Classic Methods

for Multi-Objective Optimization

Giuseppe Narzisi

Courant Institute of Mathematical Sciences
New York University

31 January 2008

Outline

Introduction

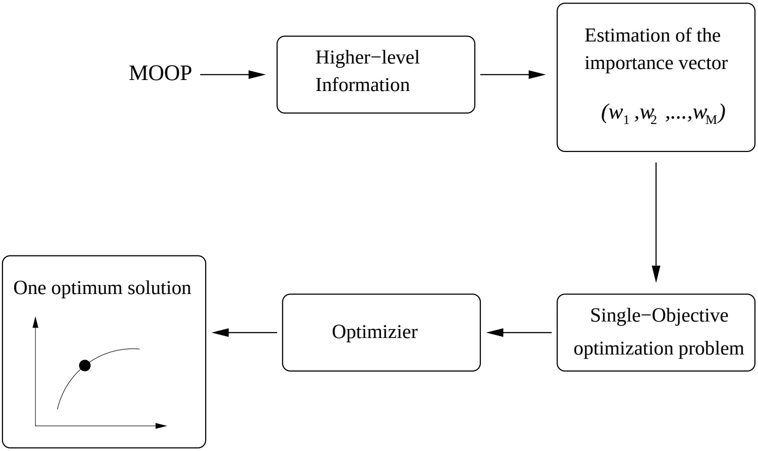
- We will talk about classical methods for handling multi-objective optimization problems.
- We refer to this methods as *classical* to distinguish them from the evolutionary methods (which we will cover later).

Two approaches to Multi-Objective Optimization

1. PREFERENCE-BASED PROCEDURE:
   - Composite objective function as the weighted sum of the objectives
     - Only useful if a relative preference factor of the objectives is known in advance.
2. IDEAL PROCEDURE:
   1. Find multiple trade-off solutions with a wide range of values for the objectives.
   2. Choose one of the obtained solution using higher-level information

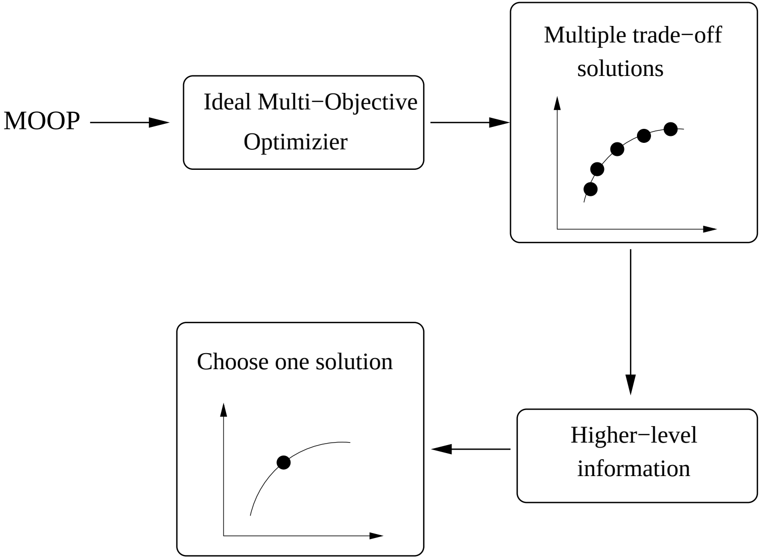
Preference-Based method

Illustration



Ideal Multi-Objective procedure

Illustration



Classification

of the classical methods

- Classical method have been around for the past four decades.
- They can be classified in the following classes:
  - *Non-preference methods*
    - These methods do not assume any information about the importance of objectives, but an heuristic is used to find a single optimum solution. They do not make any attempt to find multiple Pareto-optimal solutions.
  - *Posteriori methods*
    - Posteriori methods use preference information of each objective and iteratively generate a set of Pareto-optimal solutions.
  - *A priori methods*
    - On the other hand, a priori methods use more information about the preferences of objectives and they usually find one preferred Pareto-optimal solution.
  - *Interactive methods*
    - These methods use the preference information progressively during the optimization process.

Weighted Sum Method

- This method is the simplest approach and probably the most widely used classical method.
- This method scalarizes the set of objectives into a single objective by multiplying each objective with a user supplied weight.
- Although simple, it introduces a non-simple question: *What value of the weights must be used?* The answer depend on the relative importance of each objective.

Weighted Sum Method

formulation

$$
\begin{array}{ll}
\mathit{min} & \mathcal{F}(\mathbf{x}) = \sum_{m=1}^{M} w_m f_m(\mathbf{x}) \\
\mathit{subject\ to} & \mathcal{G}(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_J(\mathbf{x})] \geq 0 \\
 & \mathcal{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_K(\mathbf{x})] = 0 \\
 & x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, \ldots, N
\end{array}
\tag{1}
$$

- where the objectives are normalized.
- $w_m \in [0, 1]$ is the weight of the m-th objective function.
- It is usual practice to chose weights such that $\sum_{m=1}^{M} w_m = 1$.

Weighted Sum Method

Properties

Theorem

The solution to the problem presented in equation (1) is Pareto-optimal if the weight is positive for all objectives: $w_m > 0, m = 1, \ldots, M$

- The proof is achieved by contradiction considering a solution with all positive weights and that is not Pareto-optimal and showing that this brings to a contradiction.
- Note that the theorem does not imply that any Pareto-optimal solution can be obtained using a positive weight vector.

Weighted Sum Method

Properties

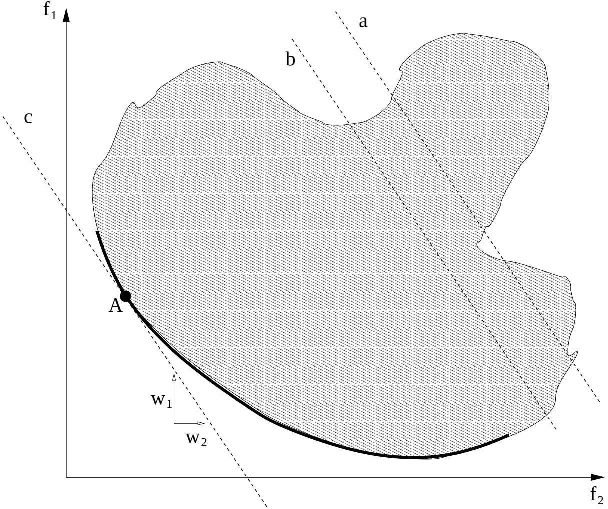
Theorem

if $\mathbf{x}^*$ *is a Pareto-optimal solution of a convex multi-objective optimization problem, then there exists a non-zero positive weight vector* $\mathbf{w}$ *such that* $\mathbf{x}^*$ *is a solution of problem (1)*

- The theorem suggests that for a convex MOOP any Pareto solution can be found using the weighted sum method (see Miettinen's book on *Nonlinear Multiobjective Optimization* [2] for the proof).

Weighted Sum Method

Illustration



Weighted Sum Method

Advantages and disadvantages

Advantages

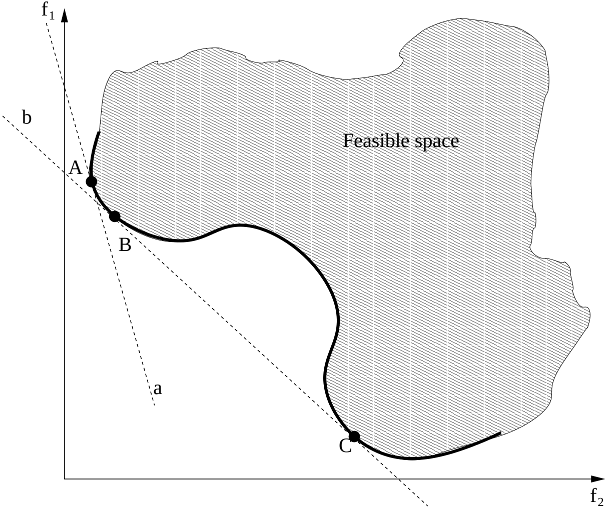
- Simple and easy to use.
- For convex problems it guarantees to find solutions on the entire Pareto-optimal set.

Disadvantages

- For mixed optimization problems (min-max), we need to convert all the objectives into one type.
- Uniformly distributed set of weights does not guarantee a uniformly distributed set of Pareto-optimal solutions.
- Two different set of weight vectors not necessarily lead to two different Pareto-optimal solutions.
- There may exists multiple minimum solutions for a specific weight vector that represents different solutions in the Pareto-optimal front (wasting the search effort).

Non-convex problems

Difficulties



ϵ-Constraint method

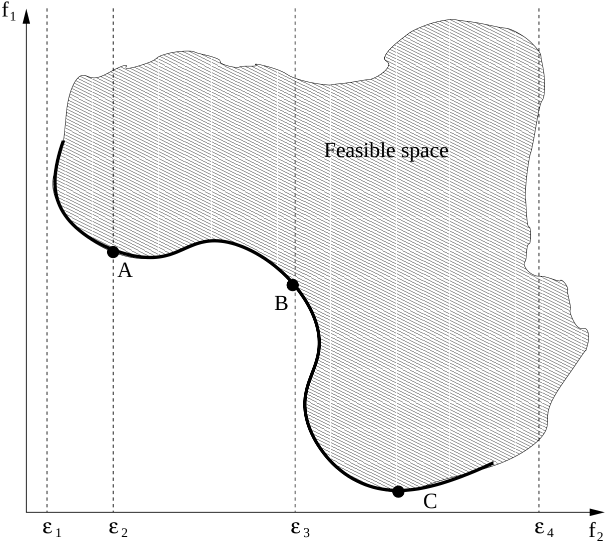
Definition

- **Idea**: keep only one of the objectives and restrict the rest of the objectives within some user-specified values.
- The modified problem is the following:

$$\begin{array}{ll} min & f_u(\mathbf{x}) \\ subject\ to & [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_{u-1}(\mathbf{x}), f_{u+1}(\mathbf{x}) \ldots, f_M(\mathbf{x})] \leq \epsilon \\ & \mathcal{G}(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_J(\mathbf{x})] \geq 0 \\ & \mathcal{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_K(\mathbf{x})] = 0 \\ & x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, \ldots, N \end{array} \tag{2}$$

ϵ-Constraint method

Illustration



ϵ-Constraint method

Properties

Theorem

The unique solution of the ϵ-constraint problem stated in (2) is Pareto-optimal for any given upper bound vector:
$\epsilon = (\epsilon_1, \ldots, \epsilon_{u-1}, \epsilon_{u+1}, \ldots, \epsilon_m)^T$

- The proof is again achieved by contradiction assuming that a unique solution of the ϵ-constraint problem is not Pareto-optimal and than showing that this assumption violates the definition of Pareto-optimality [2].

ϵ-Constraint method

Advantages and disadvantages

Advantages

- Different Pareto-optimal solution can be found using different ϵ_m values
- the method can be used also for non-convex multi-objective optimization problems

Disadvantages

- The solution to the problem largely depends on the selection of the ϵ vector. In particular, it must be chosen such that it lies between the minimum and maximum value of each objective function.
- As the number of objectives increase more information from the user is required (ϵ_m).

Weighted Metric Methods

Definition

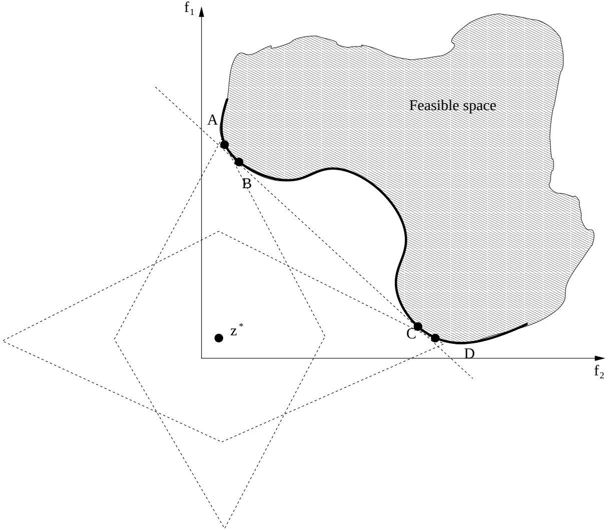
- **Idea**: instead of using a weighted sum of the objectives, we can consider other ways of combining multiple objectives.

$$
\begin{array}{ll}
\textit{min} & \mathcal{L}_p(\mathbf{x}) = \left(\sum_{m=1}^{M} w_m |f_m(\mathbf{x}) - \mathbf{z}_m^*|^p\right)^{\frac{1}{p}} \\
\textit{subject to} & \mathcal{G}(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_J(\mathbf{x})] \geq 0 \\
& \mathcal{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_K(\mathbf{x})] = 0 \\
& x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, \ldots, N
\end{array}
\tag{3}
$$

- Weights are non-negative.
- $p \in [1, \infty]$
- $\mathbf{z}^*$ is called reference point.
- When $p = 1$ is used, (3) is equivalent to the weighted sum method.

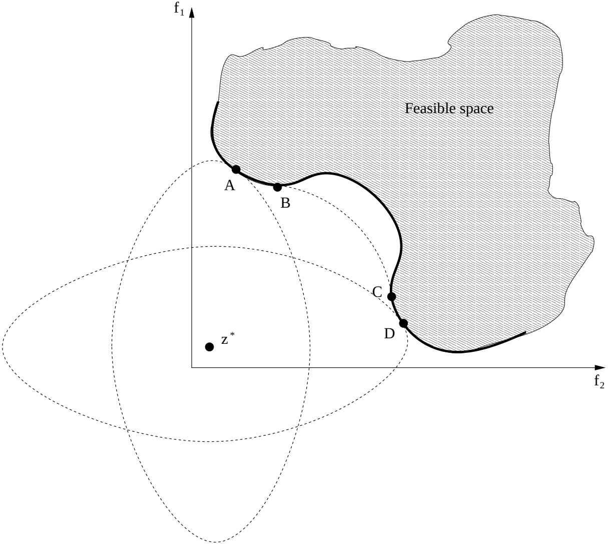
Weighted Metric Methods

($p = 1$) Taxicab or Manhattan norm



Weighted Metric Methods

($p = 2$) Euclidean norm



Tchebycheff problem

$(p = \infty)$

$$\begin{array}{ll} \mathit{min} & \mathcal{L}_\infty(\mathbf{x}) = max_{m=1}^{M} w_m |f_m(\mathbf{x}) - \mathbf{z}_m^*| \\ \mathit{subject\ to} & \mathcal{G}(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_J(\mathbf{x})] \geq 0 \\ & \mathcal{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_K(\mathbf{x})] = 0 \\ & x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, \ldots, N \end{array} \tag{4}$$

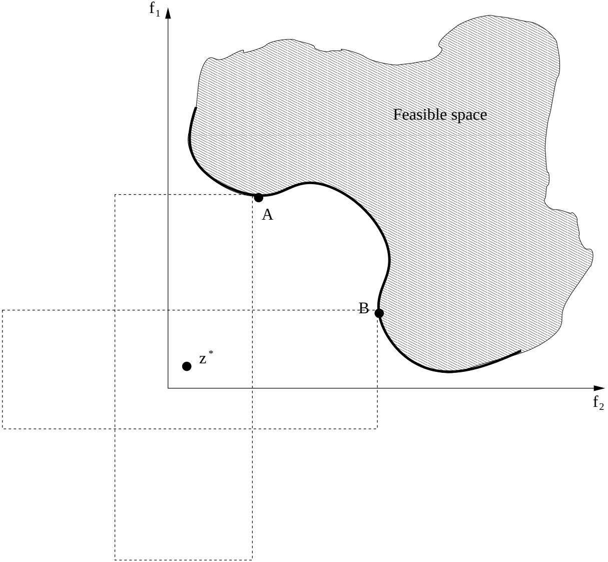
- If the Tchebycheff metric is used, any Pareto optimal solution can be found

Theorem

Let $\mathbf{x}$ *be a Pareto-optimal solution, then there exists a positive weighting* $\mathbf{w}$ *vector such that* $\mathbf{x}$ *is a solution of the weighted Tchebycheff problem (4) where the reference point is the utopian objective vector.*

Weighted Metric Methods

($p = \infty$)



Weighted Metric Methods

Advantages and disadvantages

Advantages

- The Tchebycheff metric allows to find each and every Pareto-optimal solution when $\mathbf{z}^*$ is the utopian objective vector [2].

Disadvantages

- It is advisable to normalize the objective functions, which requires knowledge of the minimum and maximum values of each objective.
- It requires the knowledge of the ideal solution $\mathbf{z}^*$. So we need to optimize each objective function before we can compute the $\mathcal{L}_p$ metric.

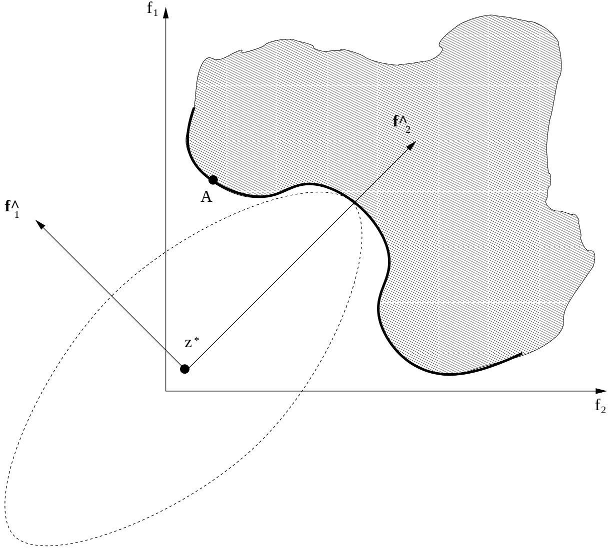
Rotated Weighted Metric Method

- Instead of using directly the $\mathcal{L}_p$ metric as it is stated in equation (3), the $\mathcal{L}_p$ metric can be applied with an arbitrary rotation from the ideal point.
- Let us say that the relation between the rotated objective vector $\hat{\mathbf{f}}$ and the original objective vector $\mathbf{f}$ is : $\hat{\mathbf{f}} = R\mathbf{f}$, where R is the rotation matrix of size $M \times M$.
- The modified $\mathcal{L}_p$ metric is:

$$\hat{\mathcal{L}}_p(\mathbf{x}) = \left(\sum_{m=1}^{M} w_m |\hat{f}_m(\mathbf{x}) - \mathbf{z}_m^*|^p \right)^{\frac{1}{p}} \tag{5}$$

Rotated Weighted Metric Method

Illustration

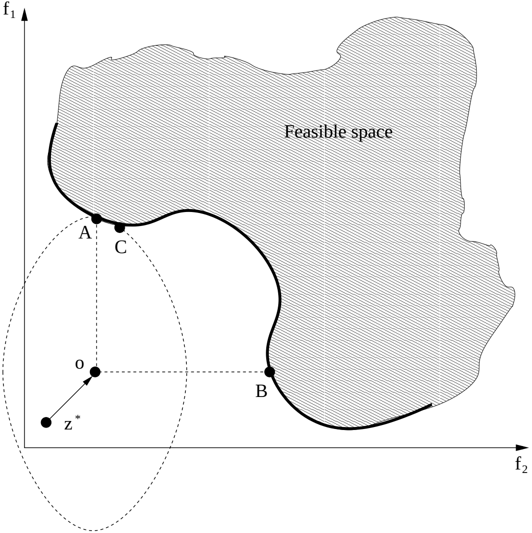


Dynamically Changing the Ideal Solution

- **Idea**: update the reference point $\mathbf{z}^*$ every time that a Pareto-optimal solution is found.
- The $\mathcal{L}_p$ distance of the ideal solution comes closer to the Pareto-optimal front and new previous undiscovered solution can now be found.

Dynamically Changing the Ideal Solution

Illustration



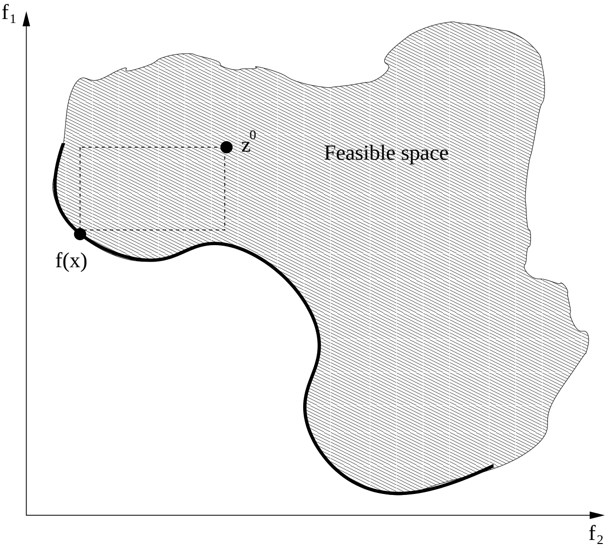
Benson's Method

- **Idea**: take the reference solution $\mathbf{z}^*$ randomly from the feasible region. Let us call it $\mathbf{z}^0$.
- The non-negative difference $(z_m^0 - f_m(\mathbf{x}))$ for each objective is calculated and their sum is maximized

$$
\begin{array}{ll}
max & \mathcal{B}(\mathbf{x}) = \sum_{m=1}^{M} max(0, (z_m^0 - f_m(\mathbf{x}))) \\
subject\ to & \mathcal{F}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})] \leq \mathbf{z^0} \\
& \mathcal{G}(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_J(\mathbf{x})] \geq 0 \\
& \mathcal{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_K(\mathbf{x})] = 0 \\
& x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, \ldots, N
\end{array}
\tag{6}
$$

Benson's Method

Illustration



Benson Method

Advantages and disadvantages

Advantages

- Avoid scaling problems: individual differences can be normalized before the summation.
- If the point $\mathbf{z}^0$ is chosen appropriately than this method can be used also for non-convex multi-objective problems.

Disadvantages

- It has an additional number of constraints.
- The objective function is non-differentiable, causing difficulties for gradient based methods.

Value Function Method

(or Utility Function Method)

- **Idea**: the user provides an utility function $\mathcal{U} : \mathbb{R}^M \rightarrow \mathbb{R}$ relating all M objectives.
- The utility function must be valid over the entire feasible space.
- Among two solutions i and j, i is preferred to j if $\mathcal{U}(\mathbf{f}(\mathbf{x}_i)) > \mathcal{U}(\mathbf{f}(\mathbf{x}_j))$.

$$
\begin{array}{ll}
\mathit{max} & \mathcal{U}(\mathbf{f}(\mathbf{x})) \\
\mathit{subject\ to} & \mathcal{G}(\mathbf{x}) = [g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_J(\mathbf{x})] \geq 0 \\
 & \mathcal{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_K(\mathbf{x})] = 0 \\
 & x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, \ldots, N
\end{array}
\qquad (7)
$$

where $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})]^T$.

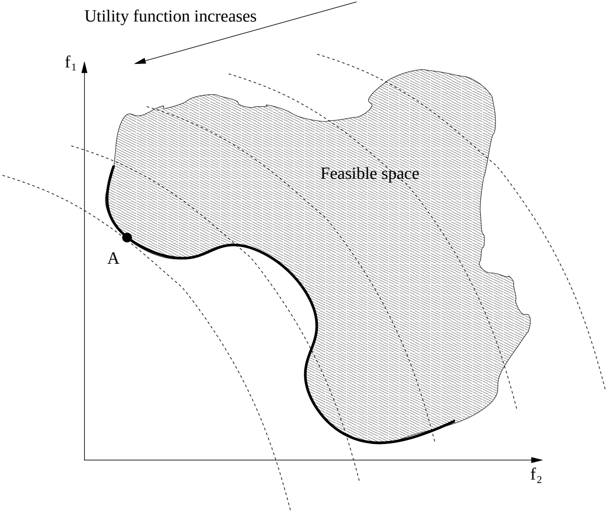
- The utility function must be *strongly decreasing* before it can be used in multi-objective optimization.
- This means that the preference of a solution must increase if one of the objective function values is decreased while keeping the other objective function values the same.

Theorem

Let the utility function $\mathcal{U} : \mathbb{R}^M \rightarrow \mathbb{R}$ *be strongly decreasing. Let* $\mathcal{U}$ *attain its maximum at* $\mathbf{f}^*$. *Then* $\mathbf{f}^*$ *is Pareto-optimal.*

Value Function Method

Illustration



Value Function Method

Advantages and disadvantages

Advantages

- The idea is simple and ideal, if adequate value function information is available.
- Mainly used for multi-attribute decision analysis problems with a discrete set of feasible solutions.

Disadvantages

- The solution entirely depends on the chosen value function.
- It requires the users to come up with a value function that is globally applicable over the entire search space.

For Further Reading

Kalyanmoy Deb
Multi-Objective Optimization Using Evolutionary Algorithms.
John Wiley & Sons, Inc, New York, NY, USA, 2001.

Kaisa Miettinen
Nonlinear Multiobjective Optimization.
Kluwer Academic Publishers, 1999. International Series in Operations Research & Management Science, Volume 12.