Multi-Objective Optimization

Faheem Zafari, Jian Li

Imperial College London, University of Massachusetts Amherst

faheem16@imperial.ac.uk, jianli@cs.umass.edu

February 1, 2018

\[1\]

Overview

1. Multi-objective Optimization (MOO)
   - Basic Concepts

2. MOO Solution Techniques
   - Scalarization-based MOO Formulation
   - Meta-heuristics
   - Hybrid Meta-heuristics

3. Game theory-based MOO Design Techniques
   - Cooperative Game Theory
For $m$ inequality constraints and $p$ equality constraints, MOO identifies a vector $\mathbf{x}^* = [x_1^*, x_2^*, \cdots, x_n^*]^T$ that optimizes a vector function

$$\bar{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_k(\mathbf{x})]^T$$

such that

$$g_i(\mathbf{x}) \geq 0, \quad i = 1, 2, \cdots, m,$$

$$h_i(\mathbf{x}) = 0, \quad i = 1, 2, \cdots, p,$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$ is a vector of $n$ decision variables and the feasible set is denoted by $F$. 
Pareto Optimality

**Strongly Pareto non-dominated solution**

A feasible solution $x$ is strongly Pareto non-dominated if there is no $y \in F$ such that $f_i(y) \leq f_i(x) \ \forall \ i = 1, 2, \ldots, k$ and $f_i(y) < f_i(x)$ for at least one $i$.\(^a\)

\(^a\)Means that there is no other feasible solution that can improve some objectives without worsening at least one other objective.

**Weakly Pareto non-dominated solution**

A feasible solution $x$ is weakly Pareto non-dominated if there is no $y \in F$ such that $f_i(y) < f_i(x) \ \forall \ i = 1, 2, \ldots, k$. 
Pareto Optimality

Pareto Efficiency/Optimality
A state in which resources cannot be reallocated to make any individual gain more without hurting any other objective.

Pareto Improvement/Pareto Dominated Solution
Given an initial allocation, if we can achieve a different allocation making at least one individual function better without hurting any other, then the starting state is called *Pareto improvement*. 
Pareto Optimality

For any minimization problem, a point $x^*$ is *Pareto Optimal* if the following holds for every $x \in F$

$$
\overline{f}(x^*) \leq \overline{f}(x)
$$

(3)

where $\overline{f}(x) = [f_1(x), f_2(x), \cdots, f_k(x)]^T$ and $\overline{f}(x^*) = [f_1(x^*), f_2(x^*), \cdots, f_k(x^*)]^T$.

Strong Pareto Optimality

A feasible solution $x$ is strongly Pareto optimal if it is strongly Pareto non-dominated.

Weak Pareto Optimality

A feasible solution $x$ is weakly Pareto optimal if it is weakly Pareto non-dominated.
Figure: MOO with two objectives (taken from\textsuperscript{2})

Scalarization-based MOO Formulation

### Weighted Sum

Weighted linear combination of the objective functions

\[
\bar{f}(x) = \sum_{i=1}^{k} r_i f_i(x)
\]  

(4)

where \(0 \leq r_i \leq 1, i = \{1, \cdots, k\}, \sum_{i=1}^{k} r_i = 1\).

### Utility Function Method

\[
\max U(\bar{f}(x)) \\
\text{s.t.} \quad [f_1(x), \cdots, f_k(x)] \leq z^* \\
[g_1(x), \cdots, g_m(x)] \geq 0, x \in S
\]  

(5)

where \(\bar{f}(x) = [f_1(x), \cdots, f_k(x)]^T\), and \(z^*\) is a vector of reference points.
Scalarization-based MOO Formulation

Goal programming

For a target goal $g_i$ set by the decision maker (DM), the goal is to

$$
\min \sum_{i=0}^{k} |f_i(x) - g_i|.
$$

(6)

Min-Max Method

$$
\min \left[ \max Z_i(x) \right] \quad \forall i = 1, \cdots, k
$$

(7)

where

$$
Z_i(x) = \frac{|f_i(x) - g(i)|}{g(i)} \quad \forall i = 1, \cdots, k.
$$

(8)

There are a number of different methods as well details of which can be found in [Cho et al., 2017]
## Table: Comparison of Scalarization-based MOO Formulation (taken from [Cho et al., 2017])

<table>
<thead>
<tr>
<th>Technique</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weighted Sum</td>
<td>Computationally efficient particularly for strongly non-dominated solution</td>
<td>Weight assignment, concave trade-off curve</td>
</tr>
<tr>
<td>Utility Function</td>
<td>Useful with game theory for designing MOO problems for resource allocation</td>
<td>Hard to have a global view (for an agent) in a distributed system</td>
</tr>
<tr>
<td>Goal Programming</td>
<td>Computationally efficient if feasible solution space is found</td>
<td>Computationally inefficient if feasible solution space is not found</td>
</tr>
<tr>
<td>Min-Max</td>
<td>Provides the best possible optimal solution if all the objectives have equal priorities</td>
<td>Computationally inefficient if feasible solution space is not found</td>
</tr>
</tbody>
</table>
Can’t provide optimality guarantee

Evolutionary Algorithms (EA)
- An evolutionary algorithm involves the process of *recombination* (crossover), *variation* (mutation), and *natural selection*.

Ant Colony Optimization (ACO)
- Requires formulating the problem as the best path finding problem on a weighted graph.
- Inspired from how different ants in a colony cooperate to obtain food.
Simulated Annealing (SA)

- Probabilistic technique for identifying the global minimum of a cost function that can multiple local minima.
- The goal is to obtain solutions by decreasing the probability of accepting worse solutions slowly.

Variable Neighborhood Search (VNS)

- Uses the distance between a current solution and its neighborhood representing local optimal leading to an improved solution.
- It is based on the idea that neighborhoods change both in descent to local optima and in escape from valleys that contain local optima.
## Meta-heuristics

### Table: Summary of Meta-heuristics

<table>
<thead>
<tr>
<th>Technique</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>EA</td>
<td>Provides heuristic, but close-to-optimal solutions</td>
<td>Computationally expensive, usually generate local optima</td>
</tr>
<tr>
<td>ACO</td>
<td>Suitable for dynamic applications (like ours)</td>
<td>Solution convergence time is not predictable</td>
</tr>
<tr>
<td>SA</td>
<td>Good approximation solution for a large size solution search space</td>
<td>No global optimality guarantee, can be computationally expensive</td>
</tr>
<tr>
<td>VNS</td>
<td>Provides efficient and good approximation solutions</td>
<td>For large constraint set, can be computationally expensive</td>
</tr>
</tbody>
</table>
# Hybrid Meta-heuristics

## Table: Summary of Hybrid Meta-heuristics

<table>
<thead>
<tr>
<th>Technique</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>EA + Dynamic Programming</td>
<td>Produces efficient feasible solution</td>
<td>May reduce solution diversity</td>
</tr>
<tr>
<td>ACO + Constraint Programming</td>
<td>Generates efficient but good quality solutions by leveraging the benefits of using CP</td>
<td>Update of global constraints incurs extra overhead</td>
</tr>
<tr>
<td>SA + Tabu Search</td>
<td>Controls worsening solutions using SA’s temperature parameter</td>
<td>Computationally expensive for problems with few local optima</td>
</tr>
<tr>
<td>VNS + Large Neighborhood Search</td>
<td>Provides good quality neighborhood search region</td>
<td>Finding LNS using exact algorithms is NP-hard</td>
</tr>
</tbody>
</table>
Cooperative Game (CG) Theory

- A group of players/users known as *Coalitions* cooperate to enhance their utilities/benefits by joining a grand coalition.
- The game is played by the coalition of players rather than the players in each coalition.
- CG is denoted by the pair \((N, v)\) where \(N = \{1, 2, \cdots, n\}\) is the set of players and \(v\) computes the value obtained from subset \(S\) of \(N\)
  - \(v(S)\) is the value of forming a coalition consisting of all the players in \(S\)
  - \(v\) captures the objective of the system.
Cooperative Game Theory

Non-transferable Utility (NTU)

- CG’s are mostly referred to as NTU games.
- Denoted by a pair \((N, V)\)
  - \(N\) is the set of players and \(V\) is the function assigning payoff to each coalition \(S \subset N\)

Hedonic Game

- Special case of NTU in which no externalities are considered.
  - The members of the coalition are only affected by other members of the same coalition

---

\(^3\)NTU means that the agents do not have a common scale to measure the payoff for a coalition
Cooperative Game Theory

Nash Bargaining Solution (NBS)

In CGT, NBS is applied when two or more players are required to select one of possible outcomes from any joint collaboration.

- Particularly when two parties negotiate something associated with each party’s interest, a bargaining game may result in a disagreement outcome, i.e., a payoff each player receives when a negotiation is not successful.

Shapley Value

- Indicates how valuable a player is to the overall cooperation and the payoff player can expect from joining the coalition.

\[
\phi_i(\nu) = \sum_{S \subseteq N \setminus i} \frac{|S|!(n - |S| - 1)}{n!} (\nu(S \cup \{i\}) - \nu(S))
\]  

(9)
## Table: Summary of CGT

<table>
<thead>
<tr>
<th>Technique</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>NTU</td>
<td>Generic</td>
<td>Generally no guarantee of a unique solution</td>
</tr>
<tr>
<td>Hedonic Games</td>
<td>Generic</td>
<td>Requires additional conditions to ensure stable partitioning for different presentations</td>
</tr>
<tr>
<td>Shapley Value</td>
<td>Simple to measure utility for a coalition</td>
<td>High communication overhead, no guarantee in hostile environments</td>
</tr>
<tr>
<td>NBS</td>
<td>Generic with less complexity</td>
<td>Not straightforward for a cooperative concept</td>
</tr>
</tbody>
</table>