Nonlinear Multiobjective Optimization

Prof. Kaisa Miettinen

kaisa.miettinen@jyu.fi

Faculty of Information Technology
University of Jyväskylä, Finland

http://www.mit.jyu.fi/miettine

Industrial Optimization Group
http://www.mit.jyu.fi/optgroup

Contents

- Some concepts
- Four classes of methods
- Some methods in each class
- Computationally expensive problems
- Hybrid methods
- Visualization
- Some applications
- Conclusions

Most figures from Miettinen (1999)

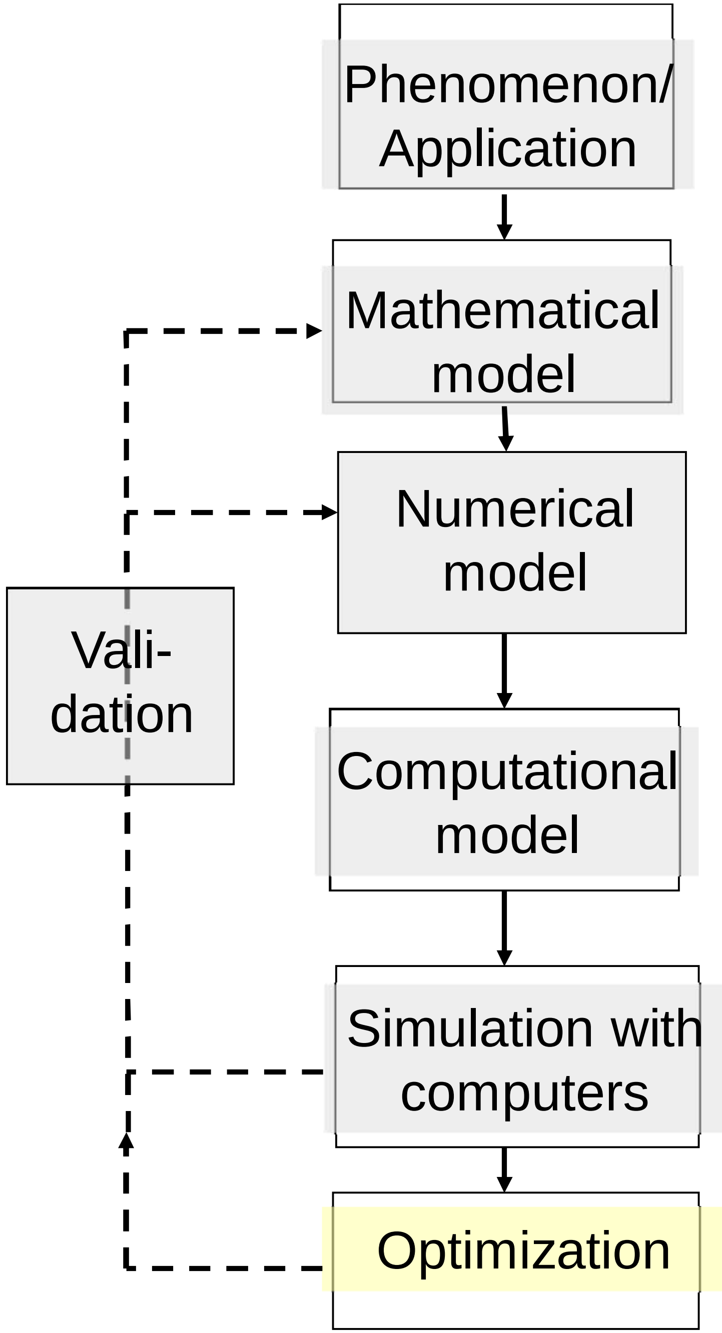
Problems with Multiple Criteria

- **Different features**
- **One decision maker (DM) – several DMs**
- **Deterministic – stochastic**
- **Continuous – discrete**
- **Nonlinear – linear**

→ **Nonlinear multiobjective optimization**

Modelling

- Modelling + simulation not enough alone!
- Reliable models required for optimization
- Optimization enables taking full advantage of high-quality models
- Challenging to combine different models



Nonlinear Multiobjective Optimization

- Most real-life problems have **several conflicting objectives** to be considered simultaneously and they may be nonlinear depending on variables
- Multiobjective optimization
  - Formulating each relevant aspect as an objective function
  - Typically easier than to try to form a single objective and measure all relevant points of view e.g. in money
  - Reveals *true nature* of problem without simplifications and *real interrelationships* between the objective functions
  - Can make the problem computationally easier to solve
  - The feasible region may turn out to be empty -> minimize constraint violations

Problem

We consider multiobjective optimization problems

$$\text{minimize} \quad \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_k(\mathbf{x}) \end{bmatrix}$$
$$\text{subject to} \quad \mathbf{x} \in S,$$

in other words

$$\text{minimize} \quad \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\}$$
$$\text{subject to} \quad \mathbf{x} \in S,$$

where

f_i: S→R = *objective function*

k (≥ 2) = number of (conflicting) objective functions

x = *decision vector* (of n *decision variables* x_i)

S ⊂ R^n = *feasible region* formed by *constraint functions* and

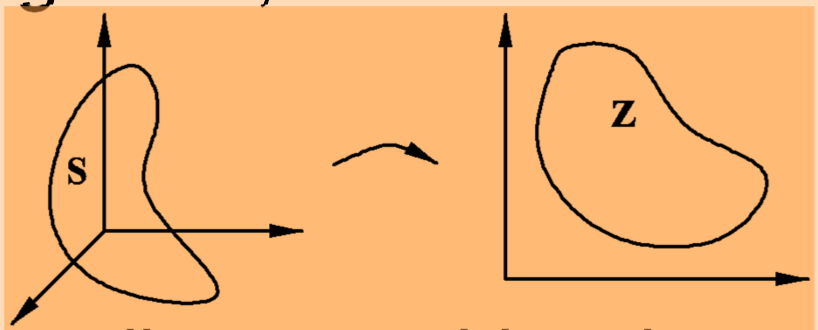
``minimize´´ = minimize the objective functions simultaneously

some constraint/objective functions is nonlinear

Concepts

- S consists of linear, nonlinear and/or box constraints for the variables
- We denote *objective function values* by $z_i = f_i(x)$
- $z = (z_1, \ldots, z_k)$ is an *objective vector*
- $Z \subset R^k$ denotes the image of S; feasible objective region

  Thus $z \in Z$



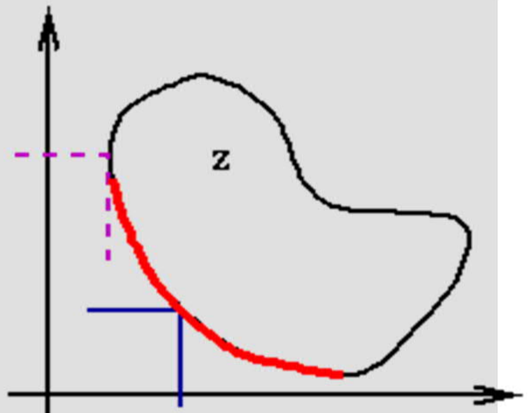
Definition: If all functions are linear, problem is *linear* (*MOLP*). If some functions are nonlinear, we have a *nonlinear multiobjective optimization problem*. Problem is *nondifferentiable* if some functions are nondifferentiable and *convex* if all objectives and S are convex

Optimality

- Contradiction and possible incommensurability $\Rightarrow$
- $x^* \in S$ is Pareto optimal (PO) if there does not exist another $x \in S$ such that $f_i(x) \leq f_i(x^*)$ for all $i=1,\ldots,k$ and $f_j(x) < f_j(x^*)$ for at least one j. Objective vector $z^* = f(x^*) \in Z$ is Pareto optimal if x^* is

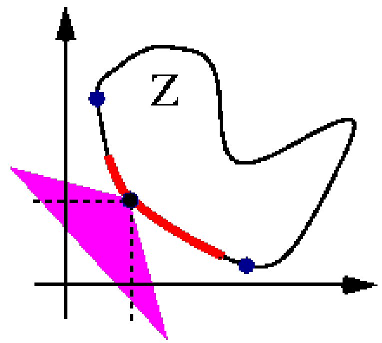
  i.e. $(z^* - R^k_+ \setminus \{0\}) \cap Z = \emptyset$,

  that is, $(z^* - R^k_+) \cap Z = z^*$.



- PO solutions form a (possibly nonconvex and disconnected) PO set
- $x^* \in S$ is weakly PO if there does not exist another $x \in S$ such that $f_i(x) < f_i(x^*)$ for all $i=1,\ldots,k$

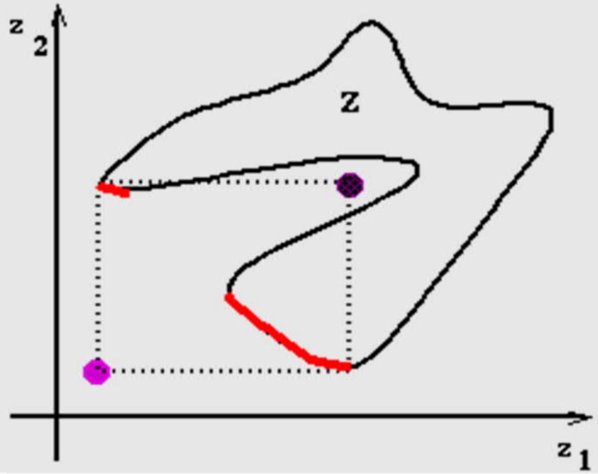
  i.e. $(z^* - \text{int}\, R^k_+) \cap Z = \emptyset$.



- Properly PO: unbounded trade-offs are not allowed. Weak PO $\supset$ PO $\supset$ proper PO

More

- Ranges of objective function values in PO set
  - Ideal objective vector **z*** of individual optima
  - Utopian objective vector **z**** (strictly better)
  - Nadir objective vector $\mathbf{z}^{nad}$ (estimated)
- Decision maker (DM) can express preferences, is responsible for final solution
- *Analyst* is responsible for mathematical side
- Help DM in finding most preferred (PO) solution
- Solution = best possible compromise
- We need preference information from DM
- Objective vector z **dominates** objective vector y if $z_i \le y_i$ for all i =1,…,k and $z_j < y_j$ for at least one j
- Thus, Pareto optimal solutions are not dominated by any other feasible solution



Local and Global Optimality

- Paying attention to the Pareto optimal set and forgetting other solutions is acceptable only if we know that no unexpressed or approximated objective functions are involved!
- Assuming DM is rational and problem correctly specified, final solution is always PO
- A point $x^* \in S$ is *locally Pareto optimal* if it is Pareto optimal in some environment of x^*
- Global Pareto optimality $\Rightarrow$ local Pareto optimality
- Local PO $\Rightarrow$ global PO, if S convex, f_i:s quasiconvex with at least one strictly quasiconvex f_i

More Concepts

- Value function $U:R^k \rightarrow R$ may represent preferences
- If $U(z^1) > U(z^2)$ then the DM prefers z^1 to z^2. If $U(z^1) = U(z^2)$ then z^1 and z^2 are equally good (indifferent)
- U is assumed to be strongly decreasing = *less is preferred to more*. Implicit U is often assumed

- Decision making can be thought of being based on either value maximization or *satisficing*
- An objective vector containing the *aspiration levels* $\check{z}_i$ of the DM is called a *reference point* $\check{z} \in R^k$

Results

- Sawaragi, Nakayama, Tanino: Pareto optimal solution(s) exist if
  - the objective functions are lower semicontinuous and
  - the feasible region is nonempty and compact
- Karush-Kuhn-Tucker optimality conditions can be formed as a natural extension to single objective optimization for both differentiable and nondifferentiable problems

Trading off

- Moving from one PO solution to another = trading off
- Definition: Given x^1 and $x^2 \in S$, the ratio of change between f_i and f_j is

$$\Lambda_{ij} = \Lambda_{ij}(\mathbf{x}^1, \mathbf{x}^2) = \frac{f_i(\mathbf{x}^1) - f_i(\mathbf{x}^2)}{f_j(\mathbf{x}^1) - f_j(\mathbf{x}^2)}.$$

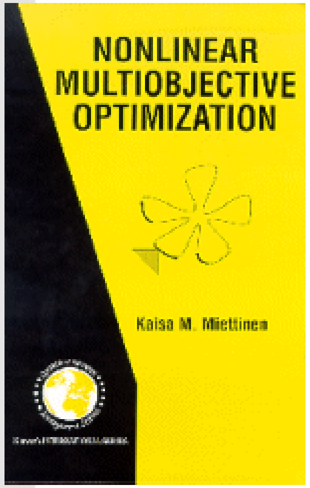
- Λ_{ij} is a *partial trade-off* if $f_l(x^1) = f_l(x^2)$ for all $l=1,\ldots,k$, $l \neq i,j$. If $f_l(x^1) \neq f_l(x^2)$ for at least one l and $l \neq i,j$, then Λ_{ij} is a *total trade-off*
- Let d^* be a feasible direction from $x^* \in S$. The *total trade-off rate* along the direction d^* is

$$\lambda_{ij} = \lambda_{ij}(\mathbf{x}^*, \mathbf{d}^*) = \lim_{\alpha \to 0} \Lambda_{ij}(\mathbf{x}^* + \alpha \mathbf{d}^*, \mathbf{x}^*).$$

- If $f_l(x^*+\alpha d^*) = f_l(x^*)$ $\forall$ $l \neq i,j$ and for all $0 \leq \alpha \leq \alpha^*$, then λ_{ij} is a *partial trade-off rate*

Methods for Multiple Objectives

- Finding a Pareto optimal set or a representation of it = *vector optimization*
- Typically methods use *scalarization* for converting the problem into a single objective one
  - **Scalarization contains preference information & original objective functions**
  - **After scalarization, single objective optimizers are used**
- Methods differ on what information is exchanged between method ↔ DM as well as how problem is scalarized
- *Classification* according to the role of the DM
  - Not present, before, after or during solution process
- Based on the existence of a value function:
  - ad hoc: U would not help
  - non ad hoc: U helps
- Kaisa Miettinen: Nonlinear Multiobjective Optimization, Kluwer (Springer), Boston, 1999



Scalarizing Functions

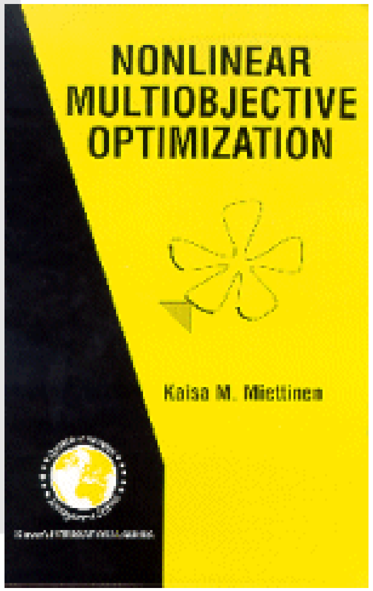
- Scalarization = combine preferences and original problem $\Rightarrow$ scalarized single objective subproblem
- Resulting subproblem is solved with an appropriate single objective optimization method
- Objective function is called scalarizing (or scalarization) function
- Desirable properties
  - Optimal solution is PO
  - Any PO solution can be found

Criteria for Good Decision Support System

- Recognizes and generates PO solutions
- Helps DM feel convinced that final solution is the most preferred one or at least close enough to that
- Helps DM to get a “holistic” view over PO set
- Does not require too much time from DM to find final solution
- Communication between DM and system not too complicated
- Provides reliable information about alternatives available

Four Classes of Methods

- How to support DM?
- Four types of methods (Hwang and Masud, 1979)
- ***No decision maker*** – some neutral compromise solution
- ***A priori methods****:* DM sets hopes and closest solution is found
  - Expectations may be too optimistic or pessimistic
  - Hard to express preferences without knowing the problem well
- ***A posteriori methods***: generate representation of PO set
  - \+ Gives information about variety of PO solutions
  - Expensive, computationally demanding
  - Difficult to represent the PO set if k > 2
  - o Example: evolutionary multiobjective optimization methods
- ***Interactive methods***: iterative search process
  - \+ Avoid difficulties above
  - \+ Solution pattern is formed and repeated iteratively
  - \+ Move around Pareto optimal set
  - \+ What can we expect DMs to be able to say?
  - \+ Goal: easiness of use
  - \+ Cognitively valid approaches: classification and reference point consisting of aspiration levels
- Further information: Kaisa Miettinen: **Nonlinear Multiobjective Optimization**, Kluwer (Springer), 1999

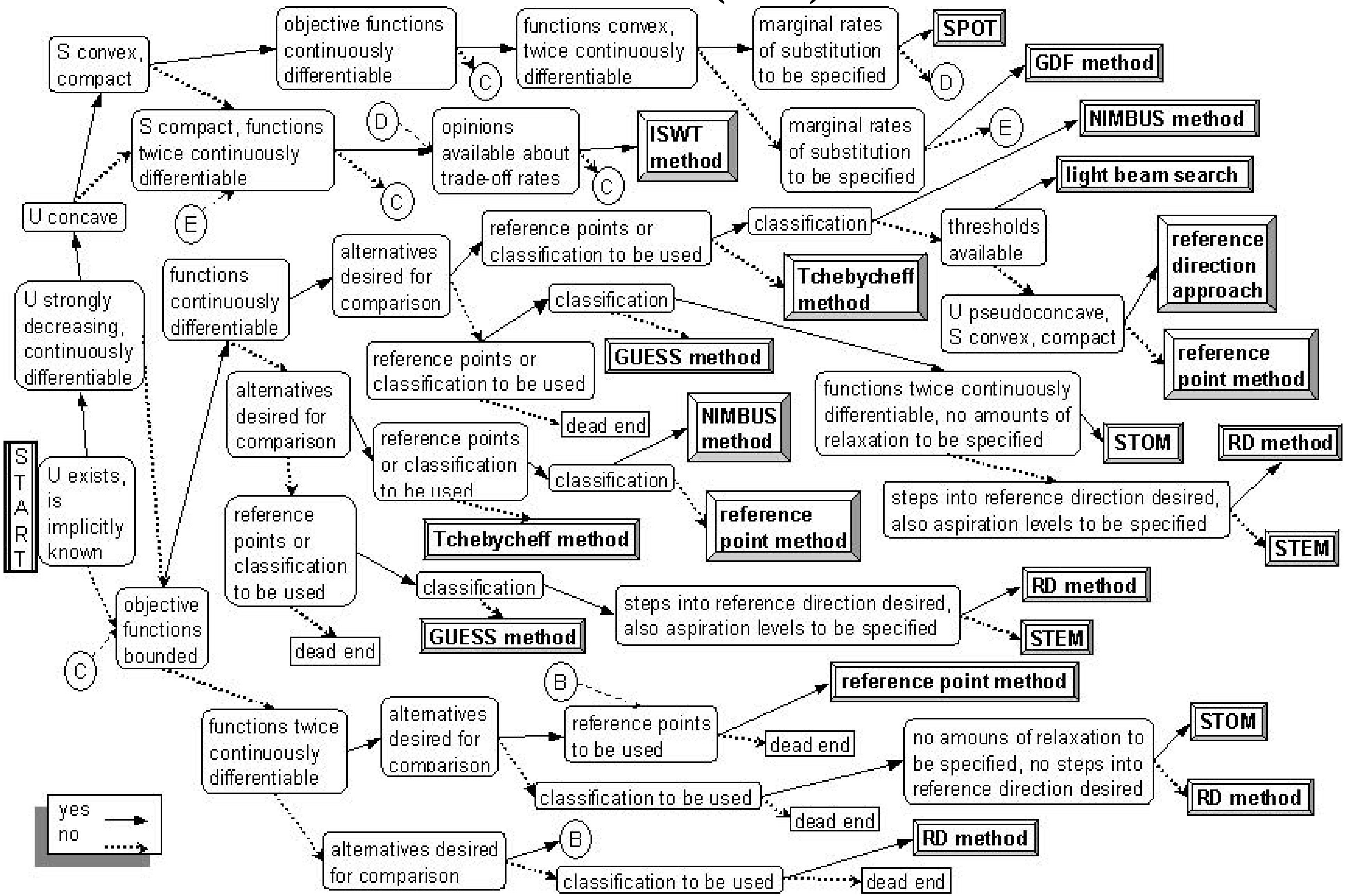


Methods cont.

- **No-preference methods**
  - Meth. of Global Criterion
- **A posteriori methods**
  - Weighting Method
  - ε-Constraint Method
  - Hybrid Method
  - Method of Weig. Metrics
  - Achievement Scalarizing Function Approach
- **A priori methods**
  - Value Function Method
  - Lexicographic Ordering
  - Goal Programming
- **Interactive methods**
  - Interactive Surrogate Worth Trade-Off Method
  - GDF Method
  - Tchebycheff Method
  - Reference Point Method
  - GUESS Method
  - Reference Direction Approach
  - Satisficing Trade-Off Method
  - Light Beam Search
  - NIMBUS Method

Tree Diagram of Methods

Miettinen (1999)



No-Preference Methods: Method of Global Criterion (Yu, Zeleny)

- Distance between z⋆ and Z is minimized by L_p-metric: if global ideal objective vector is known

$$\begin{array}{ll} \text{minimize} & \left(\sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^\star)^p \right)^{1/p} \\ \text{subject to} & \mathbf{x} \in S \end{array}$$

- or by L_∞-metric:

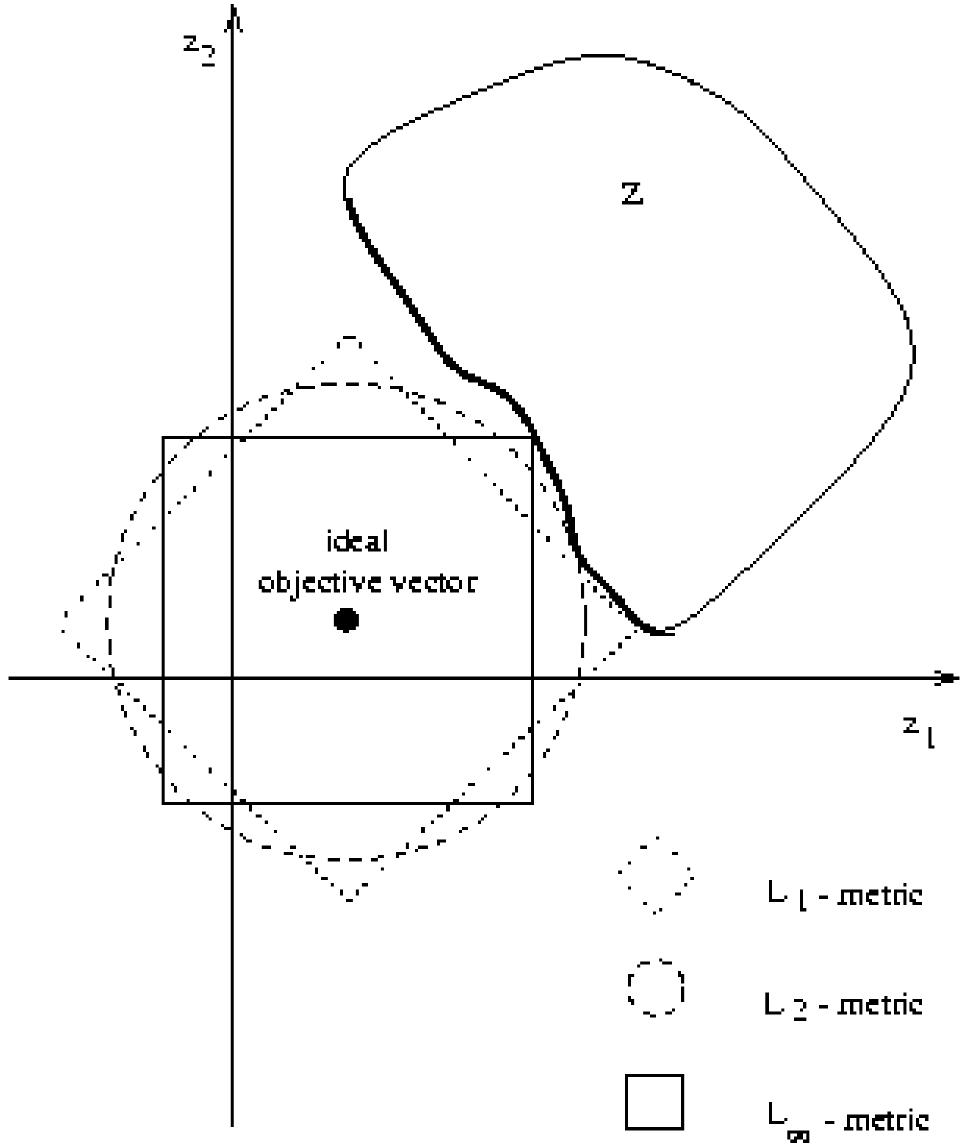
$$\begin{array}{ll} \text{minimize} & \max_{1 \le i \le k} [f_i(\mathbf{x}) - z_i^\star] \\ \text{subject to} & \mathbf{x} \in S. \end{array}$$

- Differentiable form of the latter:

$$\begin{array}{lll} \text{minimize} & \alpha & \\ \text{subject to} & \alpha \ge (f_i(\mathbf{x}) - z_i^\star), & \text{for all } i = 1, \ldots, k, \\ & \mathbf{x} \in S, & \end{array}$$

Method of Global Criterion cont.

- ? The choice of p affects greatly the solution
- + Solution of the L_p-metric ($p < \infty$) is PO
- ≈ Solution of the L_∞-metric is weakly PO and the problem has at least one PO solution
- + Simple method (no special hopes are set)



A Posteriori Methods

- Generate the PO set, actually a representation of it
- Present it to the DM
- Let the DM select one
  - Computationally expensive/difficult
  - Hard to select from a set
  - How to display the alternatives (if $k > 2$)?

Weighting Method (Gass, Saaty)

➜ Problem

$$
\begin{array}{ll}
\text{minimize} & \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \\
\text{subject to} & \mathbf{x} \in S, \\
\text{where} & \sum_{i=1}^{k} w_i = 1 \\
 & w_i \geq 0 \ \forall \ i = 1, \ldots, k.
\end{array}
$$

≈ Solution is weakly PO

\+ Solution is PO if it is unique or $w_i > 0$ for all i

\+ Convex problems: any PO solution can be found

– Nonconvex problems: some of the PO solutions may fail to be found

Weighting Method cont.

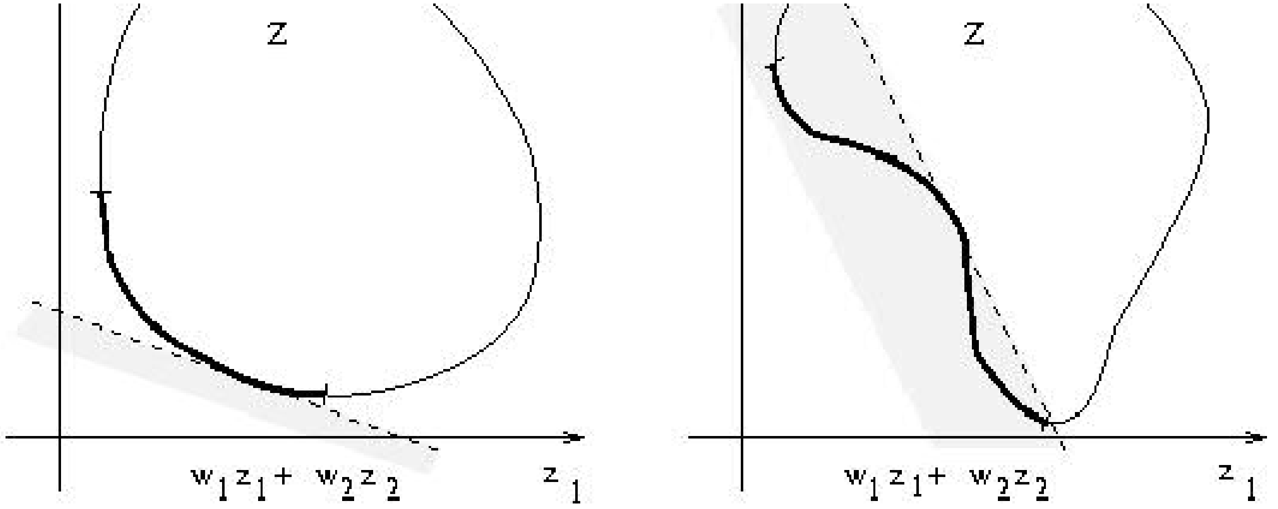


Figure 3. Convex and nonconvex problems.

– Weights are not easy to be understood (correlation, nonlinear affects). Small change in weights may change the solution dramatically
– Evenly distributed weights do not produce an evenly distributed representation of the PO set

ε-Constraint Method (Haimes et al)

- Problem

$$\begin{array}{ll} \text{minimize} & f_\ell(\mathbf{x}) \\ \text{subject to} & f_j(\mathbf{x}) \le \varepsilon_j, \ \text{for all } j = 1, \ldots, k, j \ne \ell \\ & \mathbf{x} \in S. \end{array}$$

≈ The solution is weakly Pareto optimal

\+ x* is PO iff it is a solution when $\varepsilon_j = f_j(x^*)$ (i=1,…,k, j≠l) for all objectives to be minimized

\+ A unique solution is PO

\+ Any PO solution can be found with some effort

– There may be difficulties in specifying upper bounds

Trade-Off Information

- Let the feasible region be of the form $S = \{x \in R^n \mid g(x) = (g_1(x),\ldots, g_m(x))^T \leq 0\}$
- *Lagrange function* of the ε-constraint problem is

$$f_\ell(\mathbf{x}) + \sum_{j \neq \ell} \lambda_j (f_j(\mathbf{x}) - \varepsilon_j) + \sum_{i=1}^{m} \mu_i g_i(\mathbf{x}).$$

- Under certain assumptions the coefficients $\lambda_j = \lambda_{lj}$ are (partial or total) trade-off rates

Method of Weighted Metrics (Zeleny)

- Weighted metric formulations are

$$
\begin{aligned}
&\text{minimize} \quad \left(\sum_{i=1}^{k} w_i \big(f_i(\mathbf{x}) - z_i^\star\big)^p \right)^{1/p} \\
&\text{subject to} \quad \mathbf{x} \in S
\end{aligned}
$$

and

$$
\begin{aligned}
&\text{minimize} \quad \max_{1 \le i \le k} \left[w_i \big(f_i(\mathbf{x}) - z_i^\star\big) \right] \\
&\text{subject to} \quad \mathbf{x} \in S,
\end{aligned}
$$

where $w_i \geq 0$ for all i and $\sum_{i=1}^{k} w_i = 1$.

Method of Weighted Metrics cont.

+ If the solution is unique or the weights are positive, the solution of L_p-metric ($p<\infty$) is PO
+ For positive weights, the solution of L_∞-metric is weakly PO and there exists at least one PO solution
+ Any PO solution can be found with the L_∞-metric with positive weights if the reference point is utopian but some of the solutions may be weakly PO
– All the PO solutions may not be found with $p<\infty$

✿

$$\begin{aligned} &\min \quad \max_{i=1,\ldots,k} \left[w_i (f_i(\mathbf{x}) - z_i^{\star\star}) \right] + \rho \sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^{\star\star}) \\ &\text{s.t.} \quad \mathbf{x} \in S, \end{aligned}$$

where $\rho>0$. This generates properly PO solutions and any properly PO solution can be found

Achievement Functions cont. (Wierzbicki)

- Example of order-representing functions:

$$s_{\bar{\mathbf{z}}}(\mathbf{z}) = \max_{1 \le i \le k} [w_i(z_i - \bar{z}_i)],$$

where w is some fixed positive weighting vector

- Example of order-approximating functions:

$$s_{\bar{\mathbf{z}}}(\mathbf{z}) = \max_{1 \le i \le k} [w_i(z_i - \bar{z}_i)] + \rho \sum_{i=1}^{k} w_i(z_i - \bar{z}_i),$$

where w is as above and $\rho > 0$ sufficiently small.

\+ The DM can obtain any arbitrary (weakly) PO solution by moving the reference point only

Achievement Scalar. Fun. cont.

$$s(\mathbf{f}(\mathbf{x})) = \max_{i=1,\ldots,k} [\, w_i(f_i(\mathbf{x}) - \bar{z}_i) \,] + \rho \sum_{i=1}^{k} w_i(f_i(\mathbf{x}) - \bar{z}_i)$$

- Solution is Pareto optimal
- Any properly Pareto optimal solution can be found

Two Worlds: MCDM and EMO

Multiple criteria decision making

- Role of DM and decision support emphasized
- Role of preference information important
- Different types of methods - interactive ones widely developed
- Solid theoretical background (we can prove Pareto optimality etc.)
- Scalarization combining objective and preferences into real-valued functions

Evolutionary multiobjective optimization (EMO)

- Idea to approximate the set of Pareto optimal solutions
- Criteria: minimize distance to real PO set and maximize diversity of approximation
- Not too much emphasis on DM's preferences until recently
- Cannot guarantee actual optimality
- E.g. nonconvexity and discontinuity cause no difficulties
- Background in applications
- Many benchmark problems for testing goodness of methods (to measure quality of approximation generated) + performance criteria
- Terminology: bi-multi-many
- Nondominated = PO in a subset

EMO

- Evolutionary algorithms: common metaheuristics
- Work well for mathematically difficult problems (no assumptions)
- Population-based approaches
- Population of solutions is manipulated with operations (selection, crossover, mutation) and the population approximates the PO set
- Many different EMO methods exist
- Problems
  - Diversity preserving mechanisms
  - Getting close to really PO solutions
- On the other hand
  - Computational effort is wasted in finding undesired solutions
  - Many solutions are presented to DM who can be unable to compare and find most preferred among them when $k > 2$
- Many EMO methods do not work well when $k>2$ or 3
- Combine ideas of MCDM and EMO methods

EMO cont.

- Population-based methods
  - Variables can be coded indifferent ways
  - Repeated for generations
  - At every generation, generates a set of solutions
- VEGA, RWGA, MOGA, NSGA, NSGA-II, DPGA, SPEA-2 etc.
  - Work best when k=2
- Goals: maintaining diversity and guaranteeing Pareto optimality – how to measure?
- Special operators have been introduced
- Typically tested with benchmark problems with known PO sets
- For k>3: MOEA/D, NSGA-III, RVEA etc.

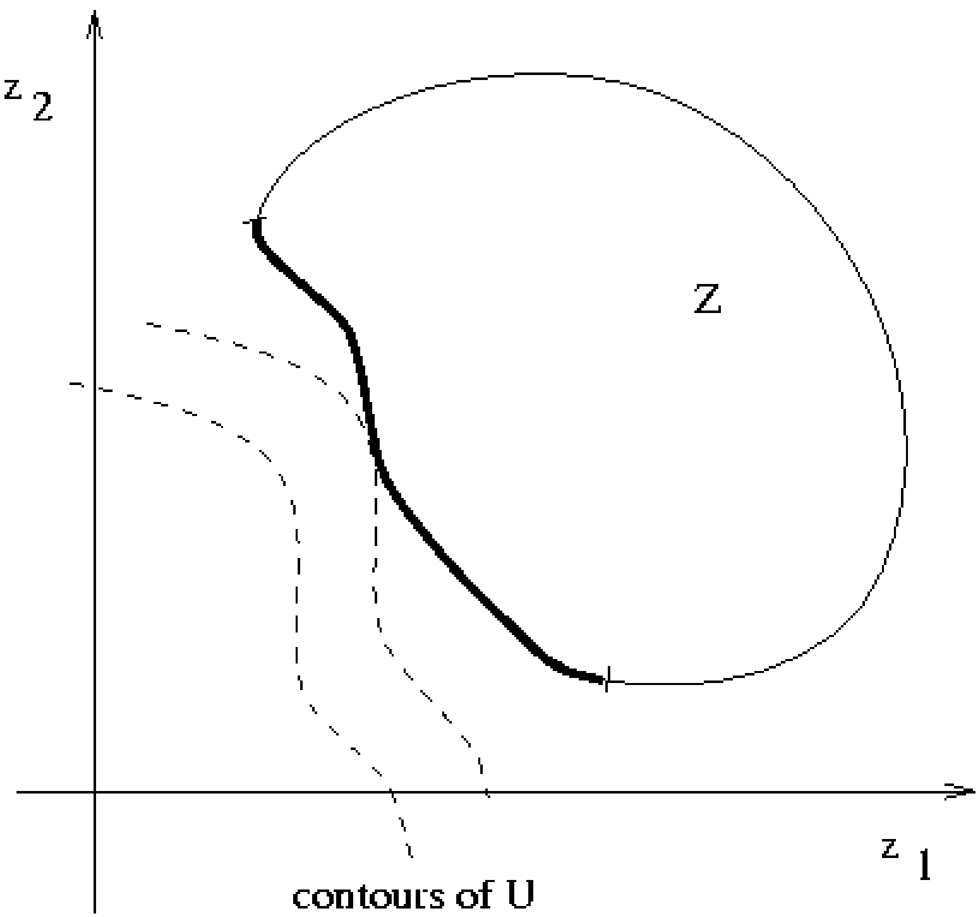
A Priori Methods

- DM specifies hopes, preferences, opinions
- – DM does not necessarily know how realistic hopes are (expectations may be too high)

Value Function Method (Keeney, Raiffa)

↗ Problem

$$\begin{array}{ll} \text{maximize} & U(f_1(\mathbf{x}), \ldots, f_k(\mathbf{x})) \\ \text{subject to} & \mathbf{x} \in S \end{array}$$



Lexicographic Ordering

- The DM must specify an absolute order of importance for objectives, i.e., $f_i >>> f_{i+1} >>> \ldots.$
- If the most important objective has a unique solution, stop. Otherwise, optimize the second most important objective such that the most important objective maintains its optimal value etc.

+ The solution is Pareto optimal.
+ Some people make decisions successively.

– Difficulty: specify the absolute order of importance.
– The method is robust. The less important objectives have very little chances to affect the final solution
– Trading off is impossible

Interactive Methods

- Most developed class of methods
- A solution pattern is formed and repeated iteratively
- DM directs the solution process, i.e. movement around PO set
- DM needs time and interest for co-operation
- Only some PO points (those that are interesting to the DM) are generated
- DM is not overloaded with information
- DM can *learn*: specify and correct preferences and selections as the solution process continues
- DM has more confidence in the final solution
- Important aspects
  - what is asked – what can we expect DMs to be able to say?
  - what is told – goal: easiness of use
  - how the problem is scalarized
- Psychological convergence!

Interactive Methods, cont.

- *DM is assumed to have knowledge about the problem in question, no deep understanding of optimization or its theory*
- Solution process is iterative
- Role of DM important
  - Final solution = best possible, i.e., most preferred PO solution
  - DM is responsible for the final solution
- **DM should understand how to use method**
  - Information asked and given must be understandable
- Goal: easiness of use
  - no difficult questions (like cognitive mapping)
  - possibility to change one’s mind, i.e. enable learning

Interactive Methods, cont.

- In each iteration, the DM is shown Pareto optimal solutions and asked to specify new preference information to generate more satisfactory new Pareto optimal solution(s)
- Thus, DM influences from which part of the Pareto optimal set solutions are considered
- DM obtains
  - new information and insight about the interdependencies among objective functions
  - understanding of the feasibility of preferences
- New knowledge obtained may affect preferences, leading to solutions which were not previously considered
- User interface plays an important role

Core Structure **Ojalehto et al, COAP (2014)**

1. Initialize solution process, e.g., calculate ideal and nadir objective vectors
2. Solve a method-specific subproblem to generate an initial Pareto optimal solution as a current solution
3. Ask the DM to provide preference information related to the current solution
4. Generate new solution(s) based on the preference information by solving appropriate subproblem(s)
5. Ask the DM to select the best solution of the previously generated solutions and denote it as the current solution
6. If current solution is satisfactory, stop. Otherwise continue from step 3

Examples of Preference Information

- ❖ Selecting desired or undesired from a sample of PO solutions
- ❖ Pairwise comparison
- ❖ Desirable values (->reference point) or ranges for objective functions
- ❖ Classify objectives (improvement possible by allowing impairment)
- ❖ Opinion of marginal rates of substitution
- ❖ Desirability of trade-offs

Different DMs prefer different formats or want to change the format – need different methods

- Luque et al., OR Spectrum (2011), Ruiz et al, Annals of OR (2012)

Tchebycheff Method (Steuer)

- Idea: Interactive weighting space reduction method. Different solutions are generated with well dispersed weights. The weight space is reduced in the neighbourhood of the best solution
- Assumptions: Utopian objective vector is available
- Weighted distance (Tchebycheff metric) between the utopian objective vector and Z is minimized:

$$\begin{array}{ll} \text{lex minimize} & \max_{i=1,\dots,k} \left[w_i(f_i(\mathbf{x}) - z_i^{\star\star}) \right], \sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^{\star\star}) \\ \text{subject to} & \mathbf{x} \in S. \end{array}$$

- It guarantees Pareto optimality and any Pareto optimal solution can be found

Tchebycheff Method cont.

- At first, weights between [0,1] are generated.
- Iteratively, the upper and lower bounds of the weighting space are tightened.
- <u>Algorithm</u>

1) Specify number of alternatives P and number of iterations H. Construct z◆◆. Set h=1.
2) Form the current weighting vector space and generate 2P dispersed weighting vectors.
3) Solve the problem for each of the 2P weights.
4) Present the P most different of the objective vectors and let the DM choose the most preferred.
5) If h=H, stop. Otherwise, gather information for reducing the weight space, set h=h+1 and go to 2).

Tchebycheff Method cont.

- Non ad hoc method

\+ All the DM has to do is to compare several Pareto optimal objective vectors and select the most preferred one.

! The ease of the comparison depends on P and k.

– The discarded parts of the weighting vector space cannot be restored if the DM changes her/his mind.

– A great deal of calculation is needed at each iteration and many of the results are discarded.

\+ Parallel computing can be utilized.

Reference Point Method (Wierzbicki)

- Idea: Direct the search by reference points representing desirable values for the objectives and generate new alternatives by shifting the reference point
- Reference point is projected onto PO set with achievement scalarizing function
- Solution is properly PO

$$\text{minimize} \quad \max_{i=1,\ldots,k} \left[\frac{f_i(\mathbf{x}) - \bar{z}_i}{z_i^{\mathsf{nad}} - z_i^{\star\star}} \right] + \rho \sum_{i=1}^{k} \frac{f_i(\mathbf{x})}{z_i^{\mathsf{nad}} - z_i^{\star\star}}$$
$$\text{subject to} \quad \mathbf{x} \in S.$$

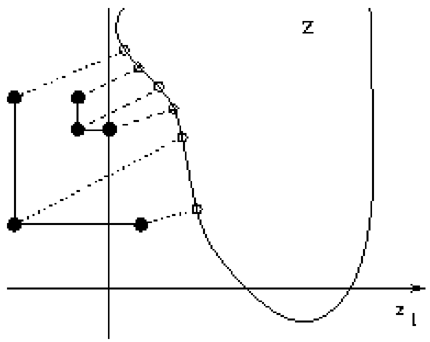


Figure 6. Altering the reference points.

Reference Point Method Algorithm

- No specific assumptions
- Algorithm:

1) Present information to the DM. Set h=1.
2) Ask the DM to specify a reference point $\check{z}^h$.
3) Minimize ach. function. Present z^h to the DM.
4) Calculate k other solutions with reference points

$$\bar{\mathbf{z}}(i) = \bar{\mathbf{z}}^h + d^h \mathbf{e}^i,$$

   where $d^h = \|\check{z}^h - z^h\|$ and e^i is the *i*th unit vector.
5) If the DM can select the final solution, stop. Otherwise, ask the DM to specify $\check{z}^{h+1}$. Set h=h+1 and go to 3).

Reference Point Method cont.

- ❁ Ad hoc method (or both)
- \+ Easy for the DM to understand: (s)he has to specify aspiration levels and compare objective vectors.
- \+ For nondifferentiable problems, as well
- \+ No consistency required
- – Easiness of comparison depends on the problem
- – No clear strategy to produce the final solution

Satisficing Trade-Off Method (Nakayama et al)

- Idea: To classify the objective functions:
  - functions to be improved
  - acceptable functions
  - functions whose values can be relaxed
- Assumptions
  - functions are twice continuously differentiable
  - trade-off information is available in the KKT multipliers
- Aspiration levels from the DM, upper bounds from the KKT multipliers
- Satisficing decision making is emphasized

Satisficing Trade-Off Method cont.

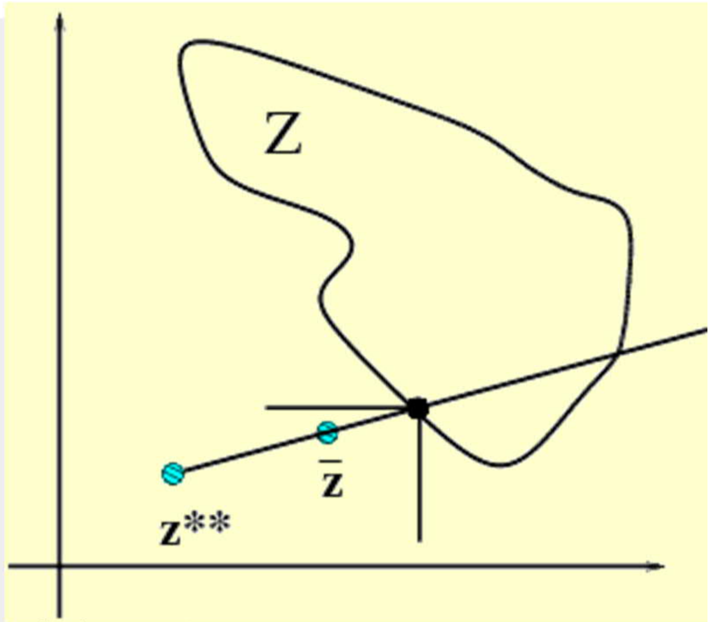
➔ <u>Problem</u>

$$\max_{1\le i\le k}\left[\frac{f_i(\mathbf{x})-z_i^{\star\star}}{\bar{z}_i-z_i^{\star\star}}\right]$$

or

$$\max_{1\le i\le k}\left[\frac{f_i(\mathbf{x})-z_i^{\star\star}}{\bar{z}_i-z_i^{\star\star}}\right]+\rho\sum_{i=1}^{k}\frac{f_i(\mathbf{x})}{\bar{z}_i-z_i^{\star\star}},$$

where $\check{z}^h > z^{\bullet\bullet}$ and $\rho>0$. Solution weakly or properly PO, respectively



- Any (properly) PO solution can be found
- Partial trade-off rate information can be obtained from optimal KKT multipliers of the differentiable counterpart problem

Satisficing Trade-Off Algorithm

1) Calculate $z^{\blacklozenge\blacklozenge}$ and get a starting solution.
2) Ask the DM to classify the objective functions into the three classes. If no improvements are desired, stop.
3) If trade-off rates are not available, ask the DM to specify aspiration levels and upper bounds. Otherwise, ask the DM to specify aspiration levels. Utilize automatic trade-off in specifying the upper bounds for the functions to be relaxed. Let the DM modify the calculated levels, if necessary.
4) Solve the problem. Go to 2).

Background for NIMBUS®

- **DM should understand how to use method**
- Solution = best possible compromise
- DM is responsible for the final solution
- Difficult to present the Pareto optimal set, expectations may be too high
- Interactive approach avoids these difficulties
- Move around Pareto optimal set
- How can we support the learning process?
- DM should be able to direct the solution process
- Goal: easiness of use $\Rightarrow$ no difficult questions & possibility to change one's mind
- **Dealing with objective function values is understandable and straightforward**

Synchronous NIMBUS®

Miettinen, Mäkelä, EJOR (2006)

- Scalarization is important: contains preference information
- But scalarizations based on same input give different solutions (Miettinen, Mäkelä, OR Spec (2002))
- Which is the best? $\Rightarrow$ Synchronous NIMBUS®
  - 1-4 scalarized problem(s) formed to obtain different PO solutions
- Show them to the DM & let her/him choose the best
- DM can see how realistic hopes were and can adjust them
- Versatile possibilities to direct solution process
  - Besides classification, intermediate solutions between PO solutions can be generated
  - Classification and comparison of alternatives are used in the extent the DM desires
- DM can learn during the iterative solution process and only PO solutions that are interesting to her/him are generated

Classification in NIMBUS

- DM directs the search by classification**: Classification of objective functions into up to 5 classes**
- **Classification:** DM indicates **desirable changes in the current PO objective function values $f_i(x^h)$**
- **Classes: functions f_i whose values**
  - **should be decreased ($i \in I^<$)**
  - **should be decreased till some aspiration level $\check{z}_i^h < f_i(x^h)$ ($i \in I^\leq$)**
  - **are satisfactory at the moment ($i \in I^=$)**
  - **are allowed to increase up till some upper bound $\varepsilon_i^h > f_i(x^h)$ ($i \in I^>$)**
  - **are allowed to change freely ($i \in I^\Diamond$)**
- **DM must be willing to give up something**
- Miettinen, Mäkelä: Optim (1995), JORS (1999), Comp&OR (2000), EJOR (2006)

NIMBUS® Method cont.

- Solve subproblem

$$\begin{aligned}
\min \quad & \max_{\substack{i\in I^{<}\\ j\in I^{\leq}}} \left[\frac{f_i(\mathbf{x}) - z_i^{\star}}{z_i^{\mathsf{nad}} - z_i^{\star\star}}, \frac{f_j(\mathbf{x}) - \hat{z}_j}{z_j^{\mathsf{nad}} - z_j^{\star\star}} \right] + \rho \sum_{i=1}^{k} \frac{f_i(\mathbf{x})}{z_i^{\mathsf{nad}} - z_i^{\star\star}} \\
\text{s.t.} \quad & f_i(\mathbf{x}) \leq f_i(\mathbf{x}^c) \quad \text{for all} \quad i \in I^{<} \cup I^{\leq} \cup I^{=}, \\
& f_i(\mathbf{x}) \leq \varepsilon_i \quad \text{for all} \quad i \in I^{\geq}, \\
& \mathbf{x} \in S,
\end{aligned}$$

  where $\rho > 0$; appropriate single objective optimizer
- **Solution properly PO. Any PO solution can be found**
- Solution satisfies desires as well as possible – feedback of tradeoffs
- Possible to save interesting solutions and return to them later
- We have 3 more subproblems to get more solutions

Other Subproblems

- Classification implies reference point but not vice versa
- We use reference point based subproblems
- Components of reference point are obtained from classification information
  - $I^{<}$: corresponding component of ideal objective vector
  - $I^{\leq}$: aspiration level specified by the DM
  - $I^{=}$: current objective function valuer
  - $I^{\geq}$: upper bound specified by the DM
  - $I^{>}$: corresponding component of nadir objective vector

NIMBUS Method - Remarks

- Intermediate solutions between x^h and x'^h: $f(x^h+t_j d^h)$, where $d^h = x^{h'} - x^h$ and $t_j = j/(P+1)$
- Search iteratively around the PO set until DM does not want to improve or impair any objective
- Ad hoc method

\+ Versatile possibilities for the DM: classification, comparison, extracting undesirable solutions

\+ Does not depend entirely on how well the DM manages in classification. (S)he can e.g. specify loose upper bounds and get intermediate solutions

\+ Works for nondifferentiable/nonconvex problems

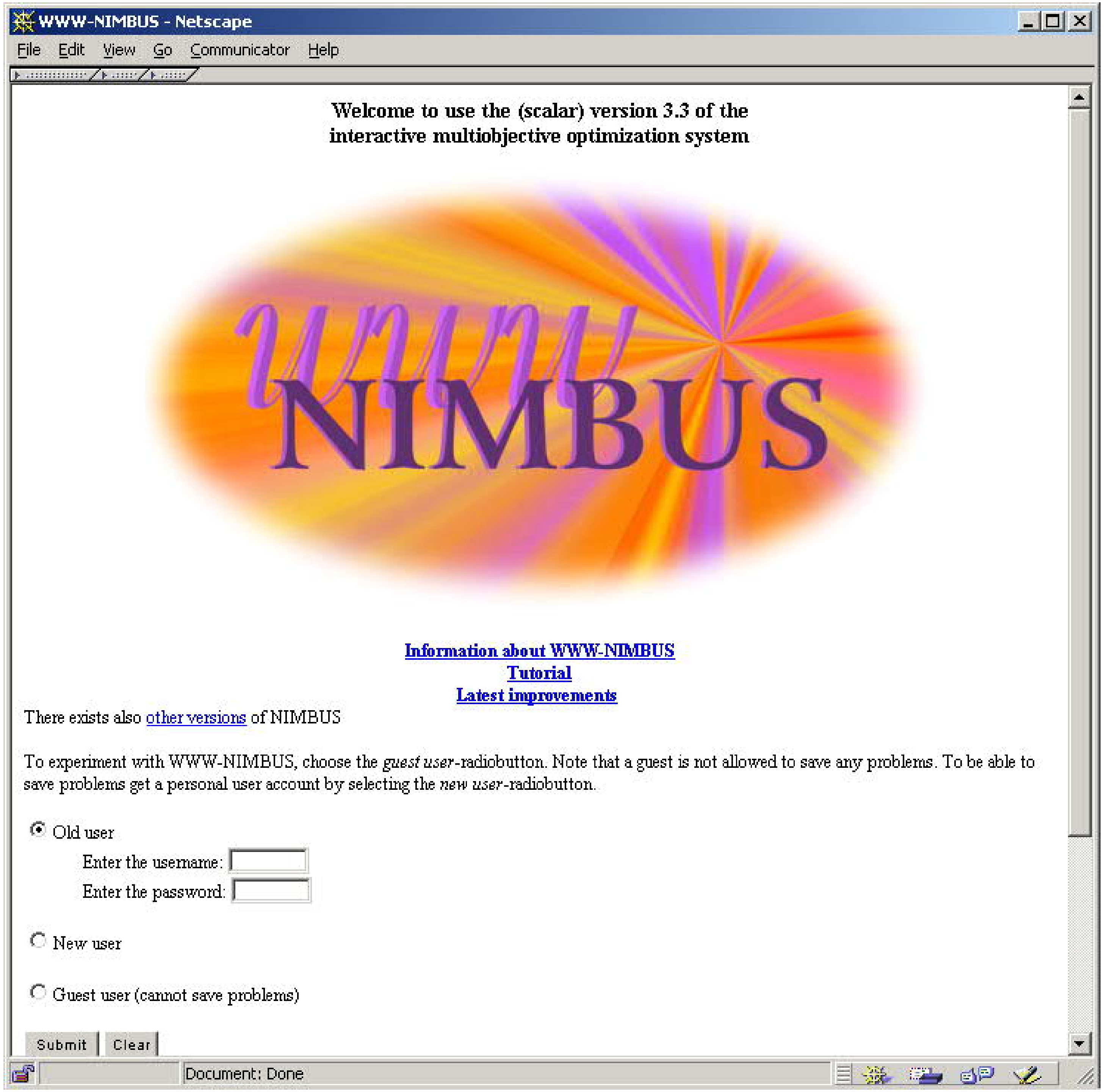
\+ No consistency is required – learning-oriented method

NIMBUS® Algorithm

1) Choose starting solution and project it to be PO.
2) Ask DM to classify the objectives and to specify related parameters. Solve 1-4 subproblems.
3) Present different solutions to DM.
4) If DM wants to save solutions, update database.
5) If DM does not want to see intermediate solutions, go to 7). Otherwise, ask DM to select the end points and the number of solutions.
6) Generate and project intermediate solutions. Go to 3).
7) Ask DM to choose the most preferred solution. If DM wants to continue, go to 2). Otherwise, stop.

WWW-NIMBUS® and IND-NIMBUS®

- **WWW-NIMBUS® http://nimbus.it.jyu.fi/**
  - The first, unique interactive optimization system on the Internet since 1995
  - Centralized computing & distributed interface
  - Latest version always available
  - Graphical user-interface via WWW
  - Available & **free** for any academic Internet user
  - Tutorial and online help
- **IND-NIMBUS® http://ind-nimbus.it.jyu.fi/**
  - For MS-Windows and Linux operating systems
  - Can be connected with different modelling and simulation tools like GAMS, Matlab, GPS-X, APROS
  - Different local and global solvers and their hybrids
  - User can change solver and its parameters
  - **E.g. wide applicability of single-objective evolutionary approaches available (**Miettinen, Materials & Manuf. Processes 2007)

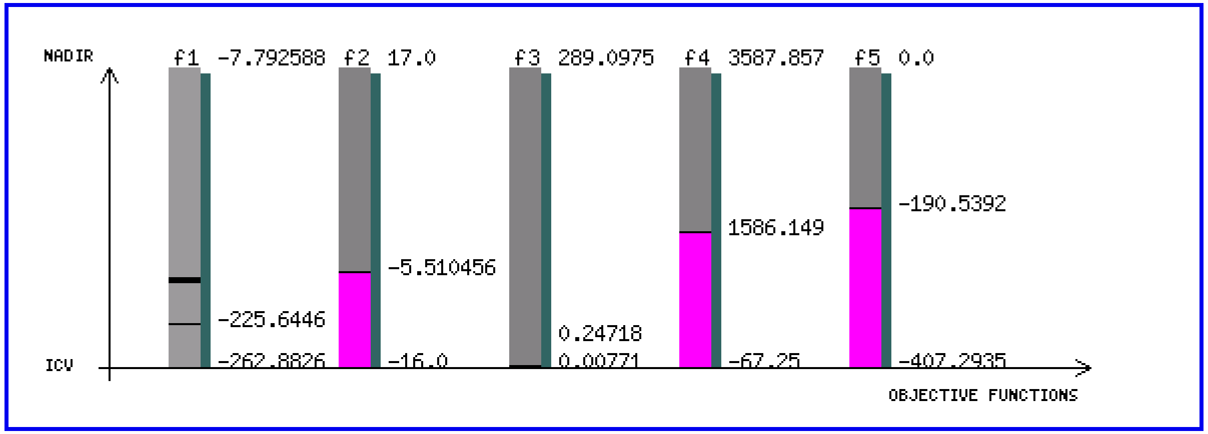

WWW-NIMBUS - Netscape
File Edit View Go Communicator Help
Welcome to use the (scalar) version 3.3 of the
interactive multiobjective optimization system
www
NIMBUS
Information about WWW-NIMBUS
Tutorial
Latest improvements
There exists also other versions of NIMBUS
To experiment with WWW-NIMBUS, choose the guest user-radiobutton. Note that a guest is not allowed to save any problems. To be able to save problems get a personal user account by selecting the new user-radiobutton.
Old user
Enter the username:
Enter the password:
New user
Guest user (cannot save problems)
Submit
Clear
Document: Done

WWW-NIMBUS ## Kalvokuva1 - Netscape

File Edit View Go Communicator Help

Classify Functions Graphically

Point out desired function values. Wait a moment after each click. **Do not use the back function of the browser on this page.**

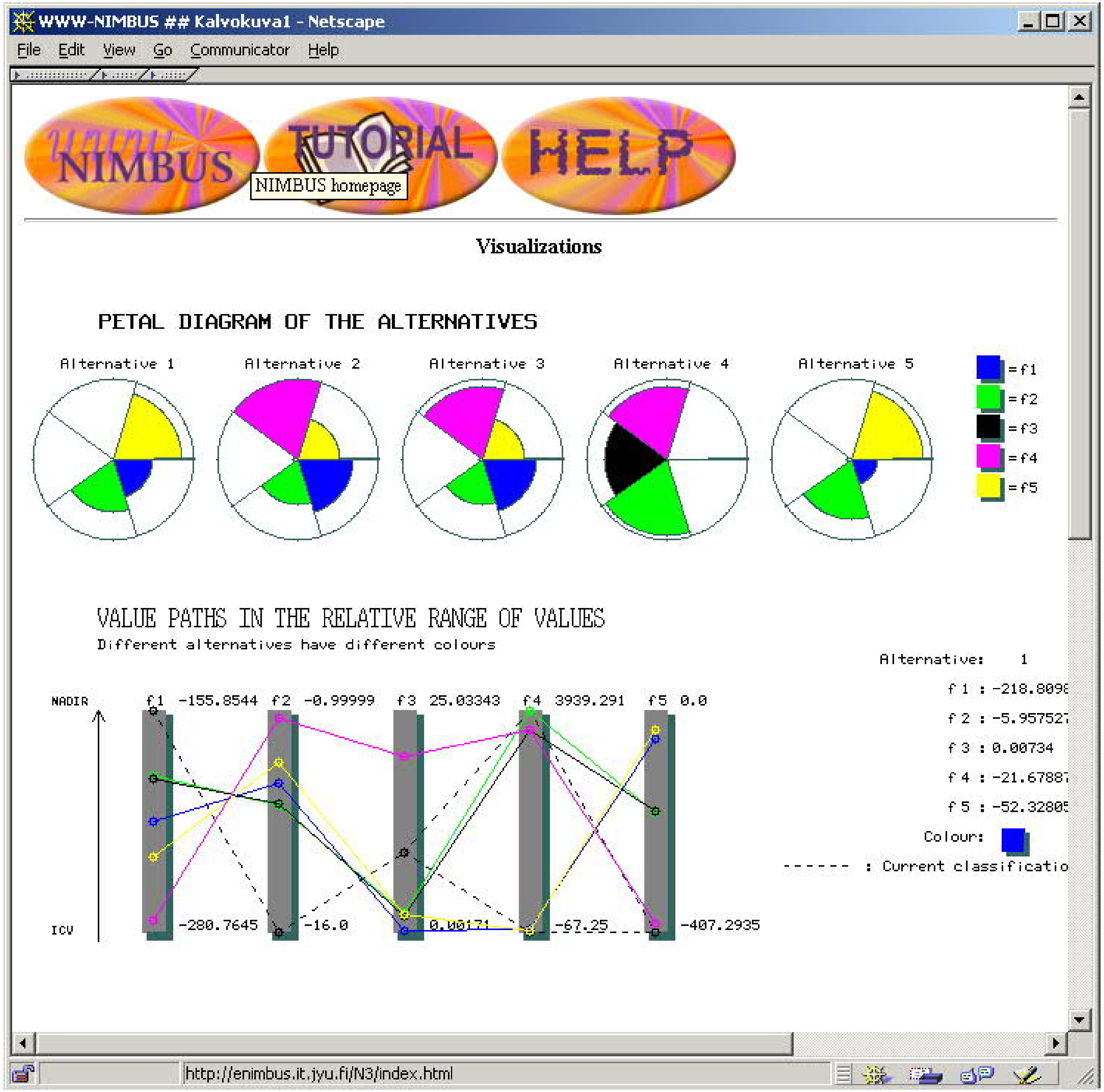


Function	ICV (estim.)	Current Solution	Nadir (estim.)
f1	-262.8827	-225.6446	-7.792588
f2	-16.0	-5.510456	17.0
f3	7.711097E-3	0.2471879	289.0975
f4	-67.25	1586.149	3587.858
f5	-407.2935	-190.5393	0.0

Next optimization: Global (GA + Deb)

Maximum number of new solutions to be generated: Four

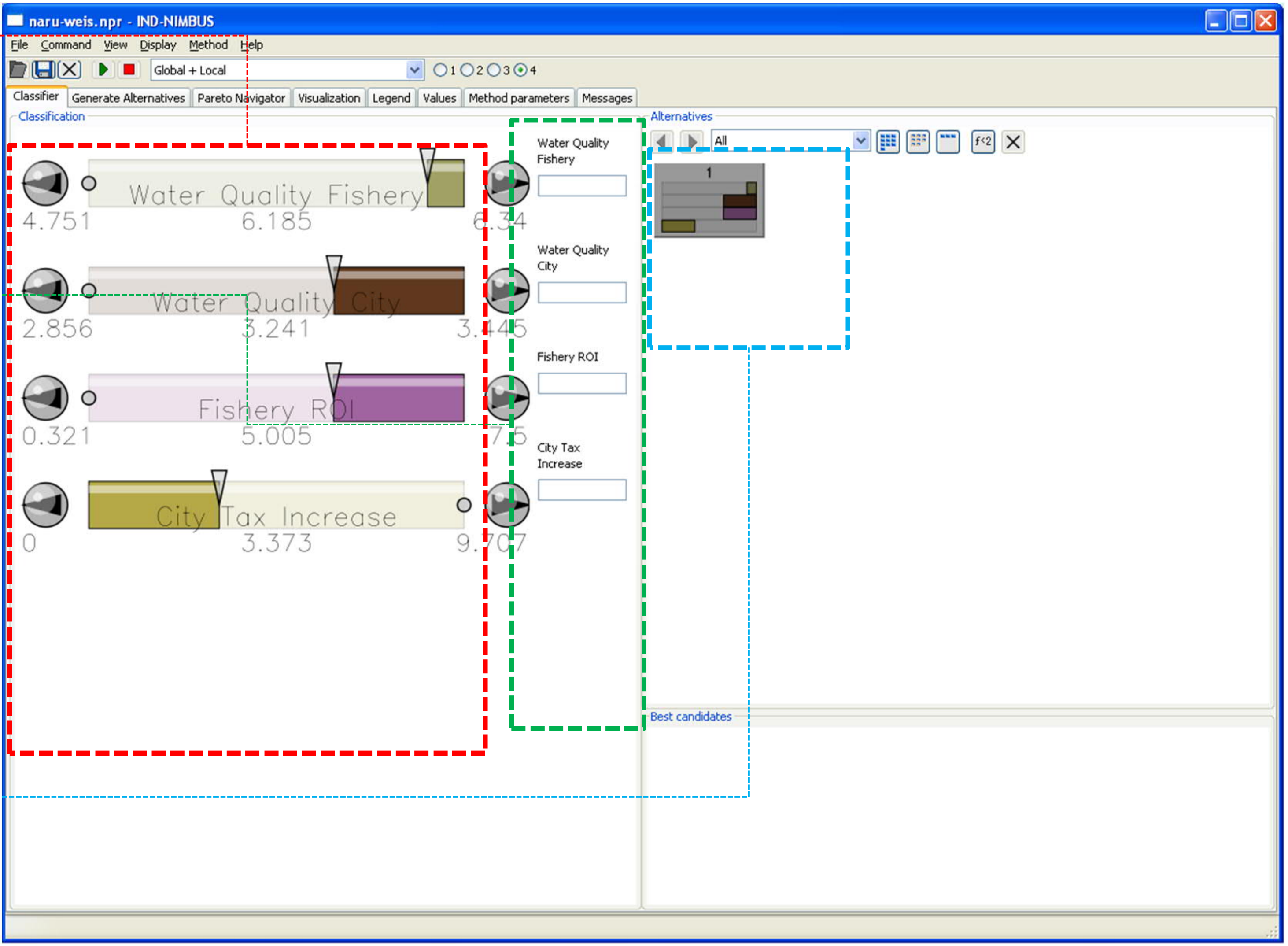
Document: Done


WWW-NIMBUS ## Kalvokuva1 - Netscape
File Edit View Go Communicator Help
NIMBUS
TUTORIAL
HELP
NIMBUS homepage
Visualizations
PETAL DIAGRAM OF THE ALTERNATIVES
Alternative 1
Alternative 2
Alternative 3
Alternative 4
Alternative 5
=f1
=f2
=f3
=f4
=f5
VALUE PATHS IN THE RELATIVE RANGE OF VALUES
Different alternatives have different colours
Alternative: 1
f 1 : -218.809
f 2 : -5.95752
f 3 : 0.00734
f 4 : -21.6788
f 5 : -52.3280
Colour:
------ : Current classificatio
NADIR
f1 -155.8544
f2 -0.99999
f3 25.03343
f4 3939.291
f5 0.0
ICV
-280.7645
-16.0
0.00171
-67.25
-407.2935
http://enimbus.it.jyu.fi/N3/index.html

IND-NIMBUS® Views

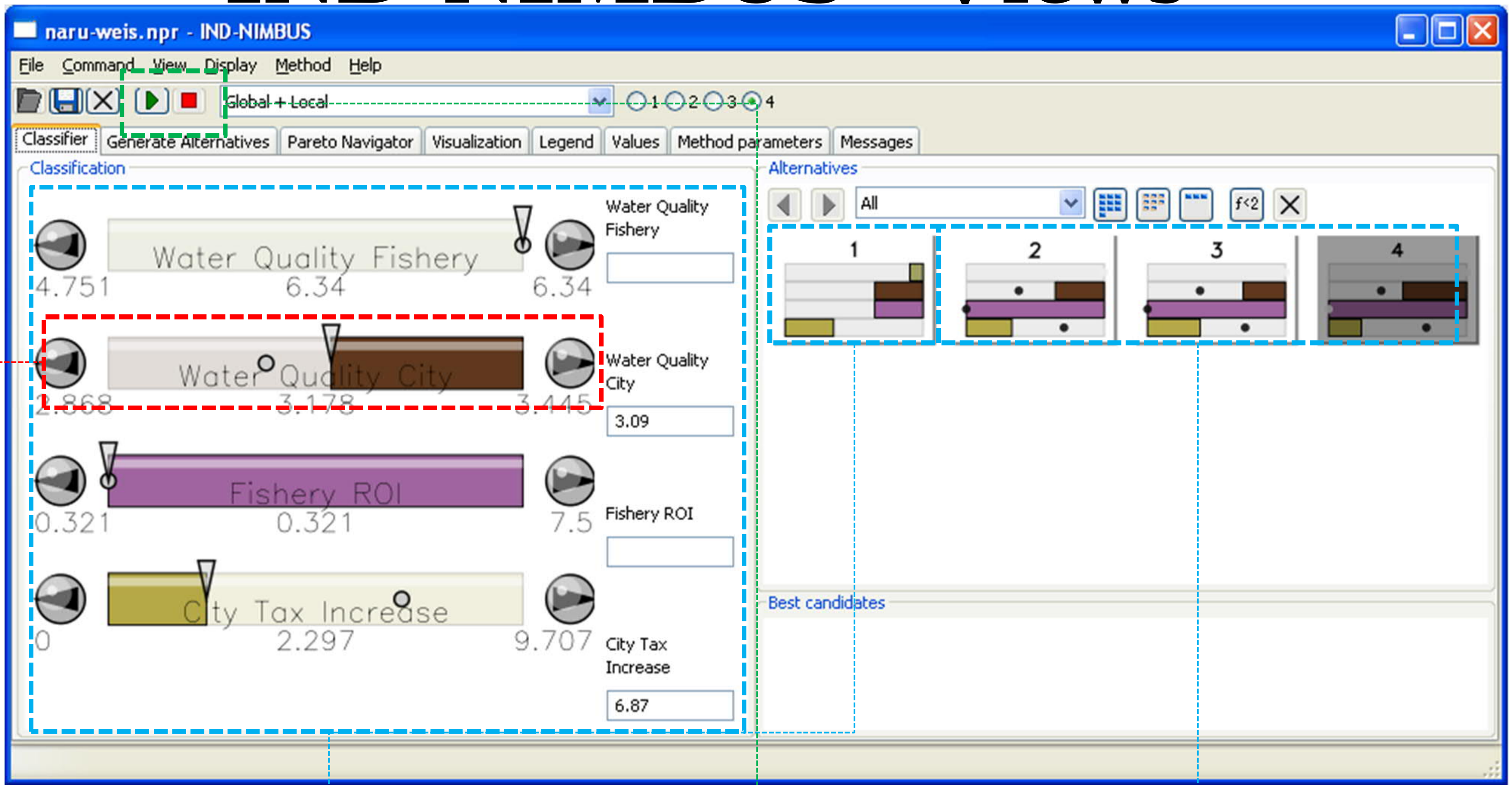
Objective function values and classification

Numerical classification boundaries



Initial PO solution

IND-NIMBUS® Views



NIMBUS classification by clicking objective bar

Initial PO solution

New solutions calculated with a play button

New PO solutions

Computational Challenges

of complex simulation-based optimization

We need tools for handling

- Computational cost
  - Objective and constraint functions depend on output of simulation models – may be time-consuming
- Black-box models
  - Global optimization needed -> computational cost
- One can train a computationally inexpensive surrogate (metamodel) to each expensive function but training is not straightforward and there are alternatives
- EMO methods for computationally expensive:
  - ParEGO, SMS-EGO, K-RVEA

Hybrid Methods

- Put together ideas of different methods to form new ones
- Aim: at the same time
  - combine strengths and benefits
  - avoid weaknesses
- A posteriori methods
  - information of whole PO set – possibilities and limitations
- Interactive methods
  - DM can **learn** about the problem, its interdependencies and adjust preferences
  - DM can concentrate on interesting solutions
  - computationally less costly
- Hybrids combining a posteriori and interactive methods

Pareto Navigator

Eskelinen et al., OR Spectrum (2010)

- Background & motivation
  - I Learning phase   II Decision phase
  - Challenges of computationally expensive problems
- Pareto optimal set = **actual PO set**
- Learning-oriented interactive method
- Hybrid method: first a posteriori and then interactive method (assume convexity)
  - relatively small) set of Pareto optimal solutions
  - polyhedral approximation of PO set in objective space – ***approximated PO set***
- Convenient and real-time navigation
  - Preference information: reference point
  - Project to actual PO set
- *Instead of approximating objective functions we directly approximate PO set*

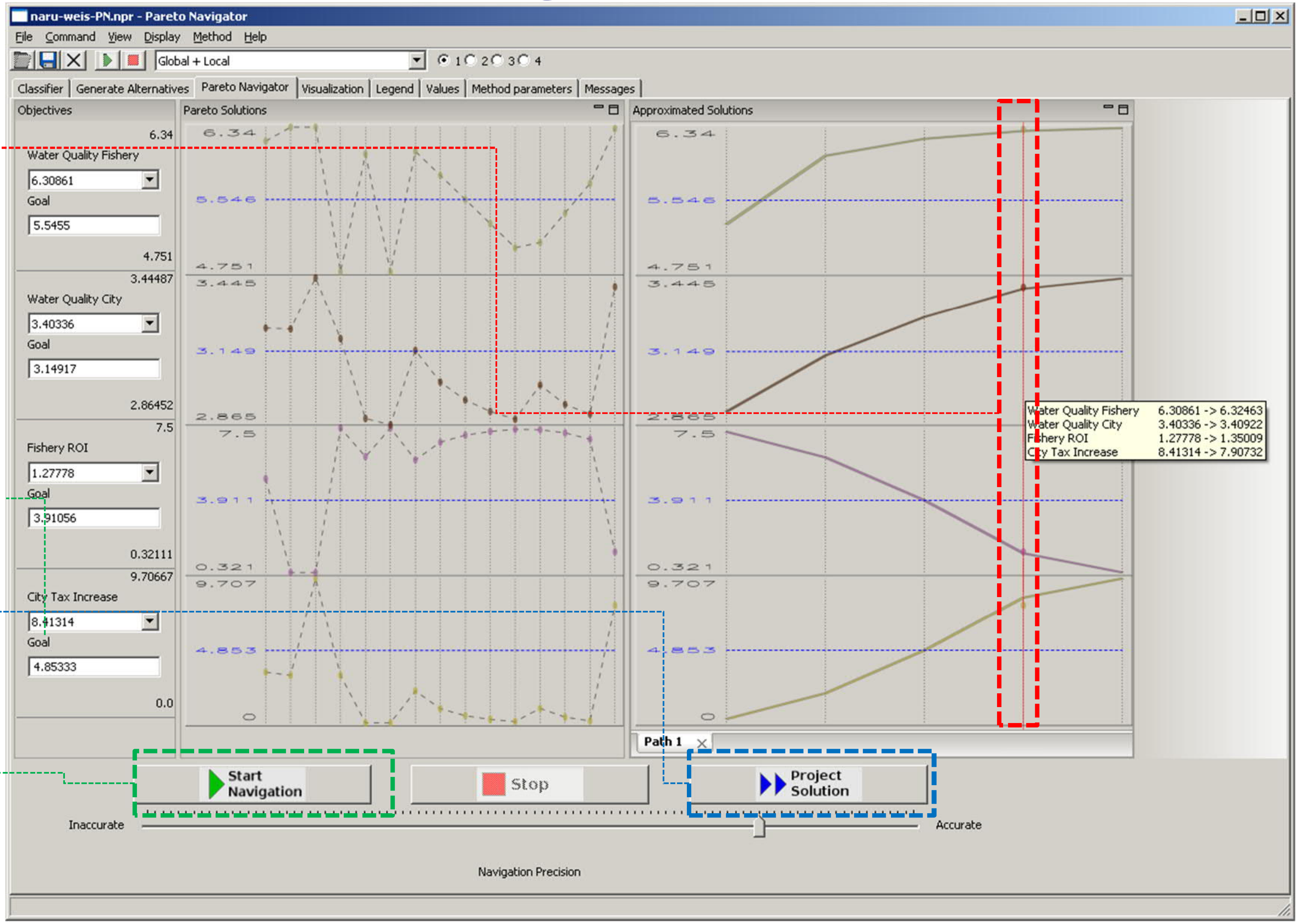
Pareto Navigator View

Based on the information given, new approximated PO solutions are generated

Approximated solutions can be used

to project them to real PO solutions

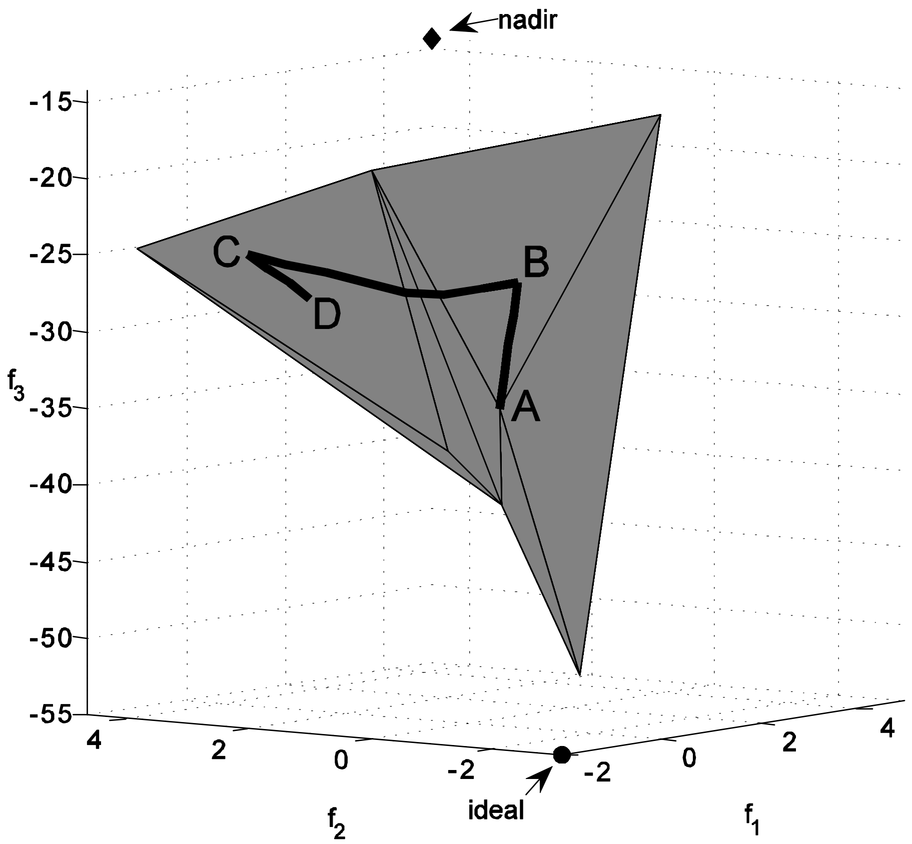
or as a starting point for new navigation

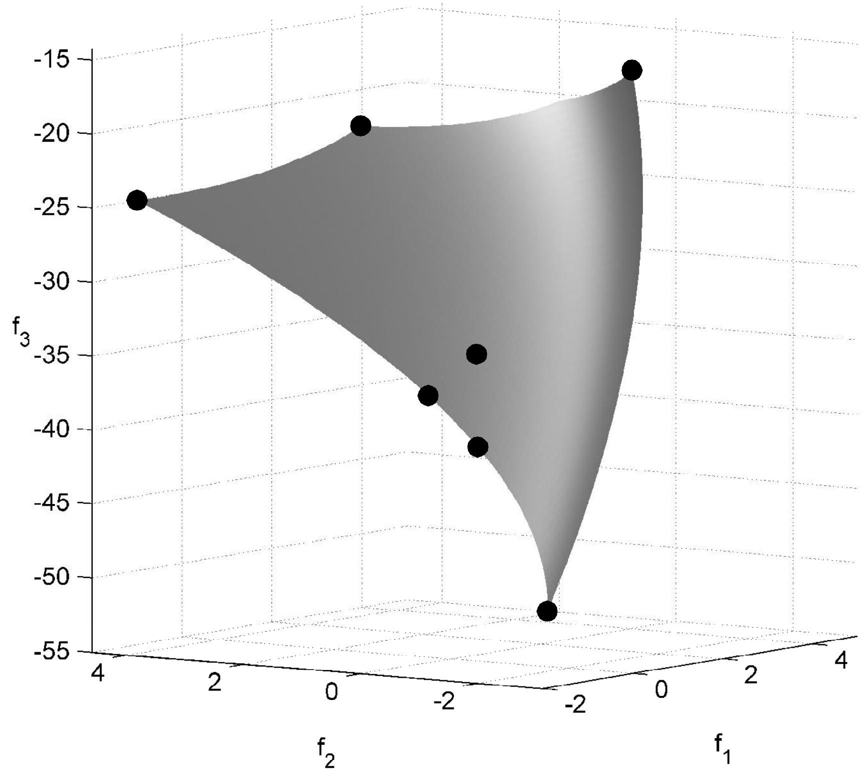


Example in 3D

- This is what happens in objective space during the solution process

  (polyhedral approximation and actual PO set)





NAUTILUS – Background

Miettinen et al., EJOR (2010)

- Challenge: typically methods deal with Pareto optimal solutions only
  - No other solutions are expected to be interesting for the DM
  - Trading off necessitated: impairment in some objective(s) must be allowed in order to get a new solution
- Past experiences affect DMs' hopes
  - DMs do not react symmetrically to gains and losses
  - Necessity of trading off (sacrifice) may hinder DM's willingness to move from the current PO solution
  - **Anchoring**: solutions considered may fix our expectations (DM fixes one's thinking on some (possible irrelevant) information
  - Time available for solution process limited
  - Choice of starting point may play a significant role

➢ Most preferred solution *may not be found*

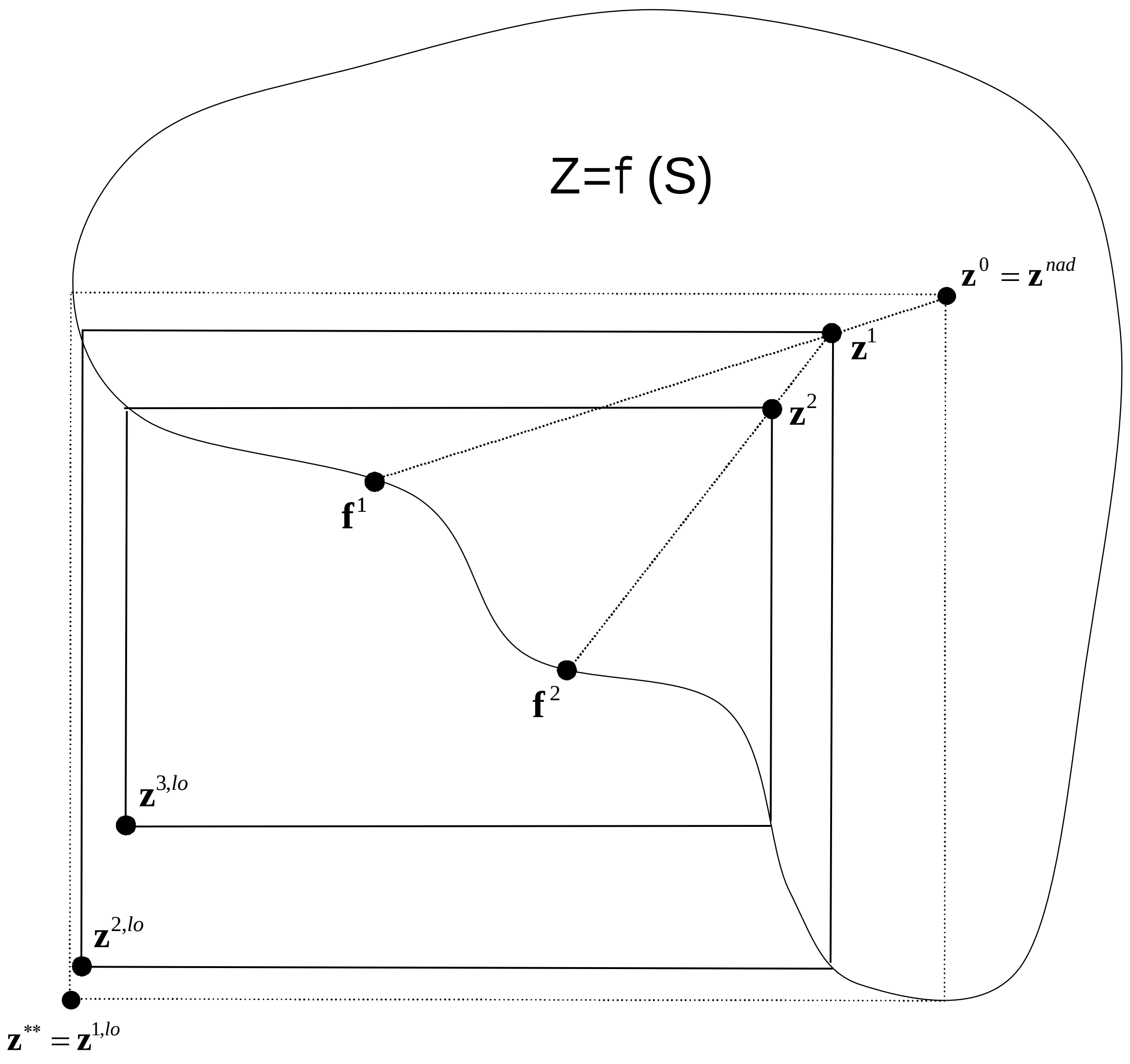
➢ Negotiation support for group decision making

  ➢ Negotiators easily anchor at starting Pareto optimal solution if it is advantageous for their interests

Idea of NAUTILUS

- DM starts from the worst e.g. nadir objective vector and moves towards PO set
  - Improvement in each objective at every iteration
  - Possible to gain at every iteration – no need for sacrifices
- At each iteration, objective vector obtained dominates the previous one
- Only the final solution is Pareto optimal
- DM can always go backwards if desired
- DM can approach any part of PO set (s)he wishes
- Different NAUTILUS variants use different ways of expressing preference information to form direction of simultaneous improvement
  - Ruiz et al, EJOR (2015)
  - Miettinen et al, JOGO (2015)
  - Miettinen, Ruiz, J Bus Econ (2016)

At each iteration range of reachable obj. function values shrinks

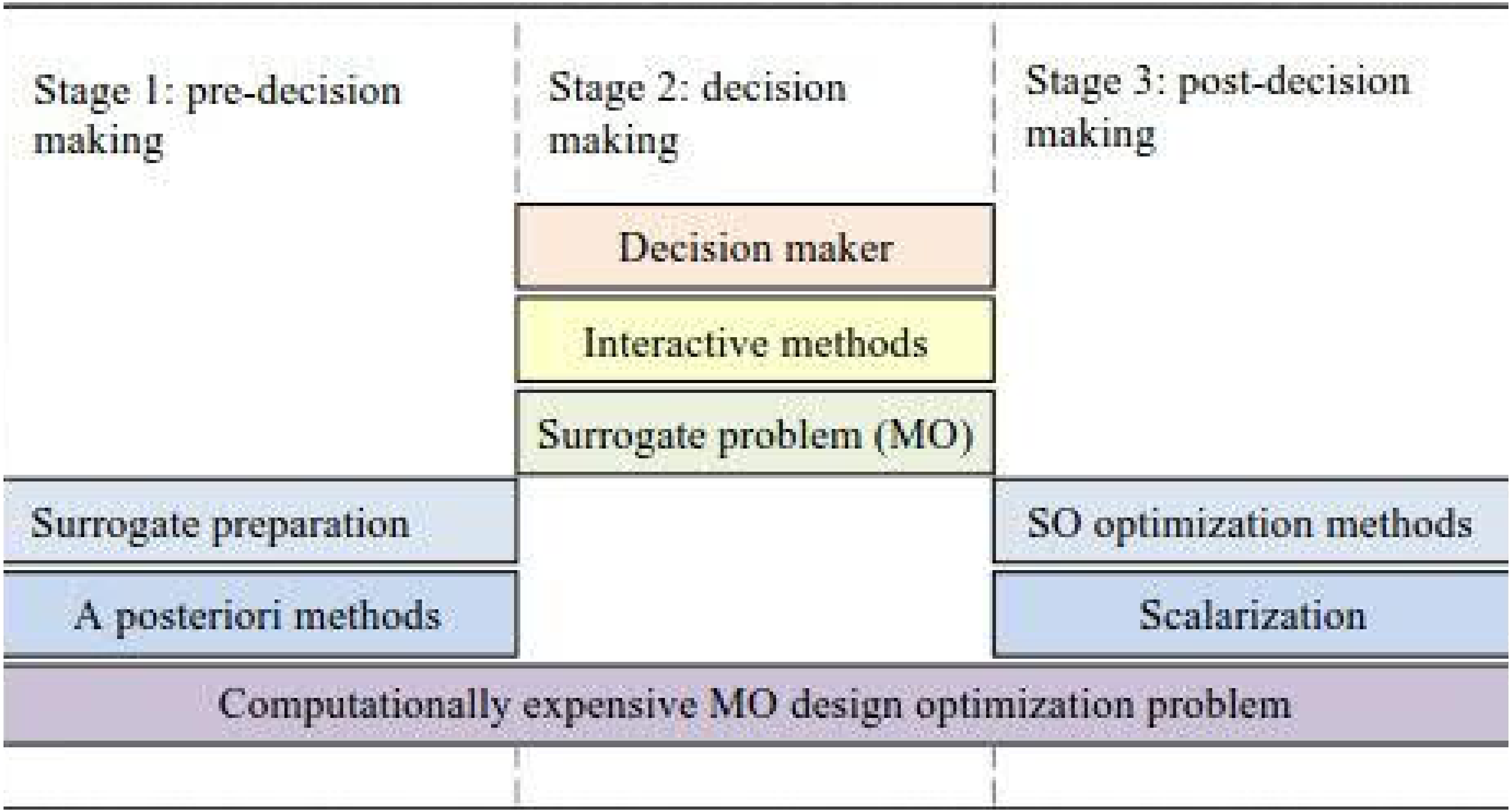


NAUTILUS - Remarks

- During the solution process, connection to decision variable space is temporarily lost
  - Iteration points generated are only defined in objective space
  - We *know* that a feasible solution and corresponding obj.vector **better than** the current vector exist
- Allows free search
- Avoids need of trading off – should allow the DM to learn better of what is available/possible
- Provides new perspective to solving multiobjective optimization problems
- Solution process can be continued with other (interactive) methods, if needed

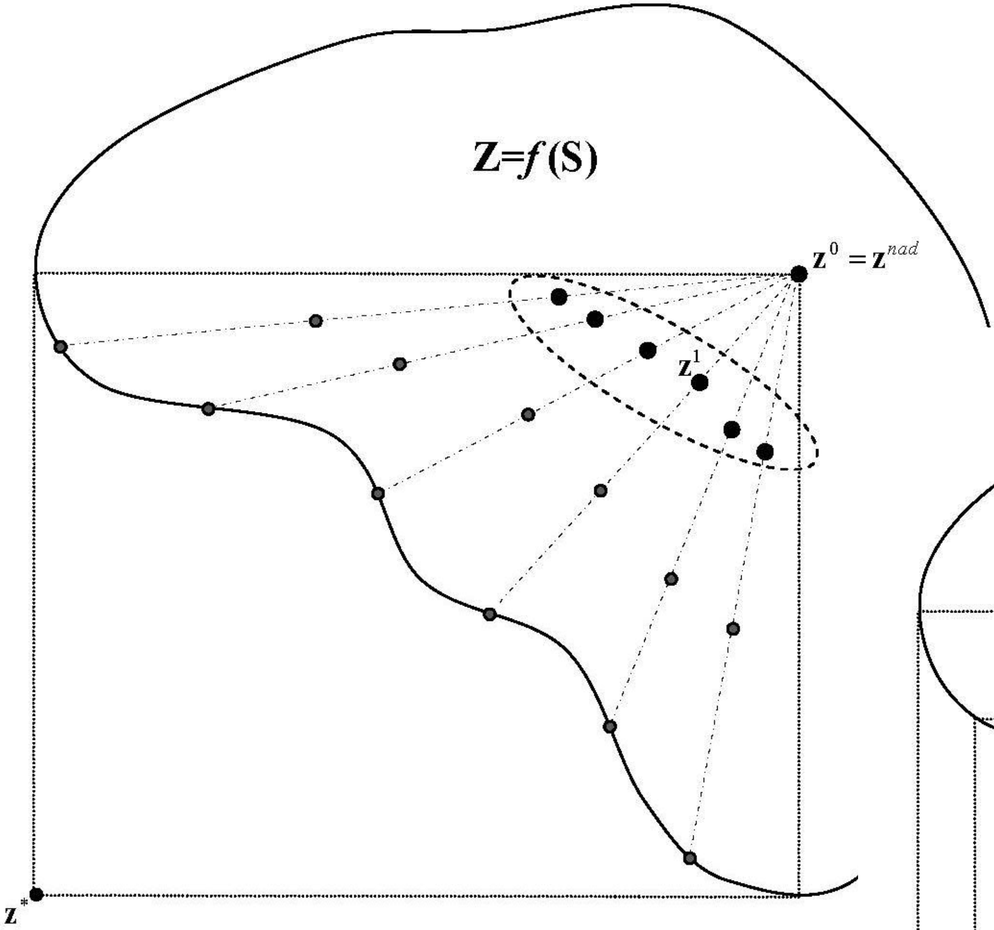
3-Stage Approach

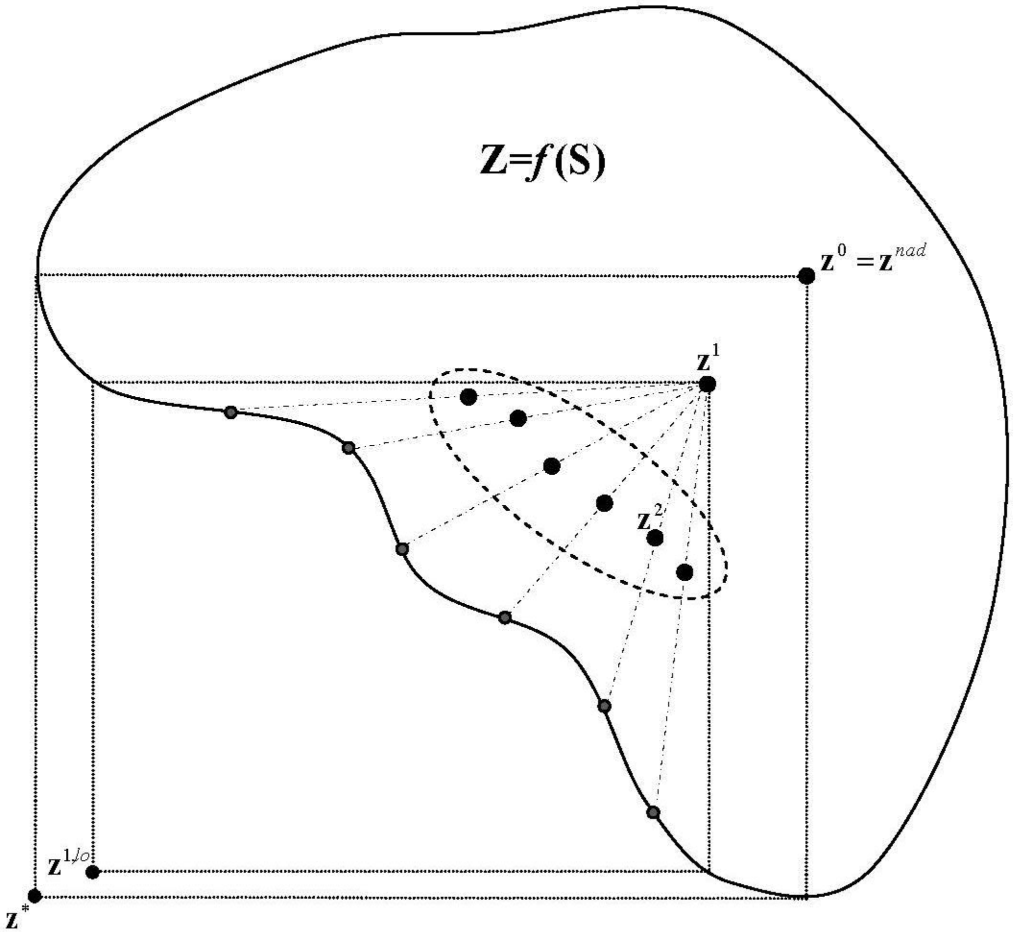
Steponavice et al., Computer-Aided Design (2014)



E-NAUTILUS

Ruiz et al., EJOR (2015





DM sets number of points to compare (here 6) and number of iterations (here 3)

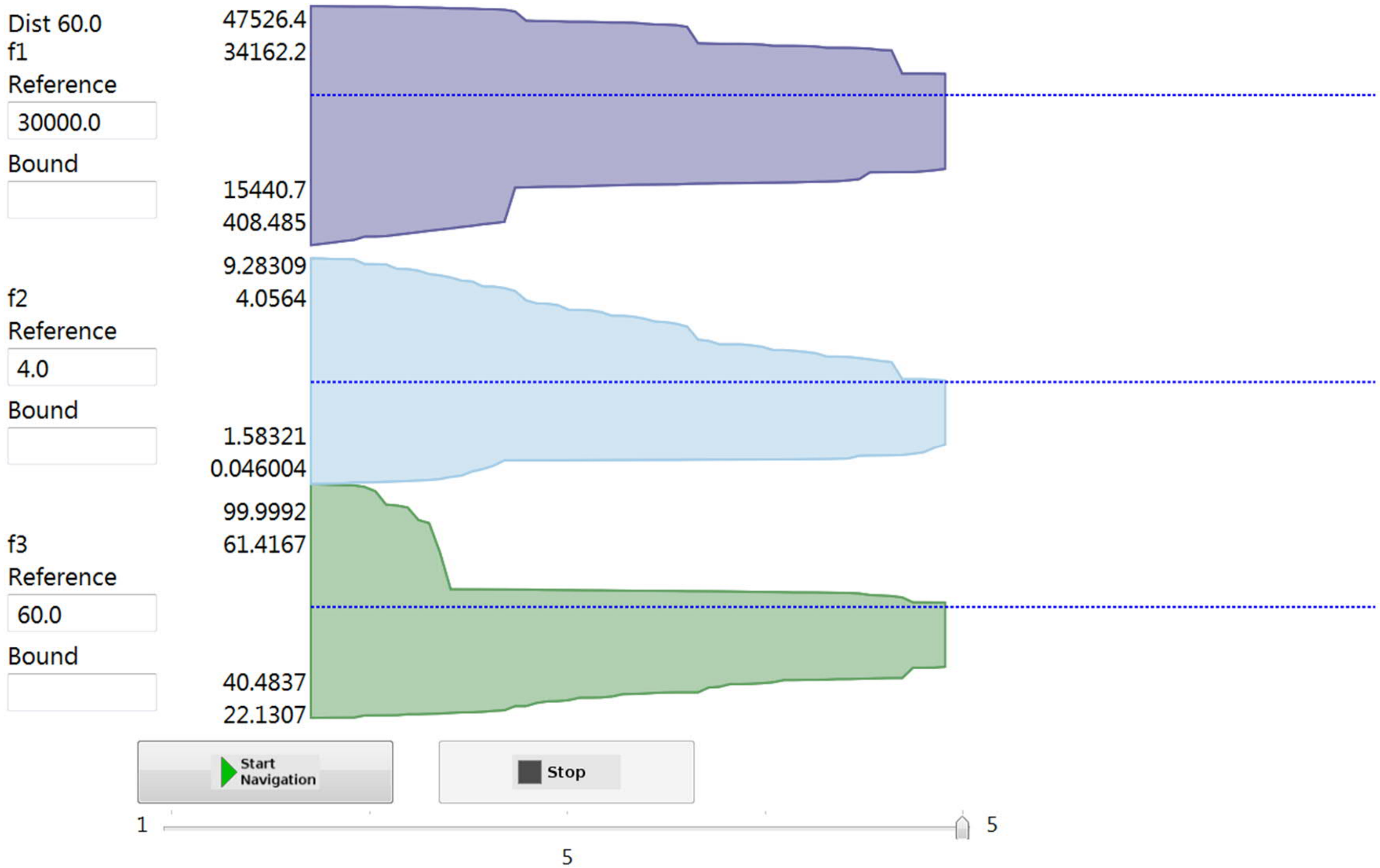
NAUTILUS Navigator

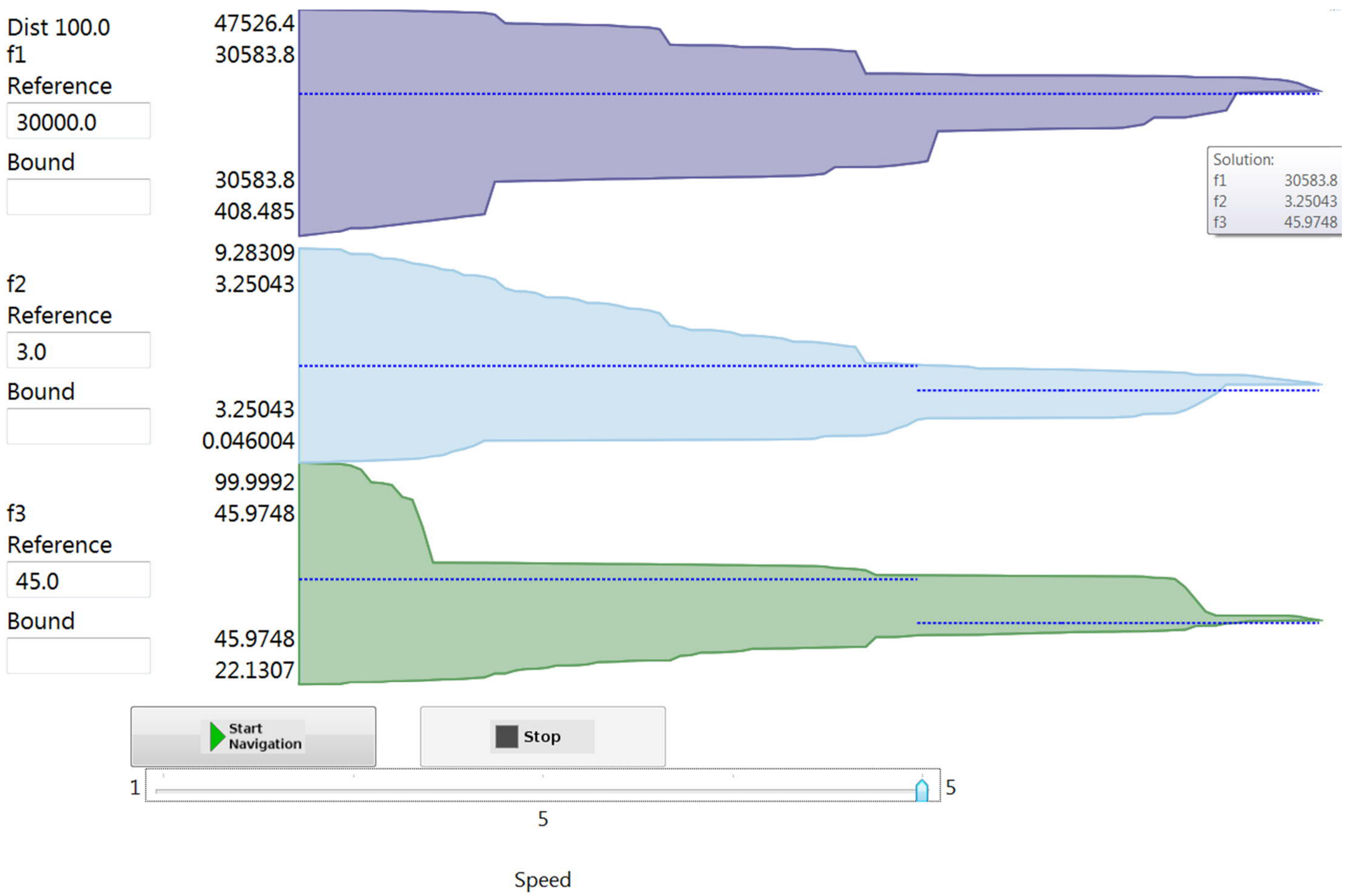
with A. B. Ruiz, F. Ruiz, V. Ojalehto

- Idea: DM can navigate from worst possible to most preferred objective function values
- *A priori:* Set of (approximated) PO solutions
  - Generated before involving DM
- *Interaction*: With NAUTILUS Navigator DM can navigate from inferior solution to most preferred one by gaining in all objective functions simultaneously, at each iteration
- *Preference information:* reference point (aspiration levels) and bounds not to be exceeded
- As solution process approaches set of PO solutions, ranges of objective function values that are still reachable without trading-off shrink and DM sees this in real time

NAUTILUS Navigator cont.

- GUI with *reachable range paths* consisting of two plot lines; lowest and highest reachable values from current iteration
- DM can see history, no need to remember it



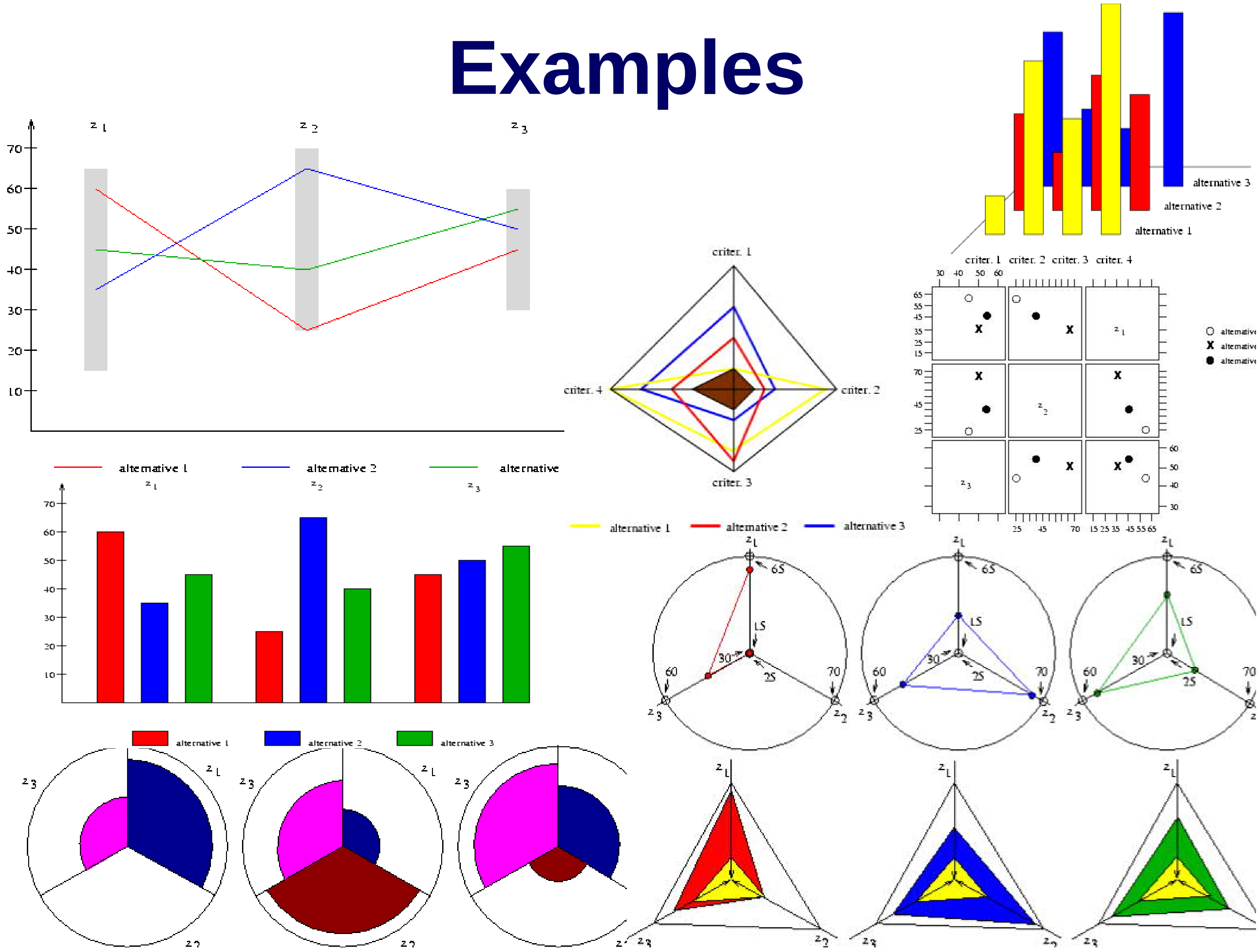


Dist 100.0
f1
Reference
30000.0
Bound
47526.4
30583.8
30583.8
408.485
Solution:
f1 30583.8
f2 3.25043
f3 45.9748
f2
Reference
3.0
Bound
9.28309
3.25043
3.25043
0.046004
f3
Reference
45.0
Bound
99.9992
45.9748
45.9748
22.1307
Start Navigation
Stop
1
5
5
Speed

On Visual Illustration
Miettinen, OR Spec (2014)

- The decision maker (DM) is often asked to compare several alternatives
  - e.g. within interactive methods
  - Graphs and table complement each other
- Illustration is difficult but important
  - easy to comprehend
  - important information should not be lost
  - no unintentional information should be included
  - makes it easier to see essential similarities and differences
- DMs have different cognitive styles

Examples



alternative 1
alternative 2
alternative 3
criter. 1
criter. 2
criter. 3
criter. 4

Experiences

- Collaboration with experts of problem domains
- **Positive experiences**
- DM receives a new perspective
  - can consider different objectives simultaneously, not one by one
  - interdependencies and interactions between objectives to be observed
  - DM learns about the conflicting qualitative properties
  - new insight to challenging and complex phenomena
- Experiences of DMs
  - methods easy to use – understandable questions
  - DM can find a satisfactory solution and be convinced of its goodness
  - confidence: best solution was found

Some Applications

- Chemical process design
  - Hakanen et al., JMCDA (2005), Appl Therm Eng (2006)
- Two-stage separation process
  - Sindhya et al., Exp Syst with Appl (2014)
- Heat Exchanger Network Synthesis
  - Laukkanen et al., Computers and Chem Eng (2010), Appl Therm Eng (2012)
- Brachytherapy planning
  - Ruotsalainen et al., Phys Med Biol (2010)
- Wastewater Treatment Planning
  - Hakanen et al., DSS (2011), Env Mod & Softw (2013)
- Design and Operation of Paper Maching
  - Steponavice et al., Comp-Aided Design (2014)
- Data-based Forest Management with Uncertainties
  - Hartikainen et al., Proceed. MOD 2016
- Design of Permanent Magnet Synchronous Generator
  - Sindhya et al. IEEE Trans Ind Elect (to appear)
- Design of air intake ventilation system in tractor cabin
  - Chugh et al. Proceed. CEC 2017 - best student paper

Furthermore

- *Open source* framework DESDEO wit interactive methods – try it!
  - **desdeo.it.jyu.fi**
- **Decision analytics** - data driven decision support – thematic research area: DEMO
  - Instead of models we have data available
  - Applications incl. forest treatment planning, inventory management and punishing criminals
  - http://www.jyu.fi/demo
- We welcome visitors!
- Open PhD student positions twice a year
- **EMO2019: www.emo2019.org/**

Conclusions

- **Compromise *is better* than optimum!**
- Plenty of real-life applications are waiting for us and provide various challenges!
- Hybridization of different methods offers a lot of potential
- Book aims at bringing MCDM and EMO fields closer to each other:

  Branke, Deb, Miettinen, Slowinski (Eds.): *Multiobjective Optimization: Interactive and Evolutionary Approaches*, Springer-Verlag, 2008
- Method selection depends e.g. on
  - Properties of problem
  - Availability of DM
  - Preference information type comfortable for DM

Acknowledgements

- Collaboration: coauthors and Industrial Optimization Group http://www.mit.jyu.fi/optgroup/
- Funding: Partly Academy of Finland, Tekes: Finnish Funding Agency for Innovation & companies