Multi-objective optimization and Symbolic Regression

Classical and meta-heuristic methods

Veronica Guidetti 14-22/02/2024, UNIMORE

Table of contents

Four lectures course: 4x2 hrs

Lecture 1:

- Introduction to MOO framework and taxonomy
- Classical scalarization methods

Lecture 2 and 3:

- Meta-heuristic methods
- Simulated annealing
- Swarm Particle
- Ant Colony

Lecture 3:

- Genetic algorithms
- Evaluation metrics

Evaluation metrics

Selection and convergence

Until now, we studied ways to get optimal solutions or approximations of the first Pareto front.

We still need to answer the following questions:

- How can we evaluate the convergence of our algorithm?

- How do we point out the best solution in the absence of DM??

Algorithm convergence/performance indicators

Performance indicators

Aim to provide a synthetic measure of the quality of solutions that should satisfy:

- The **distance between the Pareto front and its representation** in the objective space should be minimized.
- A good (according to some metric) **distribution** of the points of the corresponding approximated front in the objective space is desirable.
- The **extent** of the corresponding approximated front should be maximized, i.e., for each objective, a wide range of values should be covered by the non-dominated points.

Range covering indicator *(Shott, 1995)*

This indicator measures how well the possible range of the Pareto front is covered by the points in an approximation set A.

$$I_R(A) = \frac{1}{M} \sum_{k=1}^{M} (R_k^{max}(A) - R_k^{min}(A))$$

$$R_k^{max}(A) = \max_{z \in A} f_k(z)\,, \qquad R_k^{min}(A) = \min_{z \in A} f_k(z)$$

Larger values imply that A better covers the extreme solution in the objective space.

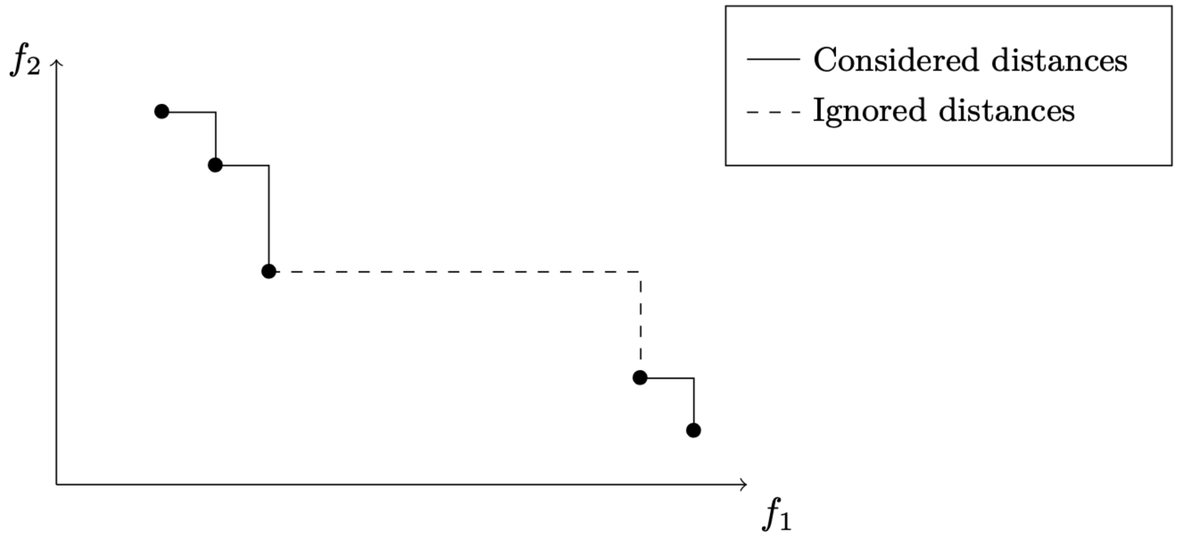
Spacing indicator *(Shott, 1995)*

It measures the spread of the points of the approximation set in the objective space.

$$I_S = \sqrt{\frac{1}{|A|-1}\sum_{z\in A}\left(\bar{D} - D(z)\right)^2}$$

$$D(z) = \min_{z'\in A\,,\, z'\neq z}\sum_{k=1}^{K} ||\vec{f}(z) - \vec{f}(z')||_1$$

$$\bar{D} = \frac{1}{|A|}\sum_{z\in A}\vec{f}(z)$$



The smaller the value, the better the distribution of the solutions of the Pareto frontier.

This method cannot account for holes in the Approximation set.

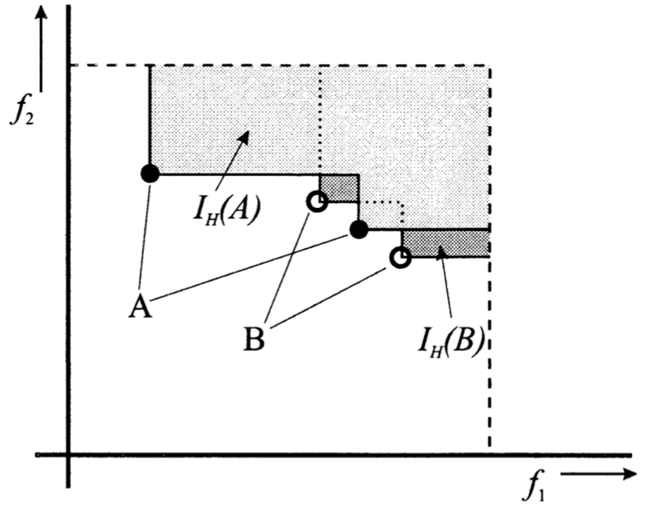
Audet, C., Bigeon, J., Cartier, D., Le Digabel, S., & Salomon, L. (2021). Performance indicators in multiobjective optimization. European journal of operational research, 292(2), 397-422.

Comparing different sets

Most indicators aim to compare the approximation set distribution with a discrete version of the Pareto front.

The Pareto front may be known (for benchmark problems) or should be approximated with multiple optimization runs.

We can extend the definition of dominance to sets of solutions to compare different approximation sets or the approximation and reference sets.

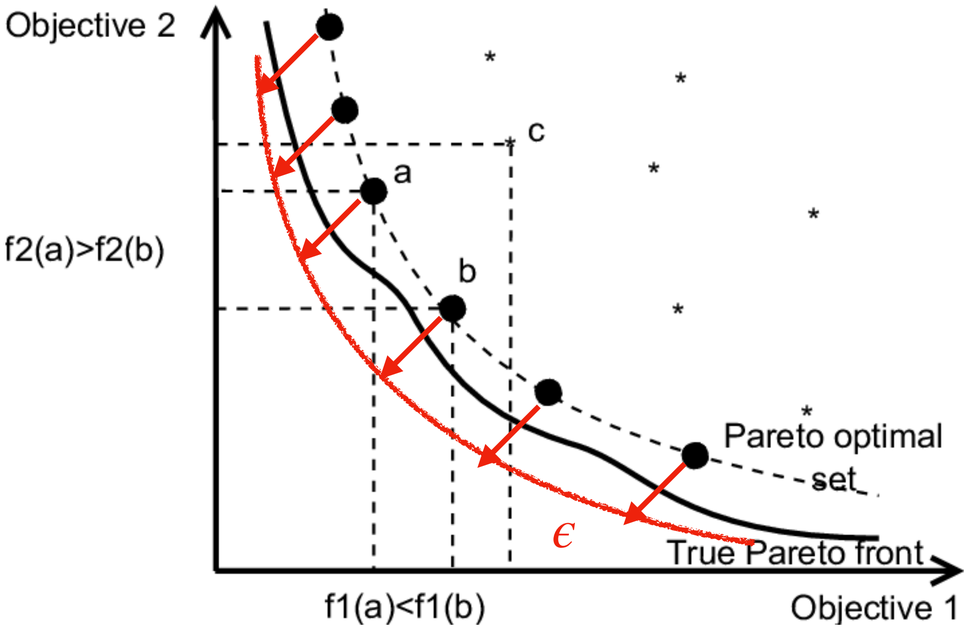


- A set S_1 **weakly dominates** a set S_2 if $\forall s_2 \in S_2 \exists s_1 \in S_1 \quad s.t. \quad s_1 \preceq s_2$ and there exists at least a point in S_1 that is not contained in S_2.
- A set S_1 **strongly dominates** a set S_2 if $\forall s_2 \in S_2 \exists s_1 \in S_1 \quad s.t. \quad s_1 \preceq s_2$ and there exists at least a point in S_1 that strongly dominates some points in S_2.
- A set S_1 **completely dominates** a set S_2 if $\forall s_2 \in S_2 \exists s_1 \in S_1 \quad s.t. \quad s_1 \prec s_2$.

Unary ϵ-indicator *(Zitzler, 2003)*

Given a reference (Pareto optimal) set $P \in R^M$ and an approximate set $A \in R^M$ the multiplicative unary ϵ-indicator computes the minimum factor such that when every point in A gets multiplied by the ϵ then the resulting set weakly dominates the reference set P.

$$I_\epsilon(A, P) = \sup_{\epsilon \in R} \{ \forall z \in P \, \exists x \in A \; s.t. \; \epsilon x \succeq z \}$$



Pareto frontier approximation indicator *(Czyzak and Jaszkiewicz, 1996)*

This indicator measures the distance between A and P in the objective space.

It is given by:

$$I_A(A, P) = \frac{1}{|P|} \sum_{r \in P} \left(\min_{z \in A} D(z, r) \right)$$

where D is some distance metric in the objective space (such as Euclidean distance).

Lower values imply better approximation.

Inverted Generational Distance *(Coello, 2005)*

This indicator measures the average minimal distance between an element of a discretized PF representation to the closest point of the approximation set A.

$$IGD(P, A) = \frac{1}{|P|}\left(\sum_{y_p \in P}\left(\min_{y_a \in A} d(y_a, y_p)\right)^p\right)^{1/p} \qquad d(y_a, y_p) = ||y_a - y_p||_2$$

Smaller values imply that A provides a better approximation of the PF.

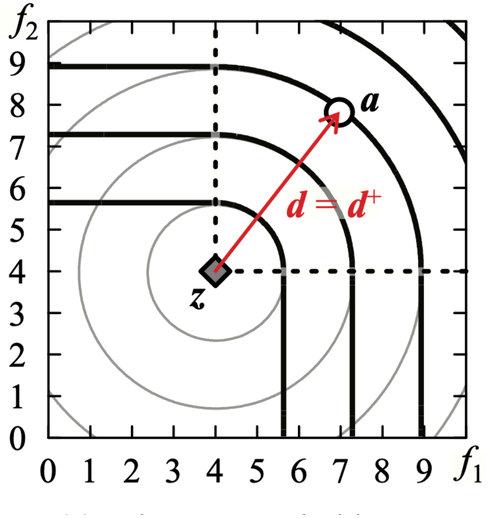
Problem: the distance $d(y_a, y_p)$ cannot be viewed as an amount of the inferiority to be minimized when y_a and y_p are non-dominated with each other.

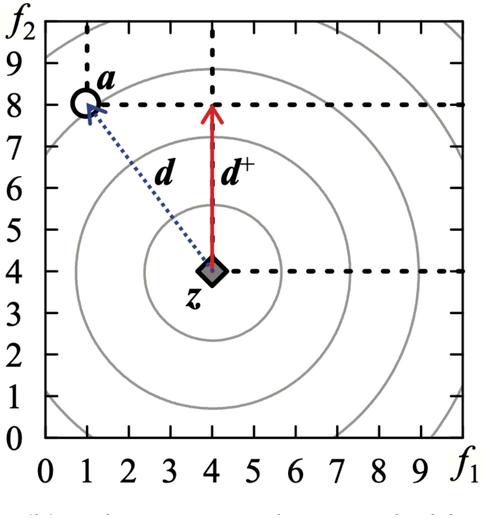
Modified Inverted Generational Distance *(Ishibuchi, 2015)*

This modified indicator takes into account the domination relation between y_a and y_p. In practice, this is done by modifying the definition of distance

$$IGD^+(P,A) = \frac{1}{|P|}\left(\sum_{y_p \in P}\left(\min_{y_a \in A} d_+(y_a, y_p)\right)^p\right)^{1/p}$$

$$d_+(y_A, y_p) = \sqrt{\sum_{i=1}^{M} d_{+i}^2} \qquad d_{+i} = \max\{(y_A - y_p)_i, 0\}$$





Smaller values imply that A provides a better approximation of the PF.

IGD^+ is **weakly Pareto compliant**, meaning that

$$\forall A_1, A_2 \quad s.t. \quad A_1 \preceq A_2 \implies IGD^+(A_1, P) \leq IGD^+(A_2, P)$$

Hypervolume indicator *(Zitzler and Thiele, 1999)*

Given a point set $S \in R^M$ and a reference point $r \in R^d$ the hypervolume indicator of S is the measure of the region weakly dominated by S and bounded by r:

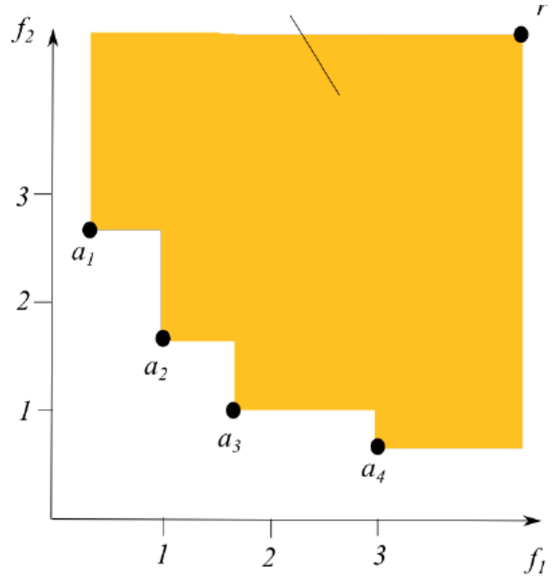
$$H(S) = \mathscr{L}(\{q \in R^d \mid \exists p \in S : p \preceq q \cap q \preceq r\})$$

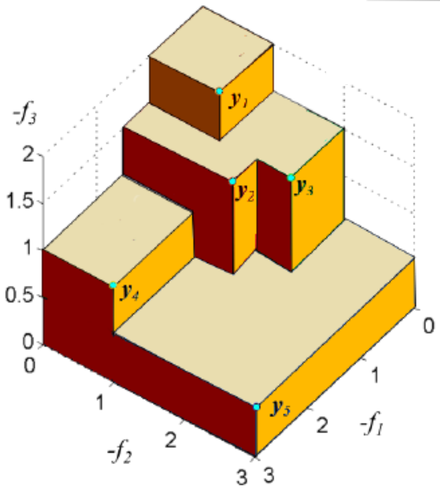
Or, alternatively

$$H(S) = \mathscr{L}\left(\bigcup_{p \in S, p \preceq r} Hypercube[p, r]\right)$$

The Hypervolume indicator is weakly Pareto compliant.

Rescale objectives or live with distortions!





R3 indicator *(Hansen and Jaszkiewicz, 1998)*

Used to compare the approximation sets of different methods or runs.

We generate a set Λ of $|\Lambda|$ uniformly dispersed weight vectors: $\overrightarrow{\lambda} = \{\lambda_1, \ldots, \lambda_M\}$

We fix the utopian point z^* so that it weakly dominates all the points in the approximation set.

$$I_{R3}(A,R) = \frac{1}{|\Lambda|} \sum_{\overrightarrow{\lambda}} \frac{u^*(\overrightarrow{\lambda}, A) - u^*(\overrightarrow{\lambda}, R)}{u^*(\overrightarrow{\lambda}, R)} \quad \text{where} \quad u^*(\overrightarrow{\lambda}, A) = \max_{z \in A} u_\lambda(z)$$

and $$u_\lambda(z) = \left(\max_{j \in 1\ldots,M} \lambda_j \cdot |z_j - z_j^*| + \rho \cdot \sum_{j \in 1,\ldots,M} |z_j - z_j^*| \right); \quad \rho \ll 1$$

This is the augmented *Tchebycheff* scalarization. Better approximation gets values near zero.

It captures how well A performs on average, considering all possible tradeoff weightings in the chosen distribution.

Basically, everything you would like to know about performance indicators in MOO can be found in

“Performance indicators in multiobjective optimization, Audet at al. 2020”

https://www.sciencedirect.com/science/article/abs/pii/S0377221720309620

Model selection indicators

Solution estimators

Without a DM, after running optimization algorithms producing multiple solutions that approximate the Pareto frontier, we need to define methods to select a single best individual or rank non-dominated individuals. The methods can be diverse, but some have better properties.

The same is needed in the presence of DM after running an MOO algorithm providing a set of optimal solutions.

Angle-based knee point method (Deb, 2011)

The **knee point** is the point of maximal curvature on the Pareto Front.
This means that a small improvement in one of the objectives causes a large deterioration in the others.

Developed for bi-objective optimization.

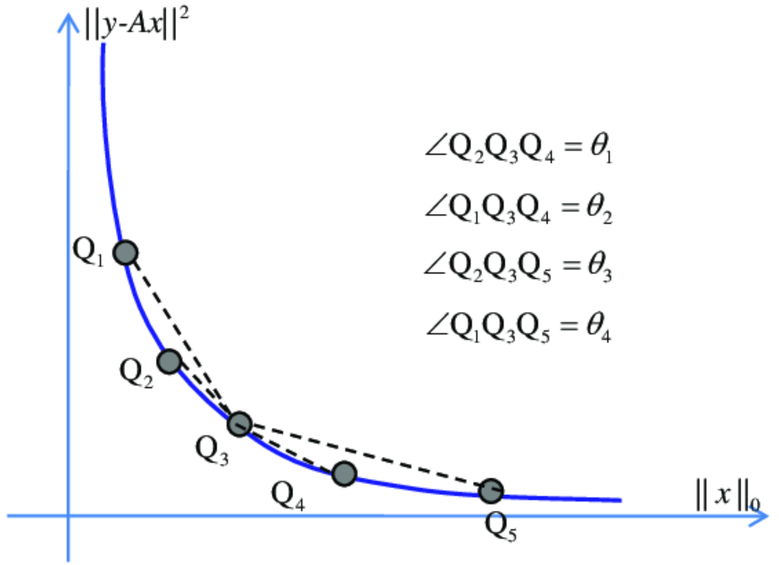
Given the approximation set $A = \{A_1, A_2, \ldots, A_K\}$

Compute the reflex angle within point triplets (A_{i-1}, A_i, A_{i+1})

$$\alpha_i = \arccos\left(\frac{\overrightarrow{(A_{i+1} - A_i)} \cdot \overrightarrow{(A_{i-1} - A_i)}}{|(A_{i+1} - A_i)| \cdot |(A_{i-1} - A_i)|}\right) \qquad i = 2,\ldots, K-1$$

The best solution is the one corresponding to A_j s.t. $\alpha_j = \min_i \alpha_i$

Use more than 2 NN points to improve the stability of results.



Deb, 2011: Finding Knees in Multi-objective Optimization

Marginal Utility Knee point method

Let us assume to consider a linear utility function

$$U(\mathbf{x}, \lambda) = \lambda f_1(\mathbf{x}) + (1 - \lambda) f_2(\mathbf{x})$$

where different λ correspond to different DMs.

We sample a set of possible weights $\Lambda = \{\lambda_i\}$ following a uniform distribution (sample on the simplex).

For each point $x \in A$ we can compute the utility function associated to $\lambda_i \in \Lambda$.

$$U(x_\alpha, \lambda_j) = \begin{cases} \min_{\beta \neq \alpha} U(x_\beta, \lambda_j) - U(x_\alpha, \lambda_j) & \text{if} \quad \alpha = \text{argmin} U(x_j, \lambda_j) \\ 0 \qquad \text{otherwise} \end{cases}$$

Based on this definition, the expected marginal utility (EMU) is calculated as an integral of the individual marginal utility function across a set of different weight vectors. Solution(s) having the largest EMU is(are) recognized as the knee point(s).

Marginal Utility Knee point method

The marginal utility of a solution x_α is the expected performance depletion a DM would face if x_α is removed from A

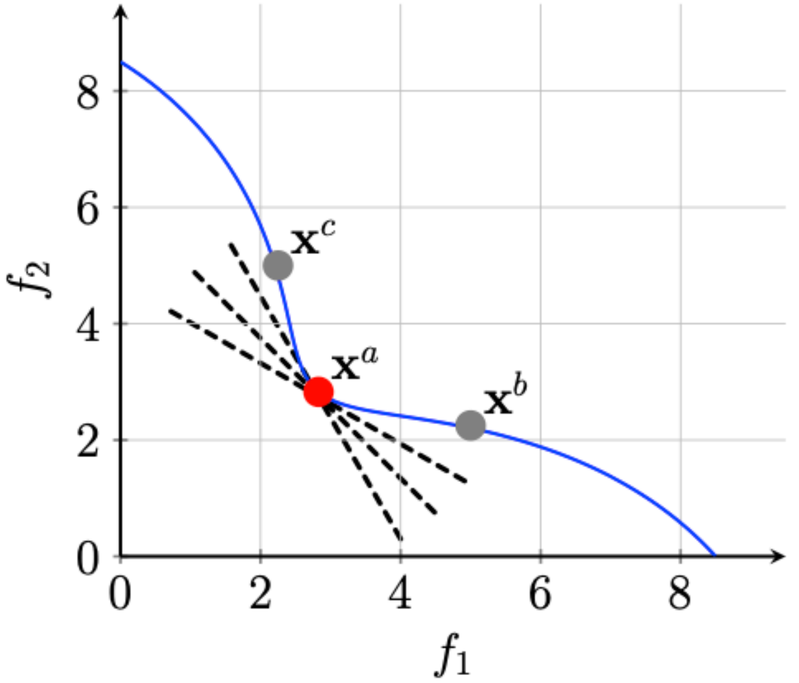
If we consider a bi-objective problem, we can:

Sort solutions according to f_1

We can define a weight λ_{ij} such that x_i and x_j

have the same utility:

$$\lambda_{ij} = \frac{f_2(x_j) - f_2(x_i)}{f_1(x_i) - f_1(x_j) + f_2(x_j) - f_2(x_i)}$$



And the expected marginal utility of the solution x_i is given by:

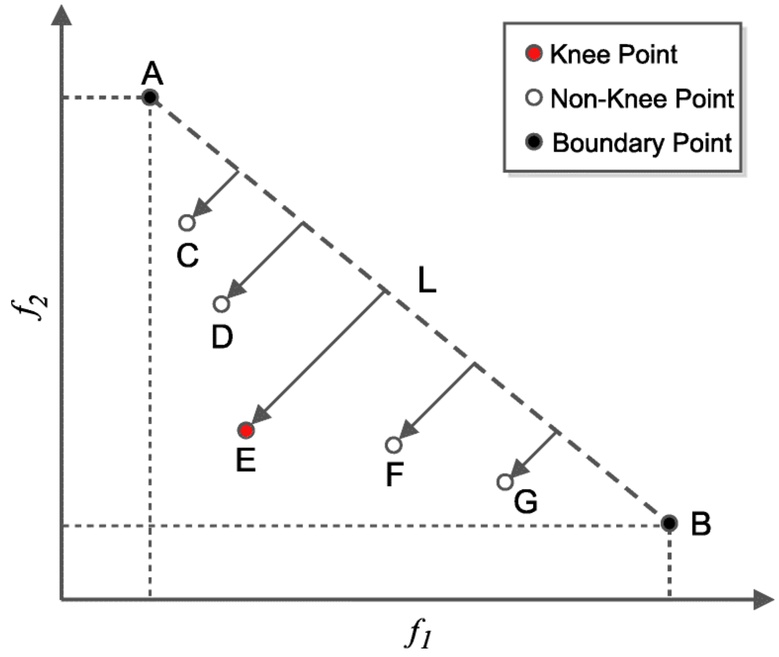
$$E(U(x_i,\lambda)) = \int_{\lambda_{i-1,i}}^{\lambda_{i-1,i+1}} (\alpha, 1-\alpha)\cdot \overrightarrow{\delta f}_{i_-} d\alpha + \int_{\lambda_{i-1,i+1}}^{\lambda_{i,i+1}} (\alpha, 1-\alpha)\cdot \overrightarrow{\delta f}_{i_+} d\alpha \qquad \overrightarrow{\delta f}_{i_-} = \left(\vec{f}(x_i) - \vec{f}(x_{i-1})\right) \qquad \overrightarrow{\delta f}_{i_+} = \left(\vec{f}(x_i) - \vec{f}(x_{i+1})\right)$$

Hyperplane normal vector knee point method

Rescale objectives to lie in [0,1] using nadir and ideal points.

Identify the hyperplane going through the anchor solutions $z_m^{i*} = \delta_{im}$,
i.e., solutions minimizing one objective at a time: $\sum_i \overrightarrow{z_i} = \overrightarrow{1}$.

Method 1) solution $\mathbf{x}$ in the set $S = \{x \,|\, \vec{f}(\mathbf{x}) \in A\}$ having the largest perpendicular distance from the hyperplane are selected. (Das, 1999)



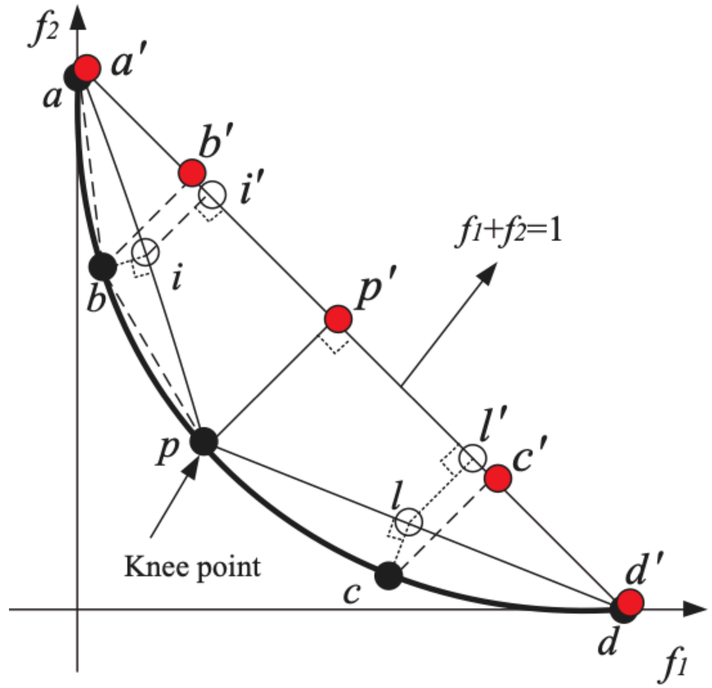
Hyperplane normal vector knee point method

Rescale objectives to lie in [0,1] using nadir and ideal points.

Identify the hyperplane going through the anchor solutions $z_m^{i*} = \delta_{im}$, i.e., solutions minimizing one objective at a time: $\sum_i \overrightarrow{z_i} = \overrightarrow{1}$.

Method 1) solution $\mathbf{x}$ in the set $S = \{x \mid \vec{f}(\mathbf{x}) \in A\}$ having the largest perpendicular distance from the hyperplane are selected. (Das, 1999)

Method 2) Project solutions onto the anchor hyperplane. The **density** of each solution is calculated based on the harmonic mean distance between the solution and its k neighbors. Then, the solutions are **clustered** (k'<k) according to the solution density, each group representing a candidate knee region. The convexity of each candidate knee group will be determined. The groups with a large curvature, having a large difference between the **radial coordinate value** of the knee point and that of its neighbors, are kept. (Yu, 2018).



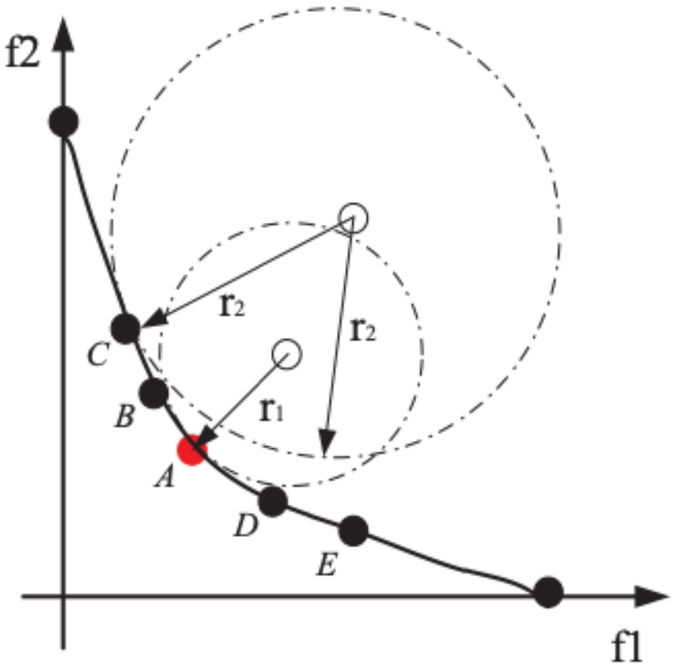
Hyperplane normal vector knee point method

Rescale objectives to lie in [0,1] using nadir and ideal points.

Identify the hyperplane going through the anchor solutions $z_m^{i*} = \delta_{im}$, i.e., solutions minimizing one objective at a time: $\sum_i \overrightarrow{z_i} = \overrightarrow{1}$.

Method 1) solution $\mathbf{x}$ in the set $S = \{x \mid \vec{f}(\mathbf{x}) \in A\}$ having the largest perpendicular distance from the hyperplane are selected. (Das, 1999)

Method 2) Project solutions onto the anchor hyperplane. The **density** of each solution is calculated based on the harmonic mean distance between the solution and its k neighbors. Then, the solutions are **clustered** (k'<k) according to the solution density, each group representing a candidate knee region. The convexity of each candidate knee group will be determined. The groups with a large curvature, having a large difference between the **radial coordinate value** of the knee point and that of its neighbors, are kept. (Yu, 2018).



Hypervolume method *(Zitzler, 2007)*

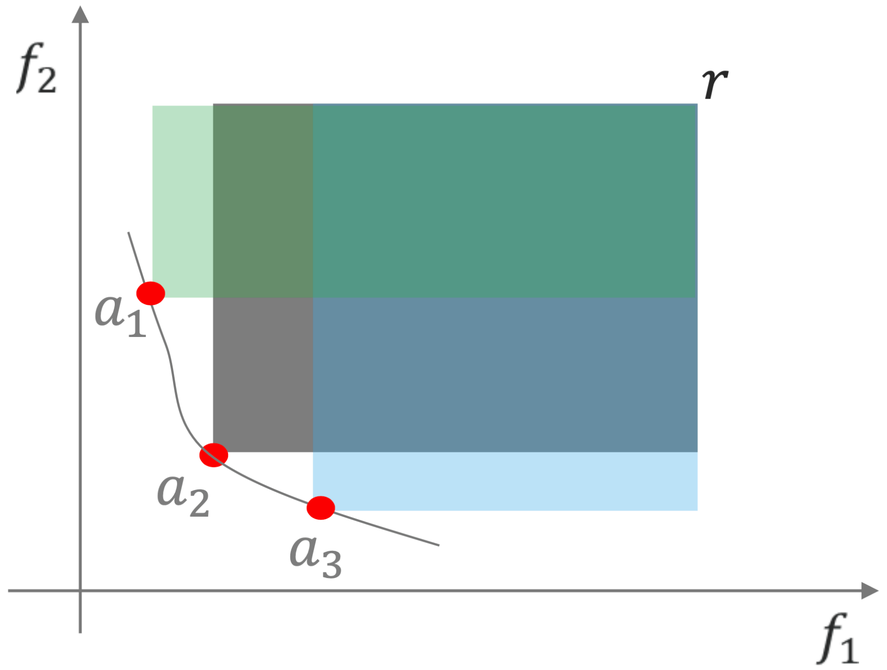
Given an approximation set A in the objective space

Define a reference point (worst case scenario)

$$r = (r_1, r_2, \ldots, r_M)$$

Evaluate the hypervolume of each solution $\vec{f}(x) \in A$

Select: y_a s.t. $a = \text{argmax}_{i \in A} H(\{y_i\})$



The ranking of the solutions according to the HV is invariant to linear global scalings of the objective functions!

Multiple-criteria decision making (MCDM) methods

MCDM methods are used when there are multiple conflicting criteria to be considered in making a decision.

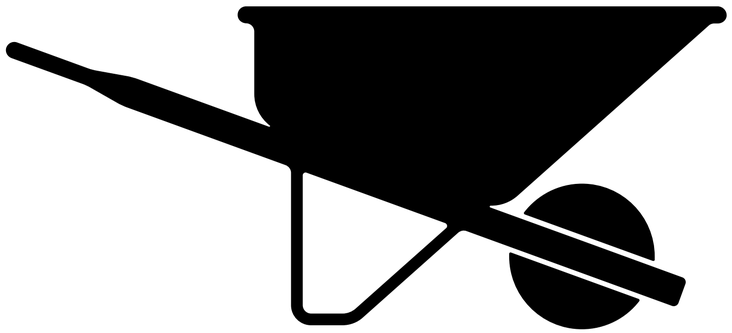
Technique for order of preference by similarity to ideal solution (TOPSIS) ***(Yoon, 1987)***
- Normalize the decision matrix (usually by standardizing the data).
- Calculate the weighted normalized decision matrix.
- Determine the ideal and negative ideal solutions based on the maximization or minimization of criteria.
- Compute the distances between each alternative and the ideal and negative ideal solutions.
- Rank the alternatives based on their proximity to the ideal solution and distance from the negative ideal solution.

Preference Ranking Organization METHod for Enrichment of Evaluations (PROMETHEE) (Brans and Vincke, 1985)
- Define preference functions that express the relative importance of criteria and the preference for each alternative.
- Compute the preference degrees for each pair of alternatives based on the preference functions.
- Aggregate the preference degrees to obtain a net preference degree for each alternative.
- Rank the alternatives based on their net preference degrees.

Least misery method:
- The Least Misery Method aims to minimize the maximum dissatisfaction or misery that could be experienced across all criteria.
- In this method, alternatives are ranked based on the minimum value of their worst performance across all criteria. This means that an alternative with the least severe dissatisfaction, even if it's not the best performer overall, may be preferred over alternatives with higher levels of dissatisfaction in one or more criteria.
- The Least Misery Method is suitable when the decision-maker wants to avoid extreme dissatisfaction with any particular criterion.

… many others …

Hands on



In need of a broadly accepted termination criterion for GP?

https://www.egr.msu.edu/~kdeb/papers/c2020003.pdf

Excursus on NSGA-III (Deb, 2013)

Use reference point methodology in GP and attempt to find a set of preferred Pareto-optimal solutions near the regions of interest to a decision-maker.

- The method is indifferent of the shape of the Pareto-optimal frontier.
- Applicable to many objectives, a large number of variables, and linear or non-linear constraints.
- Diversity among population members is aided by supplying several well-spread reference points.

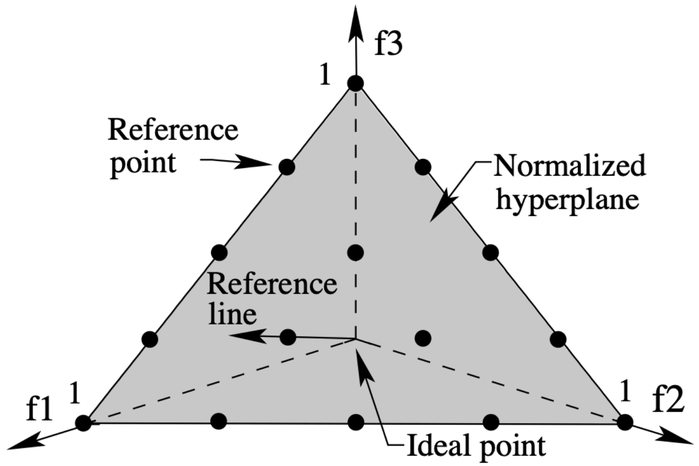
Disclaimer: parents are RANDOMLY selected

Preference lines creation:

- User defined preference conditions
- (Standard) Das and Dennis approach: uniform grid on a simplex with

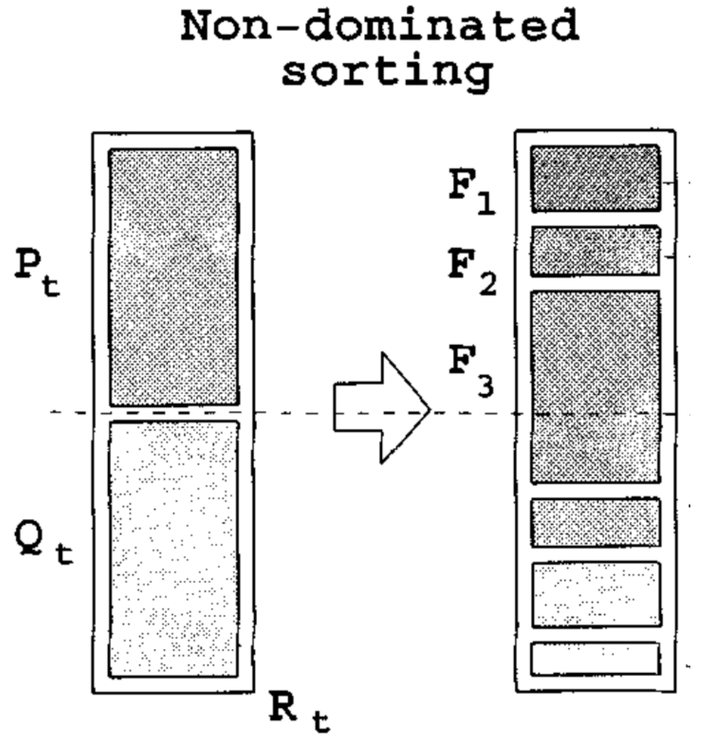
$$|\text{ref points}| = \binom{M+p+1}{p}$$

M=number of objectives; p= number of divisions on each simplex edge



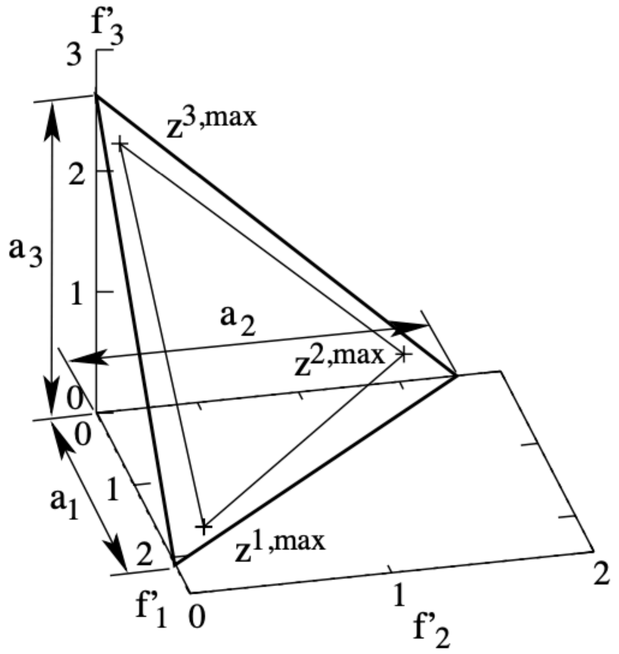
NSGA-III - Selection of next population

1. **Ranks** → select the Pareto fronts as long as they completely fit in the new population



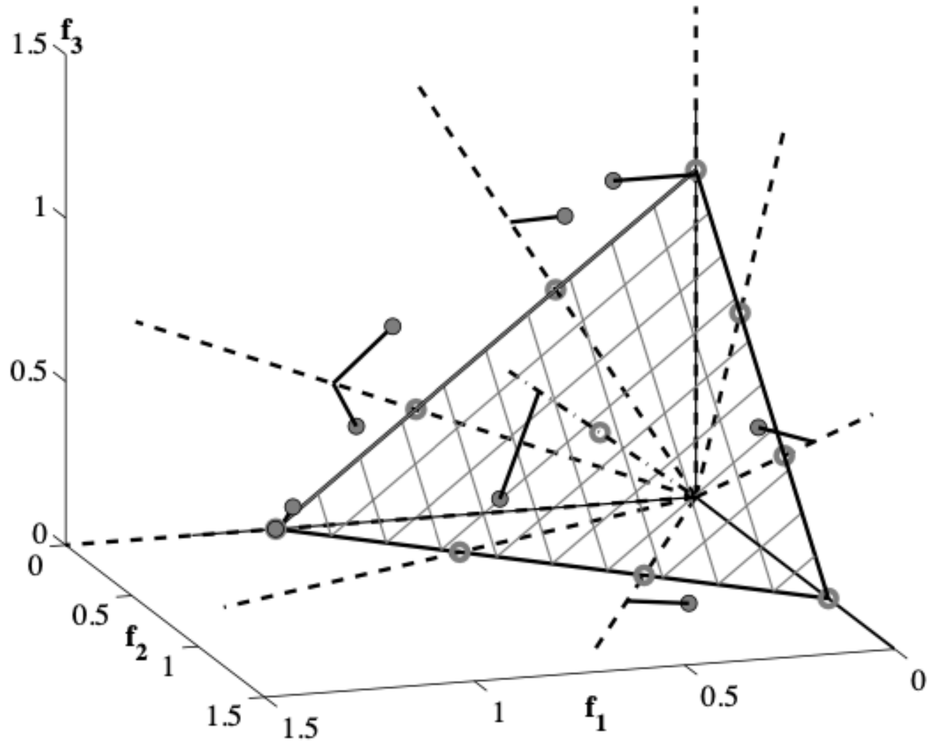
NSGA-III - Selection of next population

1. **Ranks** → select the Pareto fronts as long as they completely fit in the new population
2. **Normalize the objectives**:
   A. Compute ideal point $z_j^{\min} = \min_{\mathbf{s} \in S_t} f_j(\mathbf{s})$
   B. Compute extreme points $\mathbf{z}^{j,\max} = \mathbf{s} : \arg\min_{\mathbf{x} \in S_t} \{ \max_{i=1}^{M} (f_i(\mathbf{x}) - z_i^{\min}) / w_j^i \}$ where $w_j^k = \delta_{jk} + (1 - \delta_{jk})\,\epsilon$
   C. Compute normalized objectives $f_i^n(\mathbf{x}) = \dfrac{f_i(\mathbf{x}) - z_i^{\min}}{a_i - z_i^{\min}}$ where a_i are the intercepts of the hyperplane $\{z_j^{\max}\}$
   D. Map reference points on the normalized hyperplane (simplex)



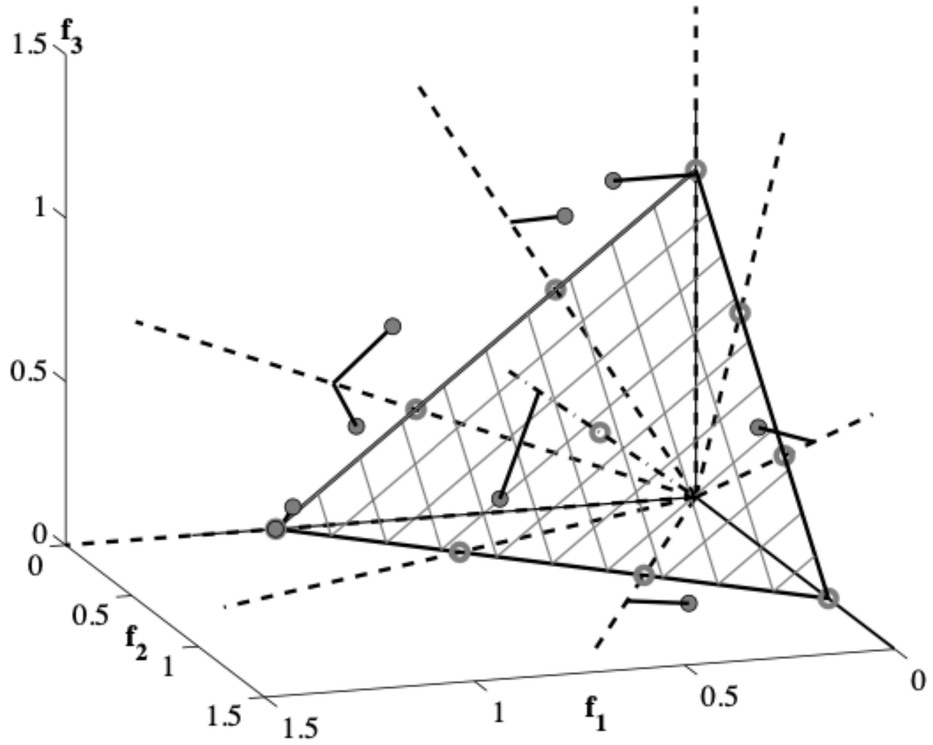
NSGA-III - Selection of next population

1. **Ranks** → select the Pareto fronts as long as they completely fit in the new population
2. **Normalize the objectives**
3. **Construct reference lines** (from the origin to reference points)



NSGA-III - Selection of next population

1. **Ranks** $\rightarrow$ select the Pareto fronts as long as they completely fit in the new population
2. **Normalize the objectives**
3. **Construct reference lines** (from the origin to reference points)
4. **Associate each member of S_t to their nearest reference line** (perpendicular distance) and store the distance.



NSGA-III - Selection of next population

1. **Ranks** $\rightarrow$ select the Pareto fronts as long as they completely fit in the new population
2. **Normalize the objectives**
3. **Construct reference lines** (from the origin to reference points)
4. **Associate each member of S_t to their nearest reference line** (perpendicular distance) and store the distance.
5. **Compute the niche count of each reference point** on every solution that is already included in the next generation $\rho_j, j \in Z^r$ (how many solutions are associated with the j-th reference point)

NSGA-III - Selection of next population

1. **Ranks** → select the Pareto fronts as long as they completely fit in the new population
2. **Normalize the objectives**.
3. **Construct reference lines** (from the origin to reference points)
4. **Associate each member of S_t to their nearest reference line** (perpendicular distance) and store the distance.
5. **Compute the niche count of each reference point** on every solution that is already included in the next generation $\rho_j, j \in Z^r$ (how many solutions are associated with the j-th reference point)
6. **Fill the population using the last rank (which does not fit entirely into the new population):**
   A. Identify the reference point with the smallest niche count
   B. find the solutions in the exceeding PF that associate to it
      - If they exist, select one (best/random) solution to add to the final population and increment the niche count
      - If they do not exist, the reference point is removed from this analysis
   C. Repeat until the new generation is filled