Bilevel Optimization Without Tears

Jesús Mejía and Efrén Mezura-Montes
Artificial Intelligence Research Center
University of Veracruz, MEXICO
Introduction

Optimization and origin of Bilevel Optimization (BO)

Methods from Classical to Metaheuristics

Solution Strategies

Applications

Conclusions

Final comments

Outline

Practical examples to understand the formal definition and its properties

BO for solving interest problems
Introduction
Optimization

101

Optimization

101

Minimization

“Find a best element (with regard to some criterion) from a set of available alternatives”

$$x^* \in \arg \min_{x \in X} f(x)$$

Best element
Criterion
Set of alternatives
Optimization

101

Minimization

“Find an argument that minimizes an objective function”

\[ x^* \in \arg \min_{x \in X} f(x) \]

Optimum

Objective function

Search space
Optimization

\[ x^* \in \arg \min_{x \in X} f(x) \]
Optimization

Example

\[ f(x) = x_1^2 + x_2^2 \quad X = [-1, 1]^2 \]

\[ x^* = (0, 0) \]
Optimization

Homework

Minimize

\[ f(x) = \sum_{i=1}^{D} (x_i - y)^2 \]

where \( X = [-1, 1]^D \)

and \( y \) is a constant equal to 1
Optimization

Homework

Minimize

\[ f(x) = \sum_{i=1}^{D} (x_i - y)^2 \]

where \( X = [-1, 1]^D \)

and \( y \) is a constant equal to 1

Solution:

\[ x_i = y, \ 1 \leq i \leq D \]
Origin

Of Bilevel Optimization

- 1934 A Leader-follower game was introduced
- 1973 Formulation for bilevel optimization
- 1977 Introduce term “Bilevel Programming”
- 1988 BO problems are NP-hard!
- 1992 BO problems are strongly NP-hard!
- 1994 First Genetic Algorithm for BO
- 2000-Present High interest from EC community

Bilevel Optimization
Bilevel Optimization

Wrong ideas on BO

“I am expert in Multi-objective optimization and my trusted friend told me that BO is a bi-objective optimization”

“Bilevel optimization is like optimizing 2D functions”

“Nobody needs BO”
Bilevel Optimization

Description

“An upper level authority takes a decision subject to an optimal response from a lower level authority”

Bilevel Optimization

Dummy example

“A guy finds the best way to escape, subject to the optimal path (to the guy’s position) planned by the chickens”
Bilevel Optimization

Dummy example

“*A guy finds the best way to escape, subject to the optimal path (to the guy’s position) planned by the chickens*”

\[ x^* \in \arg \min_{x \in X} F(x, y^*) \]

Best leader solution

Objective function

Best follower solution

Possible solutions
Bilevel Optimization

Dummy example

“A guy finds the best way to escape, subject to the optimal path (to the guy’s position) planned by the chickens”

\[ y^* \in \arg\min_{y \in Y} f(x, y) \]

- Best follower solution
- Objective function
- Current leader solution
- Possible solutions
Bilevel Optimization: Nested Scheme

1. **Leader** takes a decision $x$
2. The **follower** uses the leader's decision to take the best decision based on $f$
3. The **leader** evaluates both $x, y$ to evaluate $F$

Bilevel Optimization: Formal Definition

Minimize:

\[ F(x, y^*), \quad x \in X \]

Subject to:

\[ y^* \in \text{arg min}_{y \in Y} f(x, y) \]
Bilevel Optimization: Solutions

$$(x, y^*)$$
Feasible solutions

$x \in X$

$$(x^*, y^*)$$
Optimal Feasible solutions

$x^* \in \arg \min_{x \in X, y^* \in \Psi(x)} F(x, y^*)$

$y^* \in \Psi(x) = \arg \min_{y \in Y} f(x, y)$

Bilevel Optimization: Practical Example

Minimize:

\[ F(x, y) = x^2 + y^2 \]

Subject to:

\[ y \in \arg \min_{y \in Y} f(x, y) = (x - y)^2 \]

\[ X = Y = [-5, 5] \]
Bilevel Optimization: Practical Example

Step 1: Choose $x \in X$ said $x = a$

Step 2: Solve:

$$y \in \arg\min_{y \in Y} f(a, y) = (a - y)^2$$

$$f(a, y) \geq 0 \Rightarrow \min_y f(a, y) = 0 \Rightarrow y = a$$

$$\Psi(x) = \{a\}$$

Step 3: Evaluate: $F(a, y^*) = F(a, a) = 2a^2$
Bilevel Optimization: Practical Example

**UL & LL functions**

\[ F(x, y) = x^2 + y^2 \]
\[ f(x, y) = (x - y)^2 \]

**Feasible solutions**

\( (x, x), \ -5 \leq x \leq 5 \)

**Optimum solution**

\( (x^*, y^*) = (0, 0) \)

Leader performance
s.t. best follower decision

Best follower decision
Bilevel Optimization

Practical Example

Leader takes a decision

Follower takes the best decision

True leader performance

\[ F(x, x) = 2x^2 \]

\[ F(1, 1) = 2 \]

\[ f(1, y) = (1 - y)^2 \]
Bilevel Optimization: General Definition

Minimize:

\[ F(x, y^*) \text{, } x \in X \]

Subject to:

\[ y^* \in \arg \min_{y \in Y} \{ f(x, y) : g_i(x, y) \leq 0, i = 1, 2, \ldots, I \} \]

\[ G_j(x, y) \leq 0, j = 1, 2, \ldots, J \]

Solution Strategies
Solution Strategies

Classical

- Mathematical programming
  - Karush-Kuhn-Tucker conditions for single-level reduction
  - Branch and Bound
  - Trust region
  - Among others

Approximate

- Metaheuristics
  - Population-based algorithms have been successfully used.
    - Evolutionary Algorithms
    - Swarm Intelligence
    - Among others

Hybrid

- Metaheuristics + Mathematical programming
  - Usually, Karush-Kuhn-Tucker conditions are used in addition to population based algorithms for global optimization

Solution Strategies

**Classical**

**Mathematical programming**
- Karush-Kuhn-Tucker conditions for single-level reduction
- Branch and Bound
- Trust region
- Among others

Karush-Kuhn-Tucker conditions

\[
\begin{align*}
\min_{x \in Y, y \in Y, \lambda} & \quad F(x, y) \\
\text{Subject to:} & \\
G_j(x, y) & \leq 0, j = 1, 2, \ldots, J \\
g_i(x, y) & \leq 0, i = 1, 2, \ldots, I \\
\lambda_i g_i(x, y) & = 0, i = 1, 2, \ldots, I \\
\nabla_y L(x, y, \lambda) & = 0 \\
\lambda_i & \leq 0, i = 1, 2, \ldots, I
\end{align*}
\]

\[
L(x, y, \lambda) = f(x, y) + \sum_{i=1}^{I} \lambda_i g_i(x, y)
\]
Solution Strategies

Metaheuristics

Population-based algorithms have been successfully used.

- Evolutionary Algorithms
- Swarm Intelligence
- Among others

The metaheuristics use the nested scheme to sequentially optimize both levels.

Solution Strategies

Approximate

Metaheuristics

Population-based algorithms have been successfully used.

- Evolutionary Algorithms
- Swarm Intelligence
- Among others

## Solution Strategies

### Metaheuristics

Population-based algorithms have been successfully used.

- Evolutionary Algorithms
- Swarm Intelligence
- Among others

<table>
<thead>
<tr>
<th>Approximate</th>
<th>Paper</th>
<th>Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BLDES [2]</td>
<td>DE DE</td>
</tr>
<tr>
<td></td>
<td>$\varepsilon$-KKT [38]</td>
<td>GA GA/SQP</td>
</tr>
<tr>
<td></td>
<td>BLEG0 [15]</td>
<td>DE/EGO EGO</td>
</tr>
<tr>
<td></td>
<td>BLMA [16]</td>
<td>DE DE</td>
</tr>
<tr>
<td></td>
<td>SABLA [14]</td>
<td>DE-IP DE-IP</td>
</tr>
<tr>
<td></td>
<td>BLEAQ(-II) [32, 34]</td>
<td>GA GA/SQP</td>
</tr>
<tr>
<td></td>
<td>Surrogate-assisted BIDE [1]</td>
<td>DE DE</td>
</tr>
</tbody>
</table>

Differential Evolution (DE), Genetic Algorithm (GA), Sequential Quadratic Programming (SQP), Interior Point Method (IP), Efficient Global Optimization (EGO).

Solution Strategies

Simple (naive) solution in Julia

using Metaheuristics

\[ F(x, y) = \sum (x^2 + y^2) \]
\[ f(x, y) = \sum ((x - y)^2) \]
\[ \text{bounds} = [-5 -5 -5; 5 5 5.0] \]
\[ \psi(x) = \text{minimizer}(\text{optimize}(y \rightarrow f(x, y), \text{bounds})) \]
\[ \text{minimizer}(\text{optimize}(x \rightarrow F(x, \psi(x)), \text{bounds})) \]

Solution found in 4.62 seconds.

- Float64[-0.0016458, -0.000558254, 0.0014492]
- minimizer(res)
- 5.601127187667976e-6
- minimum(res)

Try it yourself online for free: bi-level.org/tutorial
Applications
Applications

Toll Setting Problem

**Upper level** authority that wants to optimize the tolls revenue for a network of roads.

**Followers** are the network users that want to optimize their objectives (costs, time, etc).

Applications

Automated Parameter Tuning

**Leader:** Chooses the parameters

**Follower:** Finds the hardest instances for the target algorithm

Working paper.
Applications

Principal-Agent Problems

Leader (principal) subcontracts a job to an agent (follower). Uses an incentive scheme that aligns the interests of the agent with the principal.

Follower: (agent) prefers to act in his own interests rather than those of the leader.

Conclusions
Conclusions

1. Bilevel Optimization (BO) is useful to model problem with inherent hierarchical structure
2. BO offers rich properties for solving problems
3. Approximate methods can successfully solve real-world problems
4. More theoretical studies are needed
5. BO problems can be hard to solve because of the computational complexity
Thank you!

Questions?

Download this tutorial from:
bi-level.org/tutorial