Multi-criteria Optimization

Overview for this Week

Bernd Bischl   Frank Hutter   Lars Kotthoff   Marius Lindauer

Notation

- Admissible set $\mathcal{X} \subset \mathbb{R}^n$
- Target region $\mathbb{R}^m$
- Multi-criteria objective function $f : \mathcal{X} \to \mathbb{R}^m$
- Objective function vector $f(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_m(\mathbf{x}))^\top \in \mathbb{R}^m$, which maps $\mathbf{x}$ into the space $\mathbb{R}^m$.

w.l.o.g. we look at minimization problems.

Introduction example I

Often we want to solve optimization problems concerning several goals.

- Medicine: maximum effect, but minimum side effect of a drug.
- Finances: maximum return, but minimum risk of an equity portfolio.
- Production planning: maximum revenue, but minimum costs.
- Booking a hotel: maximum rating, but minimum costs.

A *simple* approach would be to formulate all but one objective function simplified as a secondary condition.

Introduction example II

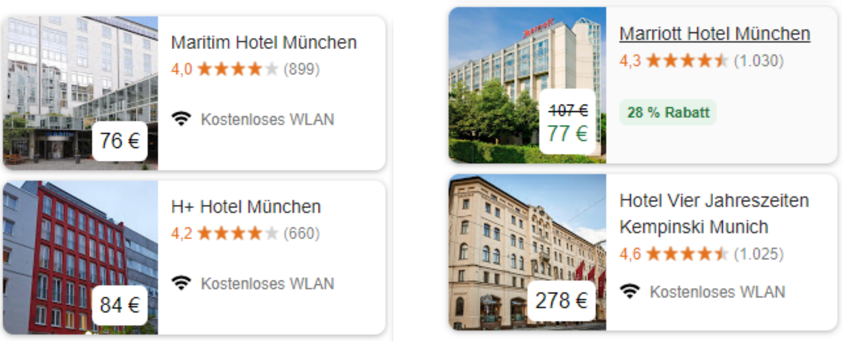
Example:
Maximize proceeds subject to costs $\leq C, C \in \mathbb{R}$.

Disadvantages:

- The result depends of course on how we select C and usually returns different solutions for different values of C.
- The more target functions we optimize, the more difficult such a formulation becomes.

Target: find a general approach to solving multi-criteria problems.

Introduction example III



When booking a hotel: find the hotel with

- Minimum price per night (**costs**) and
- Maximum user rating (**performance**).

Since we limit ourselves to minimizing problems, we minimize negative valuations.

Introduction example IV

The goals often conflict with each other:

- Lower price $\rightarrow$ often lower hotel rating.
- Better rating $\rightarrow$ frequently higher price.

Example: (negative) average rating by hotel guests (1 - 5) vs. average price per night in USD from hotels on Expedia (excerpt).

In addition, targets are often not comparable because they have different units, for example.

- Left: a hotel with rating 4 for 89 Euro ($y_1 = (89, -4.0)$ would be preferred to a hotel $y_2 = (108, -4.0)$ (left)
- Right: how to decide if $y_1 = (89, -4.0)$ and $y_1 = (95, -4.5)$?
- How much is a *scoring point* worth?

Definition: multi-criteria optimization problem

Be $\mathcal{X} \subset \mathbb{R}^n$ and $f : \mathcal{X} \to \mathbb{R}^m$, $m \geq 2$. A **multi-criteria optimization problem** is defined by

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \Leftrightarrow \min_{\mathbf{x} \in \mathcal{X}} \left(f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x}) \right).$$

- **Aim:** minimize multiple target functions simultaneously.
- Objective functions are often contradictory here.
- Often no clearly best solution, but a set of solutions that are equally good.
- Synonym terms: multi-criteria optimization, multi-objective optimization, Pareto optimization

How to define optimality? I

Be still $y = (\text{price}, -\text{evaluation})$. In some cases it is *clear* which point is the better one:

- The solution $\mathbf{y}_1 = (89, -4.0)$ dominates $\mathbf{y}_2 = (108, -4.0)$: $\mathbf{y}_1$ is not worse in any dimension and is better in one dimension. $\mathbf{y}_2$ gets **dominated** of $\mathbf{y}_1$

$$\mathbf{y}_2 \prec \mathbf{y}_1.$$

How to define optimality? II

For the points $\mathbf{y}_1 = (89, -4.0)$ and $\mathbf{y}_2 = (95, -4.5)$ we cannot say which point is the better one.

- We designate the points as **equivalent** and write

$$\mathbf{y}_1 \not\prec \mathbf{y}_2 \text{ und } \mathbf{y}_2 \not\prec \mathbf{y}_1.$$

- The set of all equivalent points that are not dominated by another point is called the **Pareto front**.

Pareto sets und Pareto optimality I

Definition:
Given a multicriteria optimization problem

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})) \,, \quad f_i : \mathcal{X} \to .$$

- A solution $\mathbf{x}_1$ **(Pareto-) dominates** $\mathbf{x}_2$, if $f(\mathbf{x}_1) \prec f(\mathbf{x}_2)$, i.e.
  1. $f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2)$ for all $i \in \{1, 2, \ldots, m\}$ und
  2. $f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2)$ for at least one $j \in \{1, 2, \ldots, m\}$
- A solution $\mathbf{x}^*$ that is not dominated by any other solution is called **Pareto optimal**.
- The set of all Pareto optimal solutions is called **Pareto set**
  $\mathcal{P} := \{\mathbf{x} \in \mathcal{X} | \; \nexists \; \tilde{\mathbf{x}} \text{ with } f(\tilde{\mathbf{x}}) \prec f(\mathbf{x})\}$
- $\mathcal{F} = f(\mathcal{P}) = \{f(\mathbf{x}) | \mathbf{x} \in \mathcal{P}\}$ is called **Pareto front**.

Example: an objective function I

We consider the minimization problem

$$\min_{x} f(x) = (x-1)^2, \qquad 0 \leq x \leq 3.$$

The optimum is at $x^* = 1$.

Example: two target functions I

We extend the above problem to two objective functions $f_1(x) = (x-1)^2$ and $f_2(x) = 3(x-2)^2$, thus

$$\min_x f(x) := (f_1(x), f_2(x)), \qquad 0 \leq x \leq 3.$$

Example: two target functions II

We consider the functions in the objective function space $f(\mathcal{X})$ by drawing the objective function values $(f_1(x), f_2(x))$ for all $0 \leq x \leq 3$.

The Pareto front is shown in green. The Pareto front cannot be left without getting worse in at least one objective function.

Lecture Overview

Two solutions I

- The Pareto set is a set of equally optimal solutions.
- One is often interested in a **single** optimal solution.
- Without further information no unambiguous optimal solution can be determined $\rightarrow$ decision must be based on other criteria.

Basically, there are two possible solutions:

- **A-priori approach**: user preferences are considered **before** the optimization process
- **A-posteriori approach**: user preferences are considered **after** the optimization process

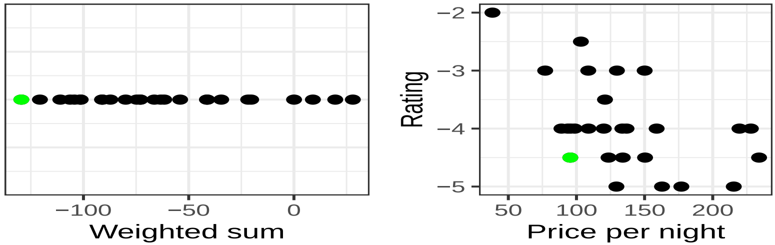
A-priori procedure I

Example: weighted total

Previous knowledge: one rating point is worth 50 Euro to a customer.

$\rightarrow$ We optimize the weighted sum:

$$\min_{\text{Hotel}} (\text{Price / Night}) - 50 \cdot \text{Rating}$$



A-priori procedure II

A-priori approach: weighted sum

$$\min_{x \in \mathcal{X}} \qquad \sum_{i=1}^{m} w_i f_i(\mathbf{x})$$

$$\text{with} \quad w_i \geq 0$$

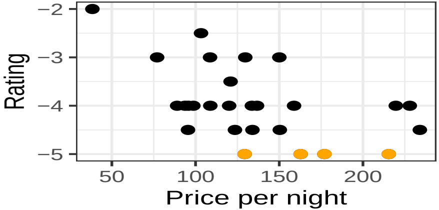
A-priori procedure III

Example: lexicographic method

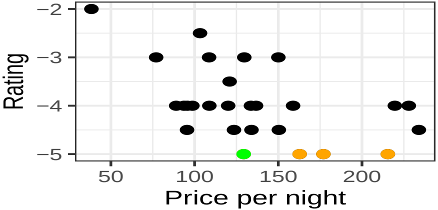
Previous knowledge: customer prioritizes rating over price.

$\rightarrow$ optimize target functions one after the other.

1) max. rating



2) min. price



A-priori procedure IV

A-priori approach: lexicographic method

$$
\begin{aligned}
y_1^* &= \min_{\mathbf{x}\in\mathcal{X}} f_1(\mathbf{x}) \\
y_2^* &= \min_{\mathbf{x}\in\{\mathbf{x} \mid f_1(\mathbf{x})=y_1^*\}} f_2(\mathbf{x}) \\
y_3^* &= \min_{\mathbf{x}\in\{\mathbf{x} \mid f_1(\mathbf{x})=y_1^* \wedge f_2(\mathbf{x})=y_2^*\}} f_3(\mathbf{x}) \\
&\vdots
\end{aligned}
$$

Also here: different sequences provide different solutions.

A-priori procedure V

Summary a-priori approach:

- In a single application, only one solution is obtained, which depends on the a-priori selection of weights, order, etc.
- In case of repeated use, several solutions are obtained if weights, order, etc. are systematically varied.
- Usually there are solutions that remain hidden from these methods.
- Implicit assumption: monocritical optimization simple

A-posteriori procedure I

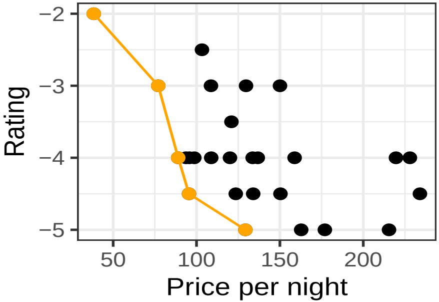
A-posteriori methods, on the other hand, have the goal to

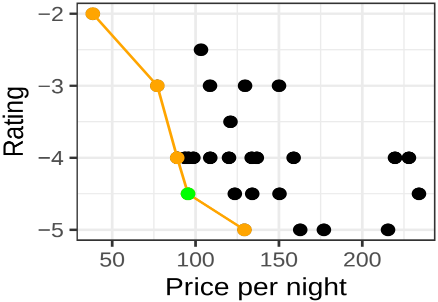
- find the set of **all** optimal solutions (the Pareto set),
- select (if necessary) an optimal solution based on prior knowledge or individual preferences.

A-posteriori methods are therefore the more generic approach to solving a multi-criteria optimization problem.

A-posteriori procedure II

Example:
A user gets more detailed information about all Pareto optimal hotels (left) and chooses an optimal solution (right) based on previous knowledge or additional criteria (e.g. location of the hotel).

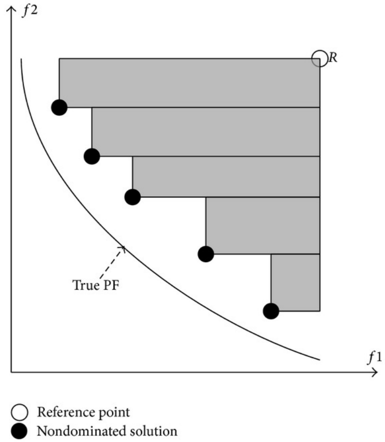




Evaluation of solutions I

A common metric for evaluating a set of solutions $\mathcal{P} \subset \mathcal{X}$ is the **dominated hypervolume** (S-metric), which we call $S(\mathcal{P}, R)$.

Evaluation of solutions II



Evaluation of solutions III

The dominated hypervolume of the set of points $\mathcal{P} \subset \mathcal{X}$ (here: 5 black points) is the area in the target function space (regarding a reference point R) which is dominated by points $\mathcal{P}$.

Lecture Overview

1. Solvers
2. Evolutionary multi-objective optimization algorithms (EMOA)
3. SMS-EMOA

A-posteriori methods and evolutionary algorithms I

Evolutionary algorithms return as a solution a **population** of solution candidates. Evolutionary multi-objective (EMO) algorithms aim to provide a set of solution candidates that corresponds to the Pareto set as well as possible.

A-posteriori methods and evolutionary algorithms II

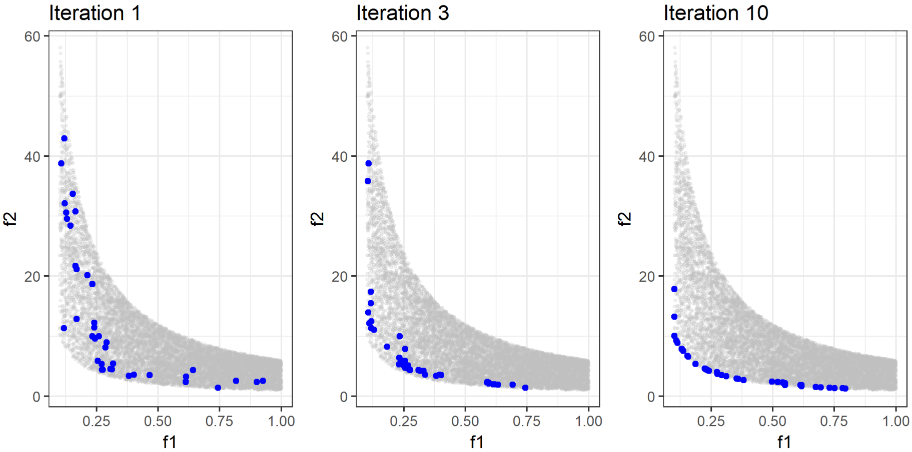


Image of the function (grey) and target function values $(f_1(\mathbf{x}), f_2(\mathbf{x}))$ for $\mathbf{x} \in \mathcal{P}_i, i = 1, 3, 10$.

A-posteriori methods and evolutionary algorithms III

Algorithm 1 Evolutionary algorithm

Initialize and rate population $P_0 \subset \mathcal{X}$ with $|\mathcal{P}| = \mu$ $t \leftarrow 0$ **repeat**

V

until

ariation: generate offspring Q_t with $|Q_t| = \lambda$ Rate fitness of offspring Selection: select survivors P_{t+1} $t \leftarrow t + 1$ Stop criterion fulfilled

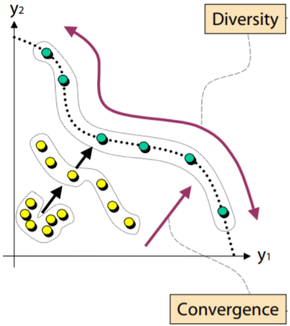
The population of solution candidates consists of $\mathbf{x} \in \mathcal{X}$.

Objectives of an evolutionary strategy I

The aim is to select the evolution strategy in such a way that the algorithm provides an approximation of the Pareto front, where

1. The individuals of the population (or the corresponding functional values in the target function space) **converge** to the Pareto front.
2. The individuals of the population provide a **diverse** as possible approximation of the Pareto front.

Objectives of an evolutionary strategy II



Caution: in this graphic the objective function values are exceptionally **maximized**.

NSGA-II I

The **non-dominated sorting genetic algorithm (NSGA-II)** was published by K. Deb in 2002.

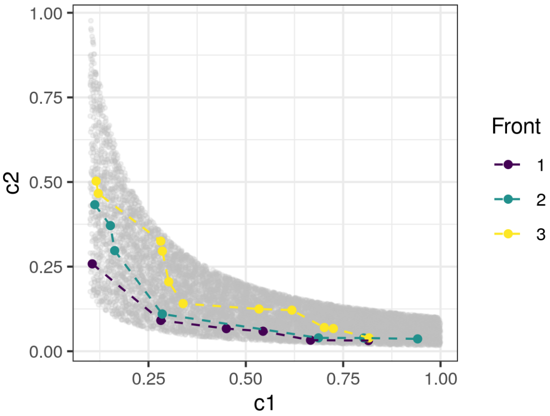
- The NSGA-II follows a $(\mu + \lambda)$ strategy
- All previously discussed strategies can be used as a variation strategy; the original paper uses polynomial mutation and simulated binary crossover.
- The selection strategy is based on
  - **Non-dominated sorting**
  - **Crowding distance assignment**

NSGA-II: non-dominated sorting I

We subdivide $R_t = P_t \cup Q_t$ into fronts $F_1, F_2, F_3, \ldots$ such that

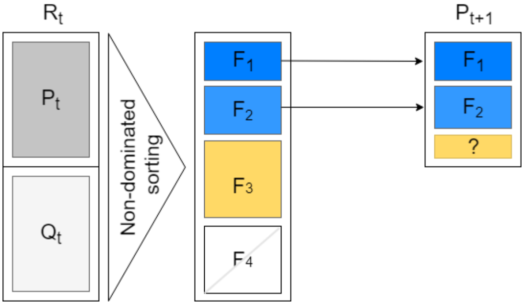
- the points in the fronts are equivalent to each other, and
- that any point $\mathbf{x} \in F_1$ dominates any point from $F_2, F_3, F_4 \ldots$; any point $\mathbf{x} \in F_2$ dominates all points from $F_3, F_4, \ldots$, etc.
  We write $F_1 \prec F_2 \prec F_4 \prec \ldots$

NSGA-II: non-dominated sorting II



NSGA-II: non-dominated sorting III

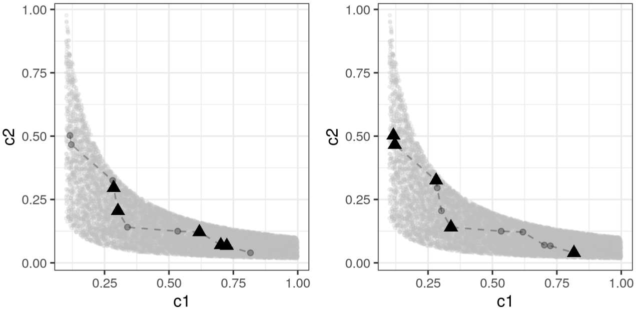
Which individuals survive? We fill μ places one by one with $F_1, F_2, \ldots$ until a front can no longer **fully** survive (here: F_3).



Which individuals survive from F_3? $\rightarrow$ **crowding sort**

NSGA-II: crowding sort I

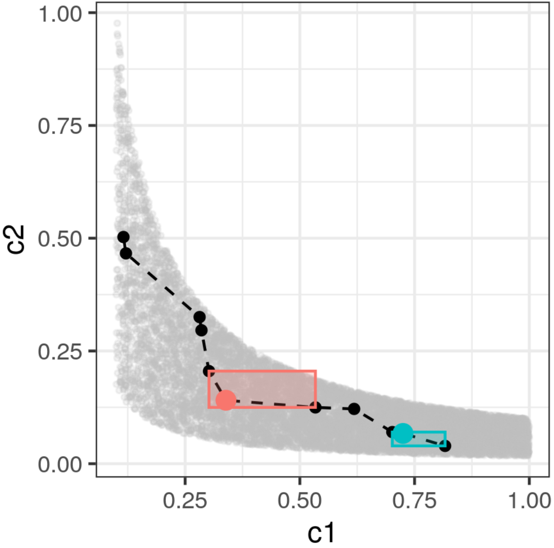
Idea: add a good representative of the front F_3 if possible.



The points on the left (marked by a triangle) do not represent the front very well because they are very close together. The front is better represented by the points on the right plot.

NSGA-II: crowding sort II

Crowding sort sorts the individuals based on their crowding distance:



One point with high crowding distance (red) and one point with very small crowding distance (blue).

Algorithm 2 NSGA-II

Initialize population P_0, $t \leftarrow 0$ $F_1, F_2, F_3, ... \leftarrow \texttt{nondominated-sort}(P_0)$ Generate Q_0 by binary tournament selection, recombination and mutation **repeat**

until

F$_1, F_2, F_3, ... \leftarrow \texttt{nondominated-sort}(P_t \cup Q_t)$ $i \leftarrow 1$ **while** $|P_{t+1} \cup F_i| < \mu$ **do**

$P_{t+1} = P_{t+1} \cup F_i$ $i \leftarrow i + 1$ $\tilde{F}_i = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k) = \texttt{SortByCrowdingDistance}(F_i)$ **while** $P_{t+1} < \mu$ **do**

$P_{t+1} = P_{t+1} \cup \mathbf{x}_j$ $j \leftarrow j + 1$ Generate Q_{t+1} by binary tournament selection, recombination and mutation Stop criterion fulfilled

Lecture Overview

1. Solvers
2. Evolutionary multi-objective optimization algorithms (EMOA)
3. SMS-EMOA

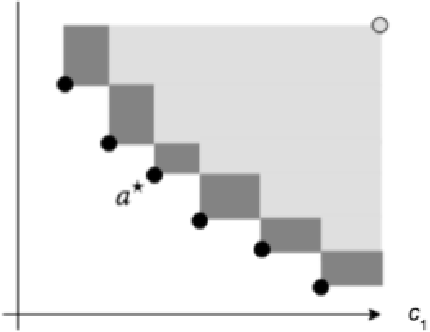
Selection criteria: contribution to the hypervolume I

- The SMS-EMOA (S-Metric-Selection-EMOA) evaluates the fitness of an individual $\mathbf{x} \in \mathcal{P} \subset \mathcal{X}$ based on its contribution to the dominated hypervolume (S-Metric):

$$\Delta s(\mathbf{x}, \mathcal{P}) = S(\mathcal{P}, R) - S(\mathcal{P} \setminus \{\mathbf{x}\}, R).$$

Selection criteria: contribution to the hypervolume II

Hypervolume contribution in a 2-dimensional objective space:



- Dark rectangles correspond to the hypervolume contribution of the black dots.
- Grey point is the so-called reference point and limits the space.

Selection criteria: contribution to the hypervolume III

- The hypervolume contribution thus corresponds to the size of the space that is dominated only by the individual $\boldsymbol{a}$, and not to any other of the space.
- $a^{\star}$ has lowest S-metric contribution .

SMS-EMOA algorithm I

Algorithm 3 SMS-EMOA

Generate start population P_0 of size μ $t \leftarrow 0$ **repeat**

| G

until

enerate **one** individual $\mathbf{q} \in^n$ by recombination and mutation of $\mathcal{P}_t$ $\{F_1, ..., F_k\} \leftarrow$ fast-dominated-sort$(P_t \cup \mathbf{q})$

$\boldsymbol{a}^\star \leftarrow$ argmin$_{\boldsymbol{a} \in F_k} \Delta s(\boldsymbol{a}, F_k)$ $P_{t+1} \leftarrow (P_t \cup \{\mathbf{q}\}) \setminus \{\boldsymbol{a}^\star\}$ $t \leftarrow t + 1$ Termination criterion fulfilled

- L5: the set of temporary $(\mu + 1)$ individuals is divided by **fast-dominated-sort** into k fronts $F_1, ..., F_k$.
- L6: determine individual $\boldsymbol{a}^\star \in F_k$ with smallest hypervolume contribution.
- L7: the individual $\boldsymbol{a}^\star$ from the worst front with the smallest contribution to the dominated hypervolume does not survive.
- The fitness of an individual is therefore primarily the rank of its associated front and secondarily its contribution to hypervolume.