An Introduction to Multiobjective Optimization

Sébastien Verel

LISIC
Université du Littoral Côte d'Opale
Equipe OSMOSE

verel@univ-littoral.fr
http://www.lisic.univ-littoral.fr/~verel

Master informatique WeDSci, ULCO,

2022, version 0.2

Single Objective Optimization

Inputs

- Search space: Set of all feasible solutions,

$$\mathcal{X}$$

- Objective function: Quality criterium

$$f : \mathcal{X} \rightarrow \mathbb{R}$$

Goal

Find the best solution according to the criterium

$$x^\star = \text{argmax } f$$

But, sometime, the set of all best solutions, good approximation of the best solution, good 'robust' solution...

Context

Black box Scenario

We have only $\{(x_0, f(x_0)), (x_1, f(x_1)), ...\}$ given by an "oracle"
No information is either not available or needed on the definition of objective function

- Objective function given by a computation, or a simulation
- Objective function can be irregular, non differentiable, non continous, etc.
- (Very) large search space for discrete case (combinatorial optimization), *i.e.* NP-complete problems
- Continuous problem, mixt optimization problem

Real-world applications

Typical applications

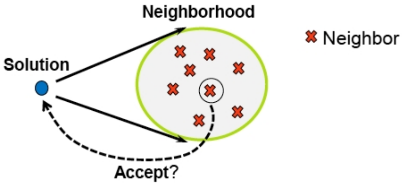
- Large combinatorial problems:
  Scheduling problems, planing problems, DOE,
  "mathematical" problems (Firing Squad Synchronization Pb.), etc.
- Calibration of models:
  Physic world ⇒ Model(params) ⇒ Simulator(params)
  $$\text{Model(Params)} = \operatorname{argmin}_M \operatorname{Error}(Data, M)$$
- Shape optimization:
  Design (shape, parameters of design)
  using a model and a numerical simulator

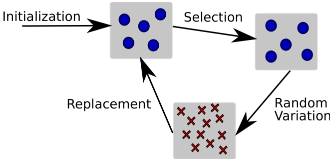
Search algorithms

Principle

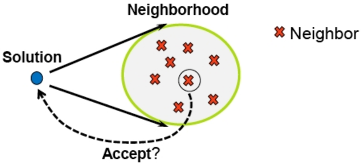
Enumeration of the search space

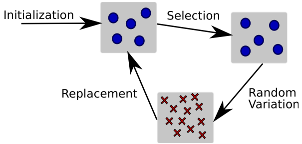
- A lot of ways to enumerate the search space
- Using random sampling: Monte Carlo technics
- Local search technics:





Search algorithms





- **Single solution-based**: Hill-climbing technics, Simulated-annealing, tabu search, Iterative Local Search, etc.
- **Population solution-based**: Genetic algorithm, Genetic programming, ant colony algorithm, etc.

Design components are well-known

- Probability to decrease,
- Memory of path, of sub-space
- Diversity of population, etc.

Research question: Parameters tuning

- One Evolutionary Algorithm key point:
  Exploitation / Exploration tradeoff
- One main practical difficulty:
  Choose operators, design components, value of parameters, representation of solutions
- Parameters setting (Lobo et al. 2007):
  - Off-line before the run: *parameter tuning*,
  - On-line during the run: *parameter control*.

One practical and theoretical question

How to combine correctly the design components according to the problem (in distributed environment...) ?

Research question: Expensive optimization

- Objective function based on a simulation:
  Expensive computation time
- One main practical difficulty:
  With few computation evaluation, choose operators, design components, value of parameters, ...
- Two main approaches:
  - Parallel computation: *distributed computing*.
  - Compute a model of function: *surrogate model*,

One practical and theoretical question

How to combine correctly the design components
with low computational budget
according to the problem in distributed environment... ?

How to solve a multi-criterium problem

Think about the decision problem!

1. Define decision variables
2. Define objective functions (criteria)
3. Define your goal: *a priori*, or *a posteriori*
4. Use an (optimization) algorithm
5. Analyze the result

A priori goal

A priori decision

Decision maker knows what he/she wants before optimization

Weighted sum

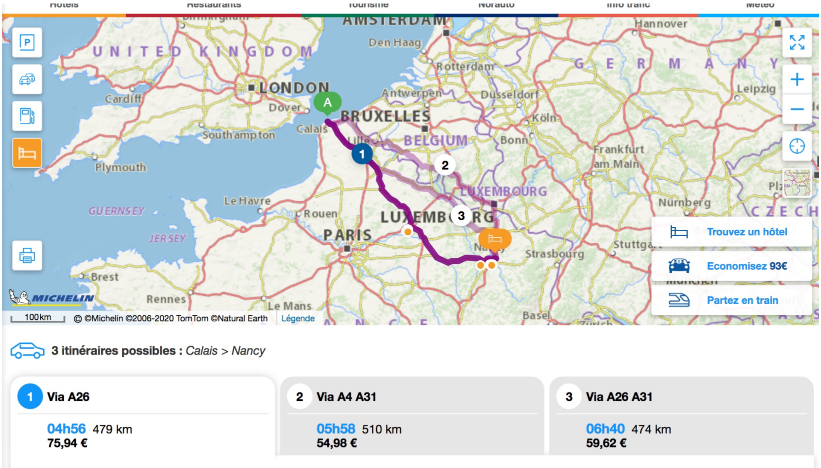
$$f_\lambda(x) = \lambda_1 f_1(x) + \ldots + \lambda_m f_m(x)$$

with $\lambda_i > 0$

- Basic model
- Common technique
- Convert a multiobjective problem into a single-objective problem
- The definition, and the interpretation are not always straitforward

Small example

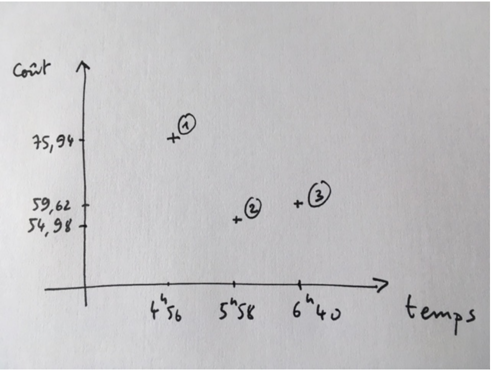
Road trip between Calais and Nancy



Which one is better ?

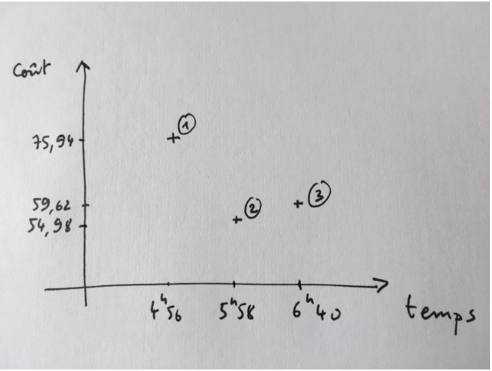
Small example

Road trip between Calais and Nancy



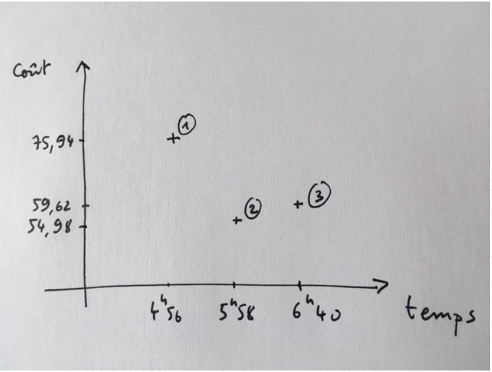
Small example

Road trip between Calais and Nancy



- According to time objective, 1 is better
- According to cost objective, 2 is better
- But, 2 is better than 3 for both objectives.

Pareto dominance



- 1 and 2 are incomparable
- 1 and 3 are incomparable
- 2 is better than 3

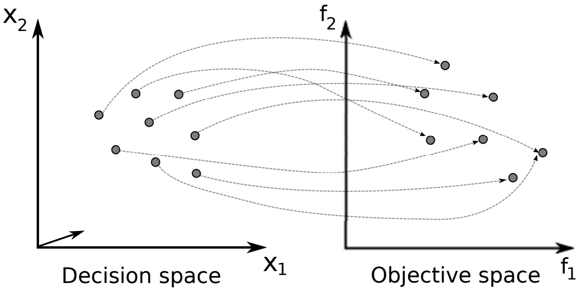
Pareto dominance

- 2 dominates 3
- 3 is dominated by 2

Multiobjective optimization

Multiobjective optimization problem

- $\mathcal{X}$: set of feasible solutions in the decision space
- $M \geqslant 2$ objective functions $f = (f_1, f_2, \ldots, f_M)$ (to maximize)
- $\mathcal{Z} = f(\mathcal{X}) \subseteq \mathbb{R}^M$: set of feasible outcome vectors in the objective space

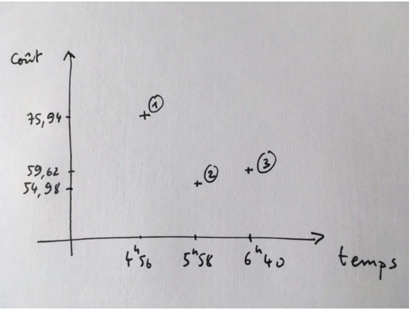


Pareto dominance definition

Pareto dominance relation (maximization)

A solution $x \in \mathcal{X}$ dominates a solution $x' \in \mathcal{X}$ $(x' \prec x)$ iff

- $\forall i \in \{1, 2, \ldots, M\}$, $f_i(x') \leqslant f_i(x)$
- $\exists j \in \{1, 2, \ldots, M\}$ such that $f_j(x') < f_j(x)$

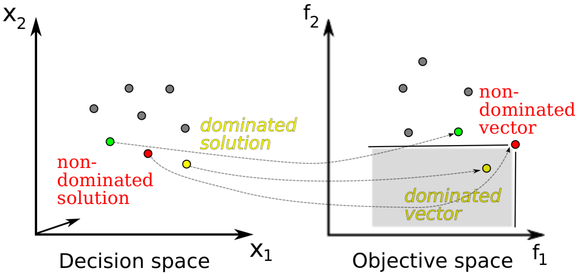


Pareto Optimale solution

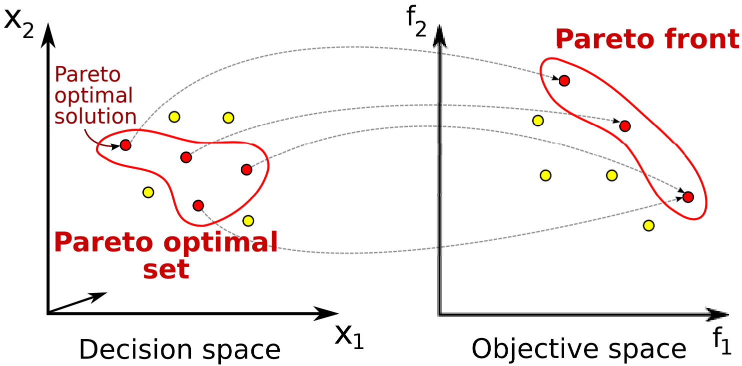
Definition: non-dominated solution

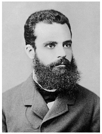
A solution $x \in \mathcal{X}$ is non-dominated (or Pareto optimal, efficient) iff

$$\forall x' \in \mathcal{X} \setminus \{x\},\ x \not\prec x'$$



Pareto set, Pareto front



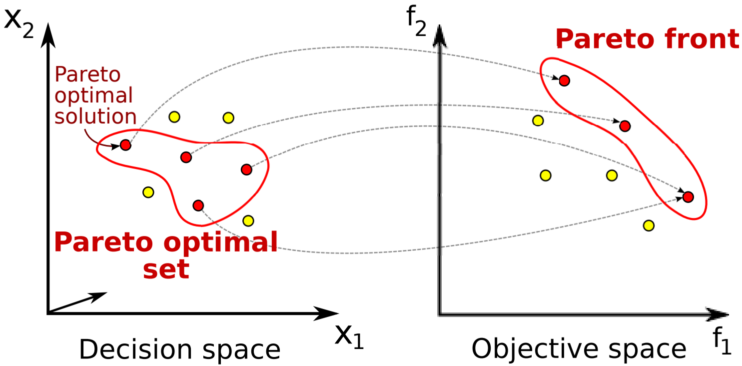


Vilfredo Pareto (1848 - 1923)
source: wikipedia

Multiobjective optimization goal

Goal

Find the Pareto Optimal Set,
or a good approximation of the Pareto Optimal Set
And not a single solution for a single aggregated objective



How to compute non dominated solutions from a set?

Filter by dominance relation with a basic algorithm: see exercice 2

Input: solution_set, the set of solutions to filter by dominance
Output: non_dominated_solutions, the set of non-dominated solutions

```
non_dominated_solutions ← ∅
for solution ∈ solution_set do
    s ← first solution of solution_set
    while s ≠ NULL && solution is not dominated by s do
        s ← next solution of solution_set
    end while
    if  s = NULL then
        non_dominated_solutions ← non_dominated_solutions ∪{ solution }
    end if
end for
return  non_dominated_solutions
```

Time complexity: $\mathcal{O}(m^2 \times d)$
where m is the size of solutions set, and d the dimension of objective space

Challenges

- **Search space**:
  many variables, heterogeneous, dependent variables
- **Objective space**:
  many, heterogenous, expensive objective functions
- **NP-completeness**:
  deciding if a solution is Pareto optimal is difficult
- **Intractability**:
  number of Pareto optimal solutions grows exponentially with problem dimension

Why multiobjective optimization?

Position of multiobjective optimization: Decision making

- No **a priori** on the importance/weights of the different objectives
- **a posteriori** selection by a decision marker:
  Selection of one Pareto optimal solution after a deep study of the possible solutions.

Methodology

Typical methodology with MO optimization

1. Define **decision variables**
2. Define all potential **objective**
3. Define **constraints** (soft/hard/objective)
4. Choose/design a relevant multiobjective **algorithm**
5. Search for an approximation of **Pareto optimal** solutions set
6. **Analyse/visualize** the solutions set

Loop between 1 to 6...

Multi-objective, many-objective optimization

Approximative definition

- Multi-objective: 2, 3 or 4 objectives
- Many-objective: 4, 5 and more objectives

Number of Pareto optimal solutions

Suppose that:

- Probability to improve: p (for all objective),
- Objective are independent.

Probability to be non-dominated for M objectives is:

Multi-objective, many-objective optimization

Approximative definition

- Multi-objective: 2, 3 or 4 objectives
- Many-objective: 4, 5 and more objectives

Number of Pareto optimal solutions

Suppose that:

- Probability to improve: p (for all objective),
- Objective are independent.

Probability to be non-dominated for M objectives is: $1 - (1 - p)^M$

Multi-objective, many-objective optimization

Approximative definition

- Multi-objective: 2, 3 or 4 objectives
- Many-objective: 4, 5 and more objectives

Number of Pareto optimal solutions

Suppose that:

- Probability to improve: p (for all objective),
- Objective are independent.

Probability to be non-dominated for M objectives is: $1 - (1 - p)^M$

Intuitive goals

Convergence toward the front, and diversity of the solutions.
Many-objective: convergence "easy", diversity "hard"

Note: objective correlation is also important.

Performance assessment

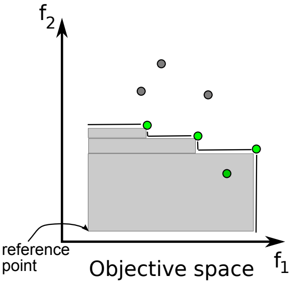
Quality of the approximation of the Pareto front [1]

- Goal: indicator function related to the quality of the set.
- No universal indicator

Indicator functions

- Hypervolume indicator
- Epsilon indicator
- Inverted Generational Distance (IGD)
- Atteinment function

Hypervolume ($\mathtt{I}_H$)



Volume cover
by the set.

Properties

- Compliant with the (weak) Pareto dominance relation
  $\rightarrow A \prec B \Rightarrow \mathtt{I}_H(A) \leqslant \mathtt{I}_H(B)$
- A single parameter:
  the reference point
- Minimal solution-set maximizing $\mathtt{I}_H$
  $\rightarrow$ subset of the Pareto optimal set

$$\arg\max_{\sigma \in \Sigma} \ \mathtt{I}_H(\sigma)$$

Slide to draw

(additive) Epsilon indicator ($\mathtt{I}_\epsilon$)

Definition

Smallest coefficient ϵ to translate the set A to "cover" each point of the set B

For maximization, minimal ϵ value such that:
$\forall z_b \in B,\ \exists z_a \in A$ such that $z_b \prec z_a + \epsilon$

Properties

- Compliant with the (weak) Pareto dominance relation (using Pareto front)
  $\rightarrow A \prec B \Rightarrow \mathtt{I}_\epsilon(A) \leqslant \mathtt{I}_\epsilon(B)$
- A parameter: the reference set

Slide to draw

Inverted Generational Distance (IGD)

Definition: Generational Distance (GD) of set A

Average over all solutions $a \in A$ of the distance between solution a and the closest solution in a reference set R:

$$IG(A, R) = \frac{1}{|A|} \sum_{a \in A} \min_{r \in R} \text{dist}(a, r)$$

where $\text{dist}(a, r)$ is the euclidian distance in objective space between solution a, and r.

Definition: Inverted Generational Distance (IGD)

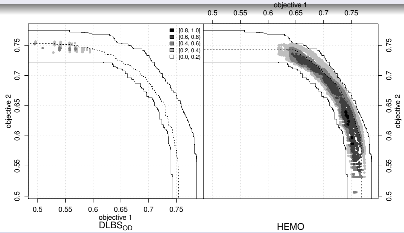
$$IGD(A, R) = IG(R, A)$$

Slide to draw

Atteinment function

Definition

Probability to reach a point in objective space



Tools

Empirical Attainment Function (EAF) Tools, Manuel López-Ibáñez: `https://mlopez-ibanez.github.io/eaf/`

Manuel López-Ibáñez, Luís Paquete, and Thomas Stützle. Exploratory Analysis of Stochastic Local Search Algorithms in Biobjective Optimization. In Experimental Methods for the Analysis of Optimization Algorithms, 2010.

Carlos M Fonseca, Joshua D Knowles, Lothar Thiele, and Eckart Zitzler.
A tutorial on the performance assessment of stochastic multiobjective optimizers.
In *Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, volume 216, page 240, 2005.