A Priori Methods in Multiobjective Optimization

Markus Hartikainen, PhD

Department of Mathematical Information Technology,
University of Jyväskylä,
Finland
Table of Contents

1. Recap on the Basics of Multiobjective Optimization
2. No Preference Methods
3. From No Preference Methods to a Priori Methods
4. General Properties of a Priori Methods
5. Examples of a Priori Methods
6. Augmentation Term
7. Critical Evaluation of a Priori Methods
8. From a Priori Methods to a Posteriori Methods
A general formulation for multiobjective optimization problems is

\[
\min \quad f(x) := (f_1(x), \ldots, f_k(x))^T \\
\text{s.t.} \quad x \in S,
\]

where \( S \subset \mathbb{R}^n \) is called the feasible set.

A feasible solution \( x \in S \) is called **Pareto optimal** (PO), if there does not exist another solution \( y \in S \) such that

1. \( f_i(y) \leq f_i(x) \) for all \( i = 1, \ldots, k \) and
2. \( f_j(y) < f_j(x) \) for some \( j \in \{1, \ldots, k\} \).

or **weakly Pareto optimal** (WPO), if there does not exist another solution \( y \in S \) such that \( f_i(y) < f_i(x) \) for all \( i = 1, \ldots, k \).

The set of objective function vectors given the Pareto optimal solutions is called the Pareto front (PF).
More Important Definitions in Multiobjective Optimization

- The ideal vector (or point) $z^{\text{ideal}} \in \mathbb{R}^k$ has
  \[ z^{\text{ideal}}_i = \min_{x \in S} f_i(x) \text{ for all } i = 1, \ldots, k. \]

- The nadir vector (or point) $z^{\text{nadir}} \in \mathbb{R}^k$ has
  \[ z^{\text{nadir}}_i = \max_{x \in S \text{ is PO}} f_i(x) \text{ for all } i = 1, \ldots, k. \]

- A decision maker (DM) is a person that can give further preference information concerning the PO solutions.
Illustration of the Important Definitions in Multiobjective Optimization
Often multiobjective optimization methods are classified w.r.t. to the role of a decision maker

- **No preference methods** are methods where the DM is not needed

- **A priori methods** are methods where the DM articulates preferences before optimization

- **A posteriori methods** aim to generate a representative set of PO solutions and the DM chooses the best one among them

- **Interactive methods** allow the DM to guide the search by alternating optimization and preference articulation iteratively
No Preference Methods

- Usually used only when the decision maker is not available e.g., online optimization
- Only one solution is computed
- E.g., the so-called method of global criterion where an optimization problem of the form

\[
\min \| f(x) - z_{\text{ideal}} \|_p \\
\text{s.t. } x \in S
\]

is solved, where \( \| \cdot \|_p \) is any \( L^p \) norm
In a priori methods:

1. The preferences of the decision maker are asked and the best solution according to the given preferences is found.

   + The DM can tell what kind of solution he wants
   - The DM has to devote some time to giving the preferences
   - The DM must understand the type of preference information
   - The DM must have some kind of (a priori) understanding about at least one of the following:
     - His own preferences
     - The interdependencies of the objectives or
     - The feasible objective values
A priori methods differ from each other in two main aspects:

1. How the preferences are asked?
2. How (achieving) these preferences is modeled?

The preferences can be asked e.g., as reservation or aspiration levels, or as weights representing relative importance of the objectives.

Achieving the given aspiration levels can be modeled as e.g., minimizing distance between the vector containing the aspiration levels and the solution found.
Desirable Properties of a Priori Methods

1. There is an easily understandable way of getting the preference information from the DM
2. The DM’s preferences can be accurately modeled in the chosen way
3. For all possible preferences, the solution found is PO
4. Each PO solution can be found with some preference
Desirable Properties of a Priori Methods

1. There is an easily understandable way of getting the preference information from the DM
2. The DM’s preferences can be accurately modeled in the chosen way
3. For all possible preferences, the solution found is PO
4. Each PO solution can be found with some preference

!!UNFORTUNATELY, THERE DOES NOT EXIST A METHOD THAT WOULD FULFIL ALL THE PROPERTIES!!
The $\epsilon$-Constraint Method

- Means optimizing problem

\[
\begin{align*}
\min & \quad f_i(x) \\
\text{s.t.} & \quad x \in S, \\
& \quad f_j(x) \leq \epsilon_j \text{ for all } j \in \{1, \ldots, k\} \setminus \{i\}.
\end{align*}
\]  

(1)

- Above, the $\epsilon_j \in \mathbb{R}$ are bounds for objectives and $f_i$ is the objective to be minimized, all chosen by the DM

- Objective bound $\epsilon_j$ means that the DM wants a solution with $f_j(x) \leq \epsilon_j$
Properties of the $\epsilon$-Constraint Method

- For any $\epsilon \in \mathbb{R}^{k-1}$, an optimal solution to (1) (if exists) is WPO.
- For any $\epsilon \in \mathbb{R}^{k-1}$, the unique optimal solution to (1) (if exists) is PO.
- Let $x \in S$ be a PO solution and let $i \in \{1, \ldots, k\}$. Then by choosing $\epsilon_j = f_j(x)$ the solution $x$ is an optimal solution to (1).
The Achievement Scalarizing Function

- A method developed by Wierzbicki in 1986, where one optimizes the problem

\[
\begin{align*}
\min & \quad \max_{i=1,\ldots,k} \frac{f_i(x) - z_{asp}^i}{z_{nadj}^i - z_{ideal}^i} \\
\text{s.t.} & \quad x \in S.
\end{align*}
\]

- Above, the \( z_{asp} \) is a vector containing the aspiration levels of the decision maker.
- Aspiration level \( z_{asp}^i \) means that the decision maker would like to have a solution with \( f_i(x) = z_{asp}^i \).
For any $z^{asp} \in \mathbb{R}^k$, an optimal solution to (2) is WPO.

For any $z^{asp} \in \mathbb{R}^k$, the unique optimal solution to (2) is PO.

Let $x \in S$ be a PO solution. Then by choosing $z^{asp} = f(x)$ the solution $x$ is an optimal solution to (2).
The Weighted Sum

- Means optimizing problem

\[
\min \sum_{i=1,\ldots,k} w_i f_i(x) \\
\text{s.t. } x \in S.
\] (3)

Above, the \( w_i \geq 0 \) are weights given by the decision maker representing the relative importance of the objectives.

- If weight \( w_i \) is bigger than \( w_j \) it means that the decision maker appreciates improvement on objective \( f_i \) more than on objective \( f_j \).
Properties of the Weighted Sum

- If $w_i \geq 0$ for all $i = 1, \ldots, k$ and $w_j > 0$ for some $j = 1, \ldots, k$, then an optimal solution to (3) is WPO.
- If $w_i > 0$ for all $i = 1, \ldots, k$, then an optimal solution to (3) is PO.
- However, there may be PO solutions that do not optimize (3) for any $w \in \mathbb{R}_+^k$!
There exist multiple methods for eliciting the weights in weighted method
- Swing Weighting
- Lotteries and expected utilities
- etc.

Especially, in discrete choice situations these are used a lot
However, no matter what method for eliciting the weights is used, it does not remove the problems of the weighting method
In these lectures, we are not going to deal with these
Augmentation Term

- Many a priori methods e.g., the above mentioned $\epsilon$-constraint method and the Achievement scalarizing function guarantee only that (possibly not unique) solutions are merely WPO.

- To guarantee PO solutions, so-called augmentation term

\[ \rho \sum_{i=1}^{k} f_i(x) \] (where $\rho > 0$ is a small constant)

is often added to the objective functions of a priori methods.

- More often than not, augmented versions are used in practice instead of the original versions.

- Also, other approaches for removing the WPO solutions exist.
Level Curves of the $\epsilon$-Constraint Method with and without the Augmentation Term
Common Problems in all a Priori Methods

1. The decision maker may not know what is feasible and what is not.
2. The decision maker may not understand how the elicited preferences affect the solutions generated.
3. Potentially better solutions may be missed as there is no feedback to the given preferences.
Also: the Weighted Sum May Produce Unintuitive Solutions

Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights) | 0.4 | 0.2 | 0.2 | 0.2 |

Two big problems emerge:

1. The winner is the worst one in the most important criterion!
2. Intuitively appealing compromise Johanna will not be chosen with any given weights!
Also: the Weighted Sum May Produce Unintuitive Solutions

Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights) | 0.4 | 0.2 | 0.2 | 0.2 |

Two big problems emerge:
- The winner is the worst one in the most important criterion!
- Intuitively appealing compromise Johanna will not be chosen with any given weights!
Also: the Weighted Sum May Produce Unintuitive Solutions

Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights) | 0.4 | 0.2 | 0.2 | 0.2 |

Two big problems emerge:

- The winner is the worst one in the most important criterion!
- Intuitively appealing compromise Johanna will not be chosen with any given weights!
Also: the Weighted Sum May Produce Unintuitive Solutions

Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights) | 0.4 | 0.2 | 0.2 | 0.2 |

Two big problems emerge:

- The winner is the worst one in the most important criterion!
- Intuitively appealing compromise Johanna will not be chosen with any given weights!
Also: the Weighted Sum May Produce Unintuitive Solutions

Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights) |

- 0.4 | 0.2 | 0.2 | 0.2 |

Two big problems emerge:
- The winner is the worst one in the most important criterion!
- Intuitively appealing compromise Johanna will not be chosen with any given weights!
Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights)

- 0.4 0.2 0.2 0.2

Two big problems emerge:
- The winner is the worst one in the most important criterion!
- Intuitively appealing compromise Johanna will not be chosen with any given weights!
Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights) | 0.4 | 0.2 | 0.2 | 0.2 |

Two big problems emerge:

1. The winner is the worst one in the most important criterion!
2. Intuitively appealing compromise Johanna will not be chosen with any given weights!
Also: the Weighted Sum May Produce Unintuitive Solutions

Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
</tbody>
</table>

Relative importances (weights)

- Two big problems emerge:
  1. The winner is the worst one in the most important criterion!
  2. Intuitively appealing compromise Johanna will not be chosen with any given weights!
Assume choosing a wife from the following candidates:

<table>
<thead>
<tr>
<th></th>
<th>Appearance</th>
<th>Cooking</th>
<th>House keeping</th>
<th>Cleanliness</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martta</td>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>6.4</td>
</tr>
<tr>
<td>Johanna</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Nina</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4.6</td>
</tr>
<tr>
<td>Relative importances (weights)</td>
<td>0.4</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td></td>
</tr>
</tbody>
</table>

- Two big problems emerge:
  - The winner is the worst one in the most important criterion!
  - Intuitively appealing compromise Johanna will not be chosen with any given weights!
In a posteriori methods, a set of solutions is generated and the decision maker may choose the best one from that.

**Pluses:**
- They allow the decision maker to see what is feasible
- In theory, they guarantee that no superior solutions will be missed

**Minuses:**
- More computation is needed
- It may be hard for the decision maker to choose from a large list