Multi-Objective Optimization

Pareto Front, Weighted Sum Method and ε-Constraint

English

Wolfgang Boehme

wolfgang.boehme@economic-modeling.de

www.economic-modeling.de

25 October 2026

This document introduces multi-objective optimization. Prerequisites: basic knowledge of linear programming.

The documentation on this website was developed with the support of the AI Claude (Anthropic).

Table of Contents

1 Introduction and Motivation **2**

2 Fundamental Concepts of Multi-Objective Optimization **2**
2.1 The General Multi-Objective Optimization Problem 2
2.2 Ideal and Nadir Point 3
2.3 Geometric Interpretation 3

3 Solution Methods **4**
3.1 Overview 4
3.2 Weighted Sum Method 4
3.2.1 Sampling the Weight Space 4
3.3 ε-Constraint Method 5
3.3.1 Algorithm: Sampling with ε-Constraint 5
3.4 Comparison of Methods 5

4 Complete Example **6**
4.1 Problem Statement 6
4.2 Ideal and Nadir Point 6
4.3 Weighted Sum Method: Selected Solutions 6
4.4 Pareto Front: Graphical Representation 7
4.5 ε-Constraint: Computation 7

5 Multi-Objective Optimization with $k > 2$ Objectives **7**
5.1 Challenges with Many Objectives 7
5.2 Visualisation Approaches for $k > 2$ 8
5.3 Example: 3 Objectives 8

6 Decision Support: Choosing a Solution **8**
6.1 Methods for Solution Selection 8
6.2 Normalisation 9

7 Complexity and Practical Remarks **9**
7.1 Complexity 9
7.2 Practical Software 9

8 Summary **10**

1 Introduction and Motivation

In der Praxis müssen Entscheidungen häufig mehrere, oft **konkurrierende Ziele** gleichzeitig berücksichtigen. Klassische Beispiele:

Typical Multi-Objective Problems

- **Portfolio optimisation:** Maximising return *and* minimising risk
- **Production planning:** Maximising profit *and* minimising environmental impact
- **Vehicle design:** Minimising fuel consumption *and* maximising performance
- **Logistics:** Minimising transport costs *and* minimising delivery time
- **Medicine:** Maximising efficacy *and* minimising side effects

The core problem: multiple objectives can generally **not be optimised simultaneously**. Improving one objective often comes at the expense of another (**trade-off**). Instead of a single optimal solution, we therefore seek the set of all *efficient* compromise solutions – the **Pareto front**.

2 Fundamental Concepts of Multi-Objective Optimization

2.1 The General Multi-Objective Optimization Problem

Definition: Multi-Objective Optimization Problem (MOP)

A **multi-objective optimization problem** (MOP) reads:

$$\min_{\mathbf{x}\in\mathcal{F}} \mathbf{z}(\mathbf{x}) = \big(z_1(\mathbf{x}),\, z_2(\mathbf{x}),\, \ldots,\, z_k(\mathbf{x})\big)^\top$$

wobei $\mathcal{F} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b},\ \mathbf{x} \geq \mathbf{0}\}$ der **Entscheidungsraum** und $\mathbf{z} : \mathcal{F} \to \mathbb{R}^k$ der **Zielraumabbildung** (Kriterienvektor) ist.
Maximisation objectives $\max z_i(\mathbf{x})$ are converted via $\min(-z_i(\mathbf{x}))$.

Definition: Dominance

Let $\mathbf{x}, \mathbf{y} \in \mathcal{F}$ be two feasible solutions.

- $\mathbf{x}$ **dominates** $\mathbf{y}$ (notation: $\mathbf{x} \prec \mathbf{y}$), if:
  1. $z_i(\mathbf{x}) \leq z_i(\mathbf{y})$ for all $i = 1, \ldots, k$ (not worse in any objective), and
  2. $z_j(\mathbf{x}) < z_j(\mathbf{y})$ for at least one j (strictly better in at least one objective).
- $\mathbf{x}$ and $\mathbf{y}$ are **incomparable** if neither $\mathbf{x} \prec \mathbf{y}$ nor $\mathbf{y} \prec \mathbf{x}$ holds.

Definition: Pareto Optimality

A solution $\mathbf{x}^* \in \mathcal{F}$ is called **Pareto-optimal** (or efficient) if no other feasible solution $\mathbf{y} \in \mathcal{F}$ exists that dominates $\mathbf{x}^*$:

$$\nexists\, \mathbf{y} \in \mathcal{F} : \mathbf{y} \prec \mathbf{x}^*$$

The set of all Pareto-optimal solutions is called the **Pareto set** (in decision space) or **Pareto front** (in objective space).

2.2 Ideal and Nadir Point

Definition: Ideal and Nadir Point

- The **ideal point** $\mathbf{z}^{\text{id}} \in \mathbb{R}^k$ contains the best achievable values of each objective:

  $$z_i^{\text{id}} = \min_{\mathbf{x} \in \mathcal{F}} z_i(\mathbf{x}) \quad \forall\, i$$

  The ideal point is generally **not attainable** (it lies outside the Pareto front).

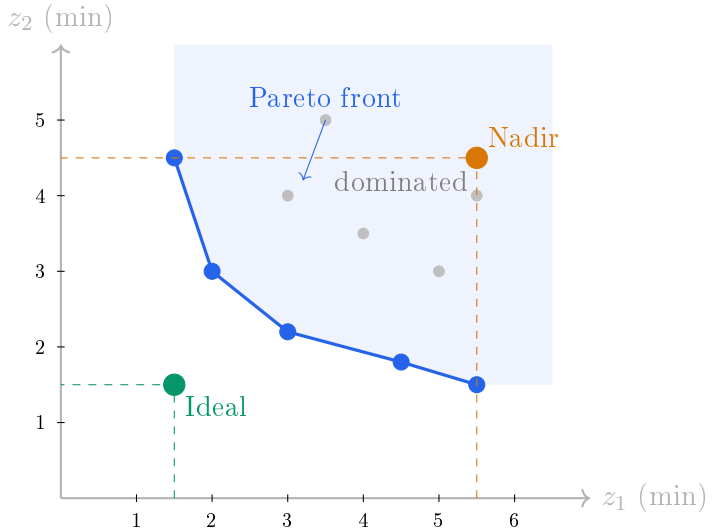
- The **nadir point** $\mathbf{z}^{\text{nad}} \in \mathbb{R}^k$ contains the worst values among the Pareto-optimal solutions:

  $$z_i^{\text{nad}} = \max_{\mathbf{x}^* \in \mathcal{P}} z_i(\mathbf{x}^*) \quad \forall\, i$$

  where $\mathcal{P}$ denotes the Pareto set.

The region between ideal and nadir point describes the **trade-off range** of the Pareto front.

2.3 Geometric Interpretation



The blue curve shows the Pareto front: no point on it can be improved in one objective without worsening another. Grey points are dominated.

3 Solution Methods

3.1 Overview

There are two fundamental classes of methods:

- **A priori methods:** Preferences are specified *before* optimisation (e.g. weighted sum method).
- **A posteriori methods:** The complete Pareto front is computed; the decision maker chooses afterwards.

3.2 Weighted Sum Method

Definition: Weighted Sum Method

The **weighted sum method** scalarises the objectives by weighting:

$$\min_{\mathbf{x}\in\mathcal{F}} \sum_{i=1}^{k} w_i \cdot z_i(\mathbf{x}), \quad \text{wobei } w_i \geq 0,\ \sum_{i=1}^{k} w_i = 1$$

For each weight vector $\mathbf{w} = (w_1, \ldots, w_k)$ on the unit simplex a scalar LP is solved.

Theorem 1: Correctness of the Weighted Sum Method

Every solution of the weighted scalarisation problem with $\mathbf{w} > \mathbf{0}$ (all weights positive) is Pareto-optimal. For convex Pareto fronts, varying $\mathbf{w}$ yields all Pareto-optimal solutions.

Proof. Sei $\mathbf{x}^*$ optimal für $\mathbf{w} > \mathbf{0}$ und angenommen, $\mathbf{x}^*$ sei nicht Pareto-optimal. Dann existiert $\mathbf{y} \in \mathcal{F}$ mit $z_i(\mathbf{y}) \leq z_i(\mathbf{x}^*)$ für alle i und $z_j(\mathbf{y}) < z_j(\mathbf{x}^*)$ für ein j. Dann gilt:

$$\sum_i w_i z_i(\mathbf{y}) < \sum_i w_i z_i(\mathbf{x}^*)$$

(da $w_j > 0$), was der Optimalität von $\mathbf{x}^*$ widerspricht. □

Limitation: For **non-convex** Pareto fronts the weighted sum method cannot reach certain parts (concave regions). The ε-constraint method is more robust.

3.2.1 Sampling the Weight Space

For $k = 2$ objectives the weights are sampled uniformly:

$$\mathbf{w}^{(t)} = \left(\frac{t}{T},\ 1 - \frac{t}{T}\right), \quad t = 0, 1, \ldots, T$$

For $k > 2$ the unit simplex is sampled by random sampling or regular subdivision.

3.3 ε-Constraint Method

Definition: ε-Constraint Method

In the **ε-constraint method** a primary objective z_p is optimised while all other objectives are bounded by ε_i:

$$\min_{\mathbf{x}\in\mathcal{F}} \; z_p(\mathbf{x}) \quad \text{s.t.} \quad z_i(\mathbf{x}) \le \varepsilon_i \quad \forall\, i \neq p$$

By systematically varying ε_i all Pareto-optimal solutions (including non-convex parts) are generated.

Theorem 2: Completeness of the ε-Constraint Method

Every Pareto-optimal solution can be found by the ε-constraint method with an appropriate choice of ε and primary objective.

Proof sketch. Let $\mathbf{x}^*$ be Pareto-optimal. Set $\varepsilon_i = z_i(\mathbf{x}^*)$ for all $i \neq p$. Then $\mathbf{x}^*$ is feasible for the ε-constraint problem. Suppose another solution $\mathbf{y}$ achieves $z_p(\mathbf{y}) < z_p(\mathbf{x}^*)$ with $z_i(\mathbf{y}) \le \varepsilon_i$. Since $z_i(\mathbf{y}) \le z_i(\mathbf{x}^*)$ and $z_p(\mathbf{y}) < z_p(\mathbf{x}^*)$, $\mathbf{y}$ would dominate $\mathbf{x}^*$ – contradicting Pareto optimality. □

3.3.1 Algorithm: Sampling with ε-Constraint

1. Compute the range of each secondary objective: $[z_i^{\min}, z_i^{\max}]$ by separate LP solutions.
2. Divide the range into T steps: $\varepsilon_i^{(t)} = z_i^{\min} + t \cdot \frac{z_i^{\max} - z_i^{\min}}{T}$.
3. Solve the constrained LP for each ε value.
4. Remove dominated points from the result set.

3.4 Comparison of Methods

Property	Weighted sum	ε-Constraint
Non-convex fronts	Incomplete	Complete
Computational cost	Low	Medium
Anzahl LP-Probleme	T	$T \cdot (k-1)$
Interpretation	Intuitive	Direct control
Uniform sampling	No	Yes

4 Complete Example

4.1 Problem Statement

Example: Bicriteria Production Planning

A company produces two products with two competing objectives: profit maximisation and resource conservation.

$$\max\ z_1 = 5x_1 + 4x_2 \qquad \text{(profit)}$$

$$\min\ z_2 = 3x_1 + x_2 \qquad \text{(resource consumption)}$$

$$\begin{aligned} 6x_1 + 4x_2 &\leq 24 \\ x_1 + 2x_2 &\leq 6 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Reformulation: minimisation problem $\min\ (-5x_1 - 4x_2)$ and $\min\ (3x_1 + x_2)$.

4.2 Ideal and Nadir Point

Ideal point: Optimise each objective separately.

z_1 maximieren: Simplex liefert $x_1 = 3$, $x_2 = 1{,}5$, $z_1^{\text{id}} = 21$.

z_2 minimieren: Ecke $(0, 0)$ liefert $z_2^{\text{id}} = 0$ (aber $z_1 = 0$ – kein Zielkonflikt sichtbar). Bei $x_1, x_2 \geq 0$: $z_2^{\min} = 0$ bei $(0, 0)$.

Nadir point: Worst values on the Pareto front. $z_1^{\text{nad}} = 0$ (bei $(0, 0)$), $z_2^{\text{nad}} = 10{,}5$ (bei $x_1 = 3, x_2 = 1{,}5$).

Trade-off: Maximum profit $z_1 = 21$ costs resources $z_2 = 10.5$. Minimum resources $z_2 = 0$ yields no profit.

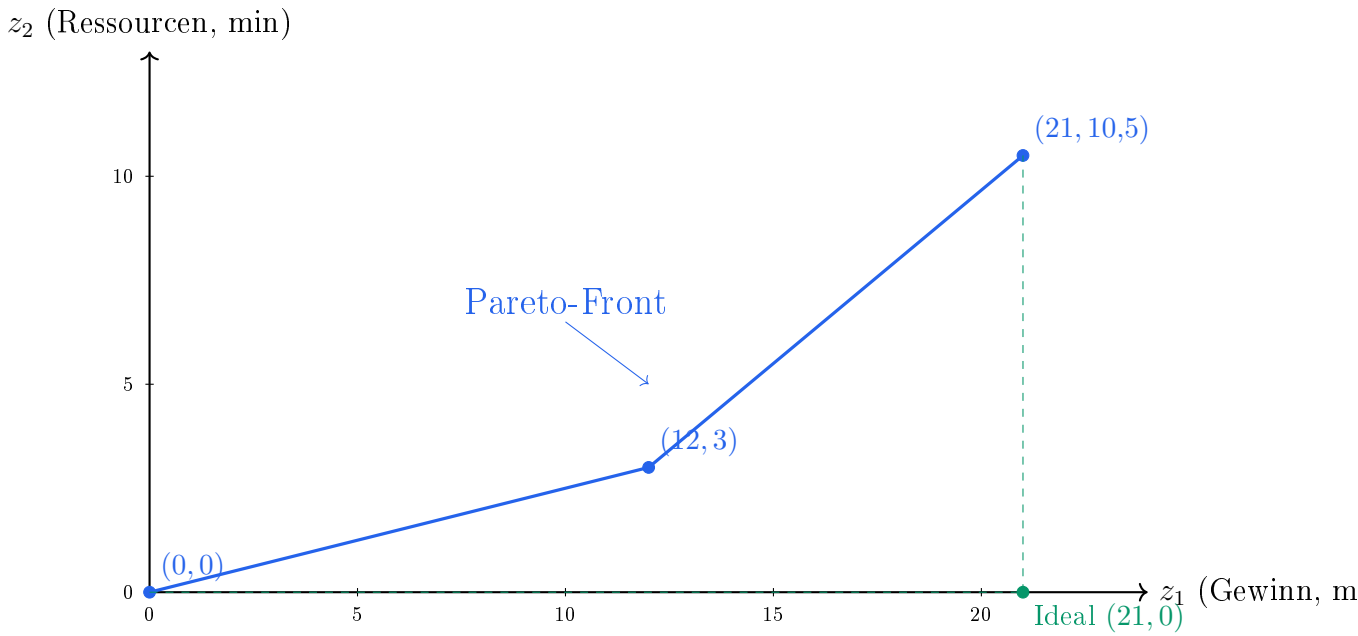
4.3 Weighted Sum Method: Selected Solutions

For $k = 2$ we use $\mathbf{w} = (w, 1-w)$ with $w \in [0, 1]$. Skalarisiertes Problem: $\min\ -w(5x_1 + 4x_2) + (1 - w)(3x_1 + x_2) = \min\ (3 - 8w)x_1 + (1 - 5w)x_2$

w	$c_1 = 3 - 8w$	$c_2 = 1 - 5w$	x_1^*	x_2^*	(z_1, z_2)
0,0	3	1	0	0	(0, 0)
0,2	1,4	0	0	3	(12, 3)
0,3	0,6	−0,5	0	3	(12, 3)
0,5	−1	−1,5	3	1,5	(21, 10,5)
0,8	−3,4	−3	3	1,5	(21, 10,5)
1,0	−5	−4	3	1,5	(21, 10,5)

Pareto-optimale Punkte (nach Entfernung dominierter Lösungen): $(0, 0) \to (z_1 = 0, z_2 = 0)$, $(0, 3) \to (z_1 = 12, z_2 = 3)$, $(3, 1{,}5) \to (z_1 = 21, z_2 = 10{,}5)$.

4.4 Pareto Front: Graphical Representation



4.5 ε-Constraint: Computation

Primary objective: max z_1. Constraint: $z_2 = 3x_1 + x_2 \leq \varepsilon$.

ε	x_1^*	x_2^*	z_1	z_2	Status
0	0	0	0	0	Pareto-optimal
2	0	2	8	2	Pareto-optimal
3	0	3	12	3	Pareto-optimal
5	0,5	3,5	16,5	5	Pareto-optimal
8	1,75	2,75	19,75	8	Pareto-optimal
10,5	3	1,5	21	10,5	Pareto-optimal

The ε-constraint method covers the same range as the weighted sum method here, since the Pareto front is convex.

5 Multi-Objective Optimization with $k > 2$ Objectives

5.1 Challenges with Many Objectives

As the number of objectives k grows, the challenges increase:

- The Pareto front is a $(k-1)$-dimensional manifold – no longer graphically representable.

- The proportion of dominated solutions *decreases* (for $k \to \infty$ random points rarely dominate each other).
- Computational cost grows with k.

5.2 Visualisation Approaches for $k > 2$

- **Parallel coordinates:** Each objective as a vertical axis, solutions as polylines.
- **Scatter plot matrix:** All pairs (z_i, z_j) in a matrix.
- **Radar chart (spider web):** Suitable for a few solutions.
- **Heatmap:** Objective values as colour intensity in a table.

5.3 Example: 3 Objectives

Example with 3 Objectives

$$\max\ z_1 = 5x_1 + 4x_2, \quad \max\ z_2 = 2x_1 + 6x_2, \quad \min\ z_3 = 3x_1 + x_2$$

$$6x_1 + 4x_2 \leq 24, \quad x_1 + 2x_2 \leq 6, \quad x_1, x_2 \geq 0$$

Selected Pareto-optimal solutions (computed with the MOO tool):

x_1	x_2	z_1 (max)	z_2 (max)	z_3 (min)	Dominating vertex
0	0	0	0	0	None (Pareto)
0	3	12	18	3	Pareto
3	1,5	21	15	10,5	Pareto
4	0	20	8	12	Pareto

We see: z_1 and z_2 favour different vertices, z_3 favours the origin. The Pareto front includes all four vertices (none dominates another completely).

6 Decision Support: Choosing a Solution

6.1 Methods for Solution Selection

After computing the Pareto front the decision maker must select a solution. Common methods:

1. Utopia point approximation (compromise solution):

$$\min_{\mathbf{x} \in \mathcal{P}} \left\| \frac{\mathbf{z}(\mathbf{x}) - \mathbf{z}^{\mathrm{id}}}{\mathbf{z}^{\mathrm{nad}} - \mathbf{z}^{\mathrm{id}}} \right\|_p$$

The point on the Pareto front with the smallest normalised distance to the ideal point ($p = 2$: Euclidean, $p = \infty$: Chebyshev).

2. Chebyshev scalarisation:

$$\min_{\mathbf{x}\in\mathcal{F}} \max_{i=1,\dots,k} w_i \left| z_i(\mathbf{x}) - z_i^{\text{id}} \right|$$

Also finds points on non-convex Pareto fronts.

3. Interactive methods: The decision maker progressively refines preferences, e.g. by answering pairwise comparisons (*STEM*, *NIMBUS*).

6.2 Normalisation

Before comparison, objectives should be normalised:

$$\tilde{z}_i(\mathbf{x}) = \frac{z_i(\mathbf{x}) - z_i^{\text{id}}}{z_i^{\text{nad}} - z_i^{\text{id}}} \in [0, 1]$$

7 Complexity and Practical Remarks

7.1 Complexity

Computing the complete Pareto front is in general NP-hard (when objectives are non-linear or variables are integer). For linear MOPs the weighted sum method with T weight vectors requires exactly T LP evaluations in $O(T \cdot \text{LP-time})$.

7.2 Practical Software

- **Python:** `pymoo` (NSGA-II, NSGA-III, weighted sum, ε-constraint)
- **MATLAB:** `gamultiobj` (genetic algorithm for MOP)
- **Gurobi/CPLEX:** Bicriteria LP optimisation via parametric methods
- **jMetal:** Java framework for evolutionary multi-objective optimisation

8 Summary

Key Points: Multi-Objective Optimization

1. With **competing objectives** there is no unique optimal solution, but a set of efficient compromise solutions.
2. A solution is **Pareto-optimal** if it cannot be improved in any objective without worsening another.
3. The **ideal point** contains the best individual values (usually unattainable); the **nadir point** the worst values on the Pareto front.
4. The **weighted sum method** solves a scalar LP for each weight vector – simple but incomplete for non-convex fronts.
5. The **ε-constraint method** bounds secondary objectives and is complete for arbitrary fronts.
6. For $k = 2$ objectives the Pareto front is a curve in 2D objective space; for $k > 2$ a higher-dimensional manifold.
7. **Normalisation** of objective values is essential for solution selection and method comparison.
8. For **solution selection** compromise methods (Chebyshev, Euclidean distance) or interactive procedures are suitable.