tournament selection between the two members. Tag 1 member is given priority when compared. Let the winning member be $(x^{(w)}, y^{(w)})$
(b) Get new local quadratic-linear approximations of objectives and constraints for the winning member $(x^{(w)}, y^{(w)})$. (see Section IV-B)
(c) Perform ϵ-KKT based local search with $(x^{(w)}, y^{(w)})$ as the initial point with its approximated objectives and constraints. If the search succeeds, divide ϵ by 10. Otherwise, multiply ϵ by 10 (see Section IV-B). Let the member produced from the search be $(x^{(v)}, y^{(v)})$
(d) Find the optimal lower level solution for $x^{(v)}$ by solving the lower level optimization problem with $y^{(v)}$ as the starting point. If lower level optimization is successful then the member is tagged as 1 otherwise 0. The best lower level member obtained from the lower level optimization task replaces $y^{(v)}$.
(e) Assign fitness to $(x^{(v)}, y^{(v)})$ based on its upper level objective and constraints values. If both $(x^{(w)}, y^{(w)})$ and $(x^{(v)}, y^{(v)})$ have same tags then use their fitness values to choose the better member; otherwise the member with tag 1 is chosen as the better member. If $(x^{(v)}, y^{(v)})$ is better it replaces $(x^{(w)}, y^{(w)})$ in the population.
(f) Add the solution $(x^{(v)}, y^{(v)})$ to its corresponding archive.</td></tr>
<tr><td>5</td><td>Update best member:
Identify the Tag 1 member in the current generation in $\mathcal{P}$ with the best fitness, denoted as (x_{best},y_{best}).</td></tr>
<tr><td>6</td><td>Termination check:
Perform a termination check (see Section IV-F). If false, proceed to the next generation (Step 2). If true, report the best member as the bilevel optimum after performing ϵ-KKT based local search on the best member, as discussed in Step 4.</td></tr>
</table>

an approximate representation of the original bilevel problem around a point (x, y).

$$\begin{aligned}
&\min_{x,y} \hat{F}(x, y) \\
&\text{subject to} \\
&y \in \underset{y \in Y}{\text{argmin}}\{\hat{f}(x, y) : \hat{g}_j(x, y) \leq 0, j = 1, \ldots, J\} \\
&\hat{G}_k(x, y) \leq 0, k = 1, \ldots, K
\end{aligned}$$

Note that if the objective functions are quadratic and the constraints are linear then the above approximate bilevel problem can be written as follows:

$$\begin{aligned}
&\min_{x,y} \hat{F}(x, y) = (x, y)R(x, y)' + (x, y)S + T \\
&\text{subject to} \\
&y \in \underset{y}{\text{argmin}}\{\hat{f}(x, y) = (x, y)A(x, y)' + (x, y)B + C : \\
&\qquad \hat{g}(x, y) = (x, y)D + E \leq 0, \} \\
&\hat{G}(x, y) = (x, y)U + V \leq 0,
\end{aligned}$$

where R and A are square matrices, S, B, D and U are column vectors, and T, C, E and V are constants.

Applying the ϵ-KKT reduction to the above quadratic bilevel problem we obtain a single-level reduction with non-linear constraints. It is possible to solve such a reduced single-level problem using SQP; however, without guaranteeing the optimum if the problem is non-convex. The presence of non-linear complementary slackness conditions may make the problem non-convex. There is a possibility to linearize the complementary slackness condition using the big-M method, but it would lead to inclusion of combinatorial variables in the problem. In this paper, we do not linearize the complementary slackness constraints rather solve the reduced bilevel problem directly using SQP.

C. Parameters

The following are the parameters used in the algorithm that have been kept fixed throughout the computations in this study:

$N = 20$ (Population size at upper level)
$n = 20$ (Population size at lower level)
$\epsilon_0 = 10^{-3}$ (Value at start) Range: $10^{-6} \leq \epsilon \leq 1$
$m = 0.1$ (Probablity of mutation)
$c = 0.9$ (Probablity of crossover)
$\gamma = 0.2$ (Probability of lower level optimization)
$\alpha = 5$ (Generations between local search)

D. Lower Level Search

The lower level optimization, whenever required, is done using SQP or an evolutionary algorithm. For details about switching between SQP and evolutionary algorithms for lower level optimization we refer the readers to [24].

E. Genetic Operators and Fitness Assignment

The genetic operations at the upper level are performed only with upper level variables, and at the lower level only with respect to the lower level variables. The genetic operations and fitness assignment used in this algorithm are same as that in [24].

F. Termination

The lower level optimization task, whenever performed, is terminated based on the ϵ-KKT idea as discussed in Section III-A. The termination at lower level occurs if $\epsilon \leq \epsilon_0$ is encountered during the optimization process. At the upper level, the termination is based on improvement at the upper level. If the function value does not improve for 20 consecutive generations, the upper level search is terminated and the algorithm stops.

V. Results

In this section, we provide the results for the ϵ-KKT based algorithm on a couple of test-suites and compare the results against the bilevel evolutionary algorithm based on quadratic approximations (BLEAQ) [18], [19], [24]. First, we evaluate the proposed algorithm on a test set of 8 simple bilevel problems. These bilevel problems are well studied in the literature and their best solutions are available. We refer this test set as TP and provide the results through Table III and Figure 1. Table III provides the comparison of the proposed approach against BLEAQ, where it compares the median function evaluations required at the upper and lower level separately from 31 runs on each test problem. Interestingly, the proposed algorithm performs worse at the upper level for a number of test problems but beats BLEAQ significantly in terms of lower level function evaluations. One of the reasons for higher function evaluations at the upper level is the requirement of sample points around population members to locally approximate the functions and constraints. Since most of the test problems contain quadratic/linear functions and constraints, the local approximation for ϵ-KKT approach represents the exact problem and is able to solve many of the problems as soon as local search is performed. However, our implementation utilizes α generations of evolutionary search before a local search. Note that just a single local search will not guarantee the bilevel optimum as the reduced bilevel problems are non-convex. For example, in case of TP2 we observed that the local search was often terminating on the local points if applied alone without embedding in the evolutionary approach. The bilevel optimum and the local points for this problem are provided in Table II, and out of 100 runs with different starting points we observed that the local search when performed without embedding in an evolutionary approach could find the right optimum only in two of the cases. However, the proposed algorithm could find the bilevel optimum in 70% of the runs for this test problem. For all the other cases, the algorithm had a 100% success rate. Figure 1 provides further details including the minimum and maximum function evaluations required at both levels for the TP suite.

TABLE II

RESULTS ON TP2 (100 STARTING POINTS) WHEN ϵ-KKT LOCAL SEARCH IS PERFORMED ALONE USING SQP WITHOUT EMBEDDING IN THE EVOLUTIONARY APPROACH.

	UL Vector		LL Vector		UL Func. Val.	LL Func. Val.	Count (out of 100)
Local	25	30	5	10	5	0	97
Local	0	0	-10	-10	0	200	1
Global	0	30	-10	10	0	100	2

TABLE III

TP TEST SUITE RESULTS

Problem	UL Func. Eval. (Median)		LL Func. Eval. (Median)		Ratio		
	ϵ-KKT	BLEAQ	ϵ-KKT	BLEAQ	Upper	Lower	Upper+Lower
TP 1	369	265	565	1276	1.39	0.44	**0.61**
TP 2	830	870	1465	2295	0.95	0.64	**0.73**
TP 3	333	256	512	1239	1.30	0.41	**0.57**
TP 4	756	358	1626	1856	2.11	0.88	1.08
TP 5	583	867	659	2361	0.67	0.28	**0.38**
TP 6	597	289	638	3270	2.07	0.20	**0.35**
TP 7	434	368	1357	2878	1.18	0.47	**0.55**
TP 8	826	692	956	2176	1.19	0.44	**0.62**

TABLE IV

SMD TEST SUITE RESULTS FOR TWO-VARIABLE UPPER LEVEL DIMENSION AND THREE-VARIABLE LOWER LEVEL DIMENSION

Problem	UL Func. Eval. (Median)		LL Func. Eval. (Median)		Ratio		
	ϵ-KKT	BLEAQ	ϵ-KKT	BLEAQ	Upper	Lower	Upper+Lower
SMD 1	912	156	9518	15456	5.85	0.62	**0.67**
SMD 2	668	168	4593	14175	3.98	0.32	**0.37**
SMD 3	1158	256	11532	18366	4.52	0.63	**0.68**
SMD 4	668	215	5156	16362	3.11	0.32	**0.35**
SMD 5	1159	156	15644	24368	7.43	0.64	**0.69**
SMD 6*	668	155	9452	14688	4.31	0.64	**0.68**
SMD 7	931	177	7292	16468	5.26	0.44	**0.49**
SMD 8	2248	385	28406	39246	5.84	0.72	**0.77**
SMD 9	410	229	5653	18346	1.79	0.31	**0.33**
SMD 10	3648	549	33803	42312	6.64	0.80	**0.87**
SMD 11	1294	389	23165	143246	3.33	0.16	**0.17**
SMD 12	6572	663	62126	132134	9.91	0.47	**0.52**

TABLE V

SMD TEST SUITE RESULTS FOR FIVE-VARIABLE UPPER LEVEL DIMENSION AND FIVE-VARIABLE LOWER LEVEL DIMENSION

Problem	UL Func. Eval. (Median)		LL Func. Eval. (Median)		Ratio		
	ϵ-KKT	BLEAQ	ϵ-KKT	BLEAQ	Upper	Lower	Upper+Lower
SMD 1	3187	466	27367	74268	6.84	0.37	**0.41**
SMD 2	1932	362	8649	71876	5.34	0.12	**0.15**
SMD 3	4810	720	41925	99269	6.68	0.42	**0.47**
SMD 4	2317	499	12221	64746	4.64	0.19	**0.22**
SMD 5	3374	467	28835	93540	7.22	0.31	**0.34**
SMD 6*	1838	2165	22680	71928	0.85	0.32	**0.33**

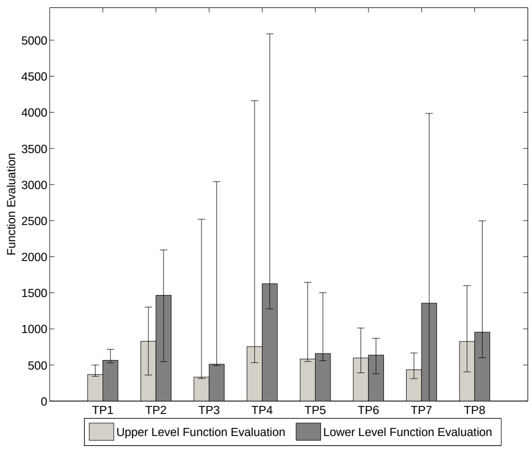

Fig. 1. Results of TP test problems showing the minimum, median and maximum function evaluations.

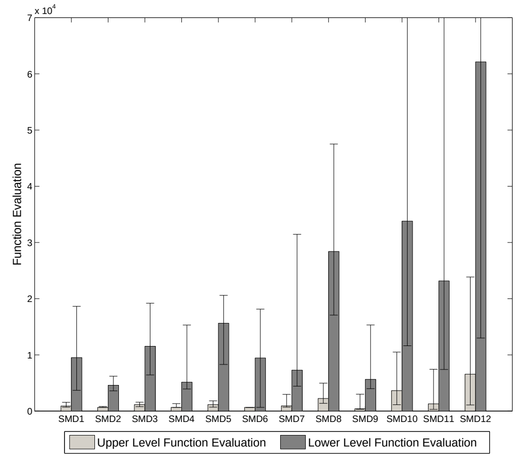

Fig. 2. Results of SMD test problems with 2 upper level variables and 3 lower level variables showing the minimum, median and maximum function evaluations.

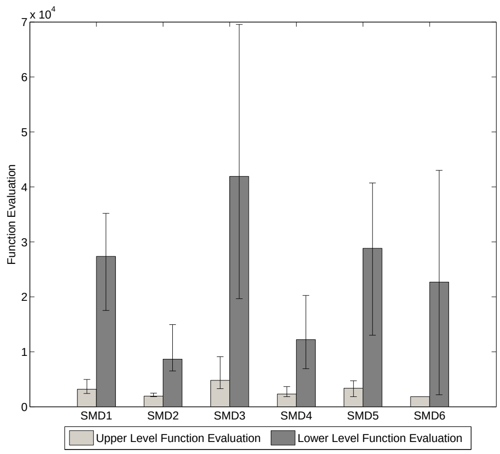

Fig. 3. Results of SMD test problems with five upper level variables and five lower level variables showing the minimum, median and maximum function evaluations.

Next, we study the performance of the algorithm on the scalable SMD test problems [20]. We break this study in two parts; first, we apply the algorithm on smaller instances of the SMD problems that contain two upper level and three lower level variables; second we apply the algorithm on larger instances of SMD problems that contain five upper and five lower level variables. The results have been provided through Tables IV and V, and Figures 2 and 3. Once again the observations are similar as before, where we observe that the proposed approach requires higher function evaluations as compared to BLEAQ at the upper level, but the requirements at the lower level are much less. The ratio of overall function evaluations, i.e. sum of upper and lower level function evaluations, shows a significant amount of saving for the proposed approach. The ratio representing the savings is provided in the last column of the tables IV and V. We also provide the convergence plots of the algorithm for the larger instances of SMD1 and SMD2. Figures 4 and 6 provide the convergence plots with random initialization (without initialization in relaxed feasible region) and local search. Figures 5 and 7 provide the convergence plots with random initialization (without initialization in relaxed feasible region) and no local search. The figures clearly demonstrate the importance of performing ϵ-KKT based local search in the context of our evolutionary algorithm.

VI. Conclusions

One of the common approaches for solving bilevel problems is its reduction to a single level problem using KKT conditions. However, the KKT conditions introduce a highly constrained search region that makes the optimization task difficult. In this paper, we reduce the bilevel problem to single level using KKT proximity measure. We embed this idea in an evolutionary approach as a local search and test its efficacy in handling a variety of bilevel optimization problems. A comprehensive comparative study has been performed against BLEAQ on a number of test problems. The method leads to an overall savings in function evaluations for most of the test problems studied in this paper. The results suggest that a single level reduction using the KKT proximity measure might be a viable reduction technique for bilevel programs and can be utilized in the existing solution methods.

References

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

[2] J. F. Bard. *Practical Bilevel Optimization: Algorithms and Applications*. Kluwer Academic Publishers, 1998.

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

[4] 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.

[5] 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.

[6] 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.

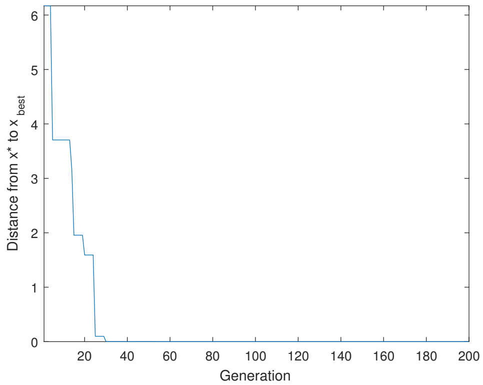

Fig. 4. Euclidean distance of the best solution at every generation from bilevel optimum for 10-dimensional SMD1 test problem when run with local search.

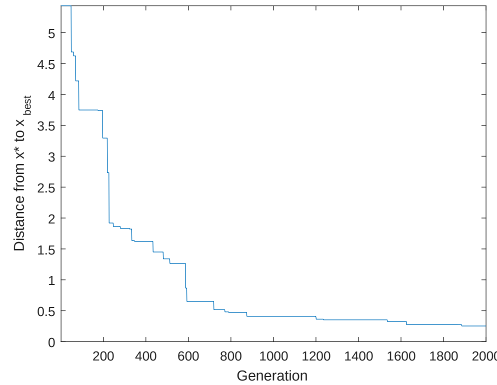

Fig. 5. Euclidean distance of the best solution at every generation from bilevel optimum for 10-dimensional SMD1 test problem when run without local search.

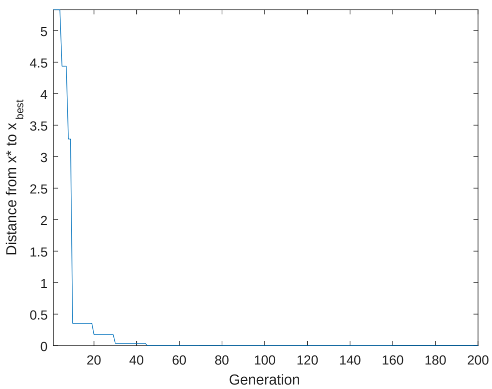

Fig. 6. Euclidean distance of the best solution at every generation from bilevel optimum for 10-dimensional SMD2 test problem when run with local search.

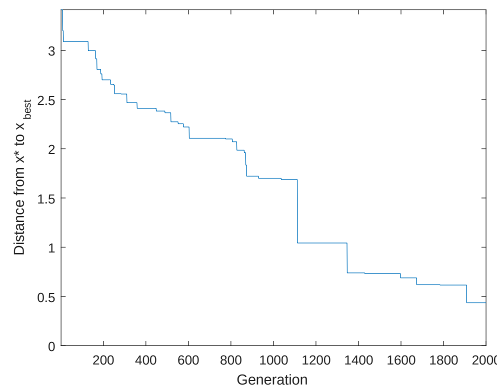

Fig. 7. Euclidean distance of the best solution at every generation from bilevel optimum for 10-dimensional SMD2 test problem when run without local search.

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

[8] Stephan Dempe. *Foundations of Bilevel Programming*. Kluwer Academic Publishers, Secaucus, NJ, USA, 2002.

[9] Joydeep Dutta, Kalyanmoy Deb, Rupesh Tulshyan, and Ramnik Arora. Approximate kkt points and a proximity measure for termination. *Journal of Global Optimization*, 56(4):1463–1499, 2013.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

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

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

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

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

[19] 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.

[20] A. Sinha, P. Malo, and K. Deb. Test problem construction for single-objective bilevel optimization. *Evolutionary Computation Journal*, 22(3):439–477, 2014.

[21] 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.

[22] A. Sinha, P. Malo, and K. Deb. Solving optimistic bilevel programs by iteratively approximating lower level optimal value function. In *2016 IEEE Congress on Evolutionary Computation (CEC-2016)*. IEEE Press, 2016.

[23] A. Sinha, P. Malo, and K. Deb. Approximated set-valued mapping approach for handling multiobjective bilevel problems. *Computers and Operations Research*, 77:194–209, 2017.

[24] A. Sinha, P. Malo, and K. Deb. Evolutionary algorithm for bilevel optimization using approximations of the lower level optimal solution mapping. *European Journal of Operational Research*, 257(2):395–411, 2017.

[25] A. Sinha, P. Malo, K. Deb, P. Korhonen, and J. Wallenius. Solving bilevel multi-criterion optimization problems with lower level decision uncertainty. *IEEE Transactions on Evolutionary Computation*, 20(2):199–217, 2016.

[26] 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.

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

[28] 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.

[29] 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.

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