Introduction to Some Methods and Applications of Nonlinear Multiobjective Optimization

Prof. Kaisa Miettinen
kaisa.miettinen@jyu.fi
Department of Math. Inform. Technology
University of Jyväskylä, Finland
http://www.mit.jyu.fi/miettine/
Industrial Optimization Group
http://www.mit.jyu.fi/optgroup/

Problems with Multiple Criteria

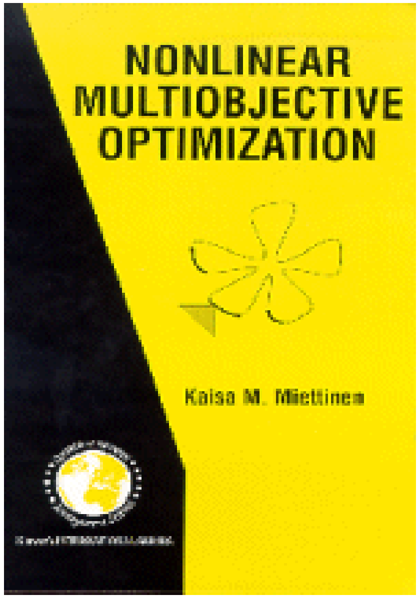
- **Finding the best possible compromise**
- **Different features**
- **One decision maker (DM) – several DMs**
- **Deterministic – stochastic**
- **Continuous – discrete**
- **Nonlinear – linear**

➔ **Nonlinear multiobjective optimization**

Contents

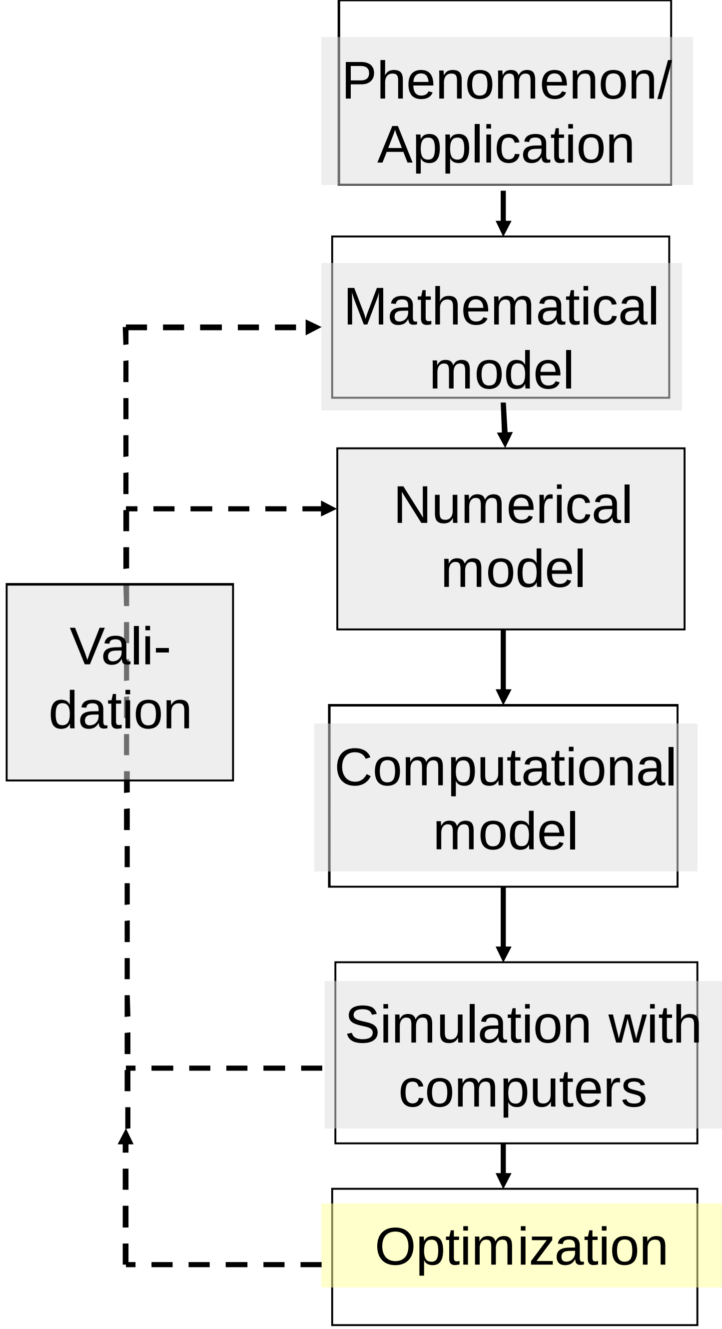
- *Nonlinear Multiobjective Optimization* by

  Kaisa M. Miettinen, Kluwer (Springer), Boston, 1999
- Concepts
- Optimality
- Methods (4 classes)
- Tree diagram of methods
- Graphical illustrations
- Some new methods
- Applications
- Conclusions



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



Decision Making

- Some background
  - Problems are more and more complex
  - Individuals, groups and organizations need better decisions
  - Complicated interdependencies hard to handle for humans without decision support
- **Optimization** provides systematic and analytic ways to find the best possible solution (according to the criterion used)
- Optimization is NOT what-if analysis or trying a few solutions and selecting the best of them
- Operations research is a scientific approach to decision making (Saul I. Gass) – applying mathematical and other techniques in decision problems in business, industry, government, military etc.

Multiobjective Optimization

- Most real-life problems have **several conflicting objectives** to be considered simultaneously
- It is not ok to use typical approaches to
  - convert all but one into constraints in the modelling phase or
  - invent weights for the objectives & optimize the weighted sum
  - simplify the consideration and **lose information** in this way
- 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
  - Needed in strategic, operative and decision making in general
  - The feasible region may turn out to be empty -> minimize constraint violations

Challenges

- Hidden needs in various application fields
  - How to attract interest and raise awareness
- We must continuously **question** current practices
  - New technologies enable revolutionary approaches
- We need tools for handling complexity
- Computational efficiency is still important
- Multidisciplinary: mathematics, information technology, numerical methods, usability, good links to specific applications, etc.

Concepts

We consider multiobjective optimization problems

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

in other words

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

where

f_i: S→R = *objective function*

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

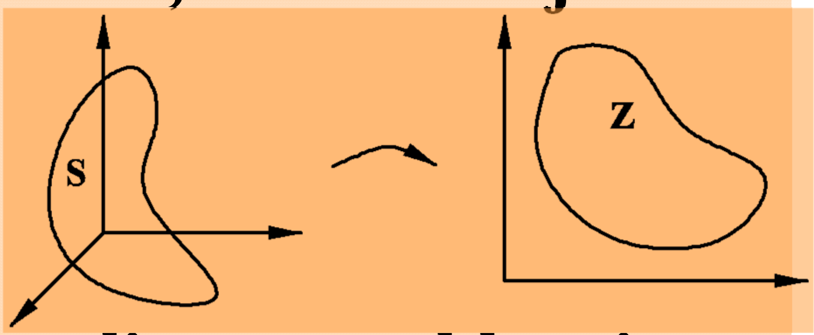
x = *decision vector* (of n *decision variables* x_i)

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

``minimize´´ = minimize the objective functions simultaneously

Concepts cont.

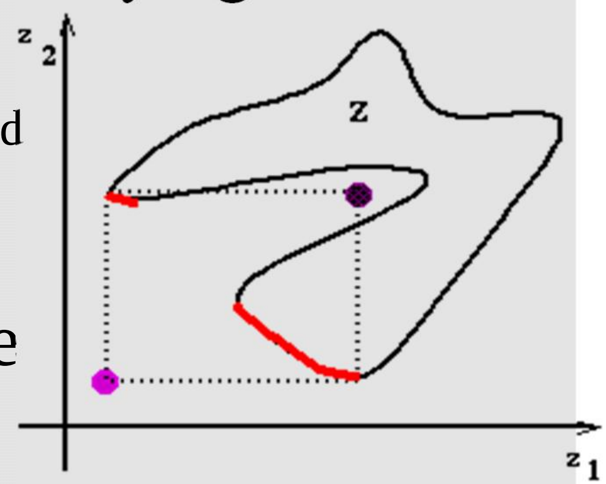
- **S consists of linear, nonlinear and box constraints for the variables**
- **We denote *objective function values* by $z_i = f_i(x)$**
- **$z = (z_1,\ldots, z_k)$ is an *objective vector***
- **$Z \subset R^k$ denotes the image of S; feasible objective region. Thus $z \in Z$**



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

Concepts cont.

- A *decision maker* (DM) is needed. (S)he has insight into the problem and can express preference relations
- Multiobjective optimization = help DM in finding most preferred solution
  - We need preference information from DM
- An *analyst* is responsible for the mathematical side
- *Solution process* = finding a solution
- *Final solution* = feasible PO solution satisfying the DM
- Ranges of the PO set: *ideal objective vector* $z^{\star}$, approximated *nadir point* z^{nad}
- *Ideal objective vector* = individual optima of each f_i and utopian objective vector $z^{\star\star}$ is strictly better than $z^{\star}$

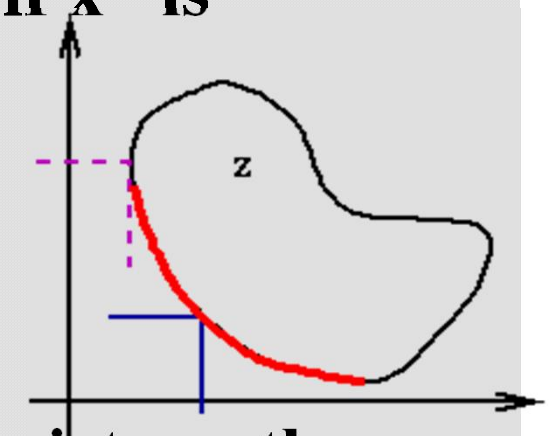


Optimality

- Contradiction and possible incommensurability $\Rightarrow$
- $x^* \in S$ is (globally) *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

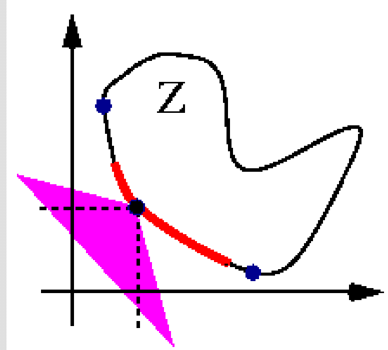
  i.e. $(z^* - R^k_+ \setminus \{0\}) \cap Z = \emptyset$,

  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

Optimality cont.

- 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: We know that 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*.**

Marginal Rate of Substitution

- x^1 and x^2 are *indifferent* if they are equally desirable to the DM.
- Definition: A *marginal rate of substitution* $m_{ij}=m_{ij}(x^*)$ is the amount of decrement in f_i that compensates the DM for one-unit increment in f_j, while all the other objectives remain unaltered.
- For continuously differentiable functions we have

$$\lambda_{ij} = \frac{d\, f_i(\mathbf{x}^*)}{d\, f_j(\mathbf{x}^*)} \quad \text{and} \quad m_{ij} = \frac{d\, U(\mathbf{z}^*))}{d\, z_j} \Big/ \frac{d\, U(\mathbf{z}^*))}{d\, z_i}.$$

Final Solution

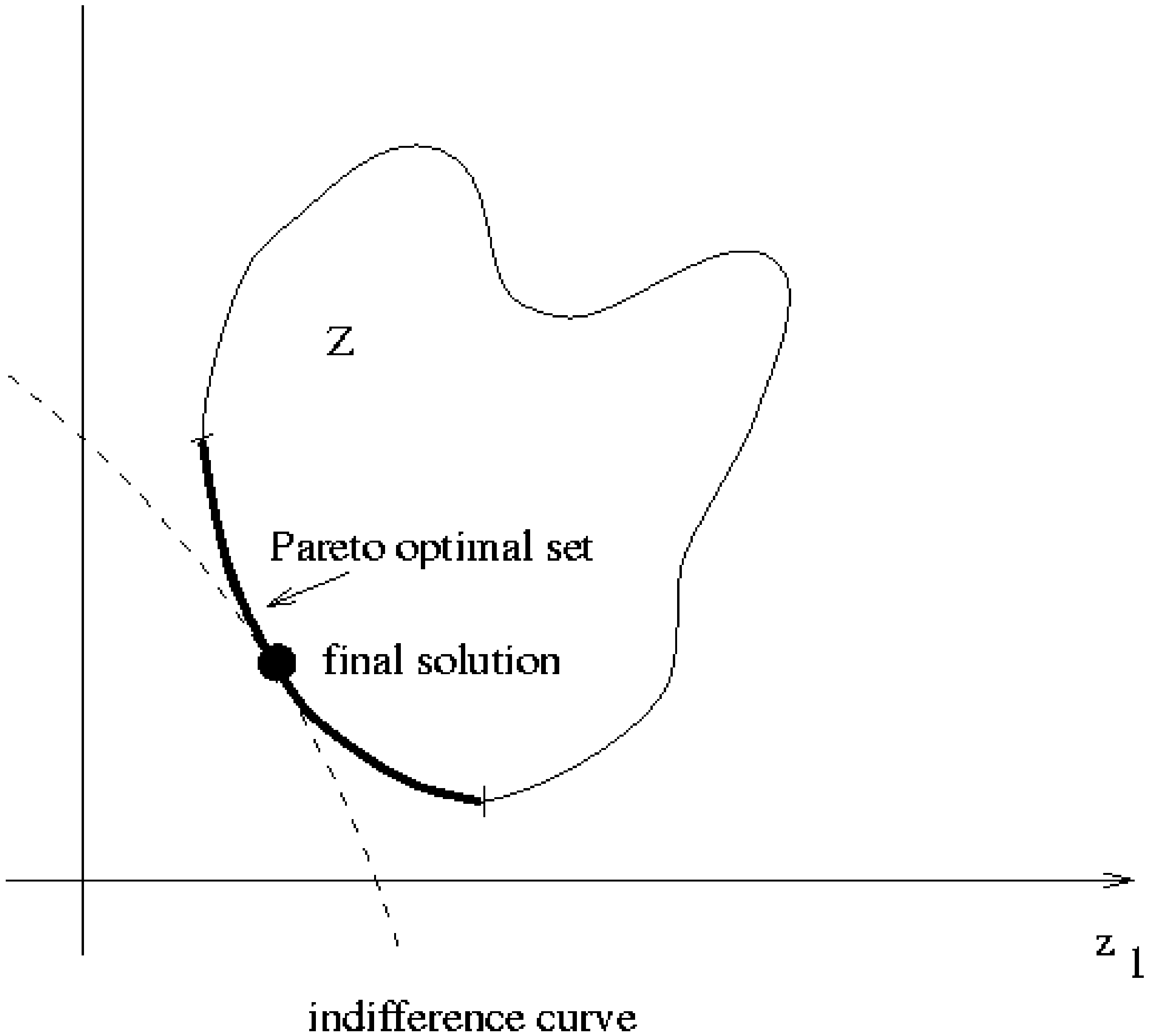


Figure 1. The final solution.

Testing Pareto Optimality

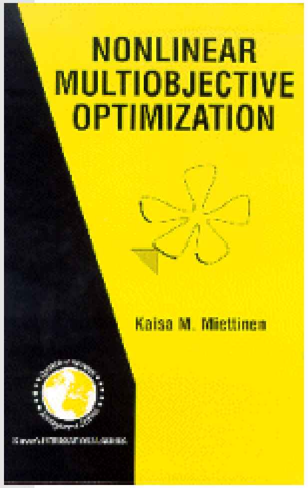
- **x* is PO if and only if**

$$
\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{k} \varepsilon_i \\
\text{subject to} \quad & f_i(\mathbf{x}) + \varepsilon_i = f_i(\mathbf{x}^*) \text{ for all } i = 1, \ldots, k, \\
& \varepsilon_i \geq 0 \text{ for all } i = 1, \ldots, k, \\
& \mathbf{x} \in S.
\end{aligned}
$$

has an optimal objective function value 0. Otherwise, the solution obtained is PO

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



Four Classes of Methods

- How to support DM?
- ***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

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

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:

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

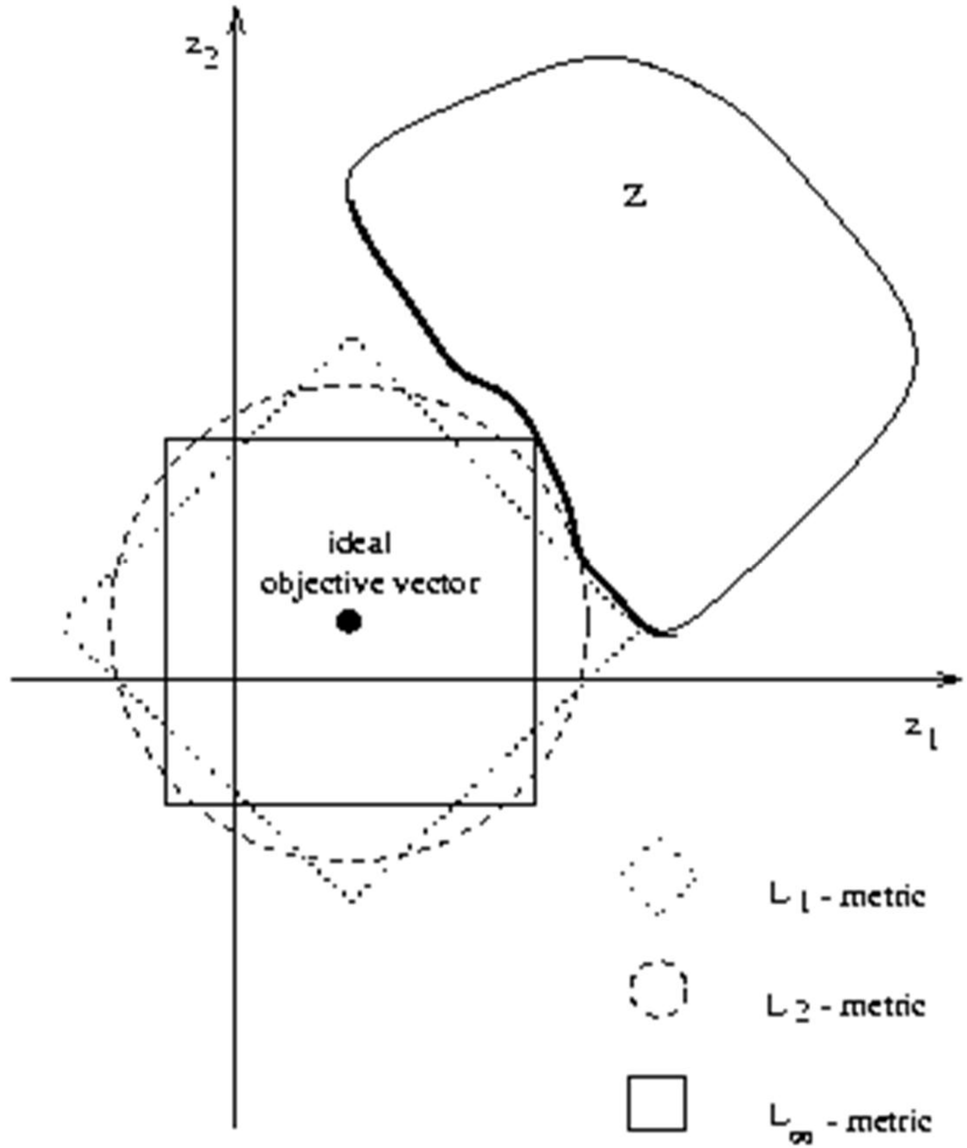
Method of Global Criterion cont.

? 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**

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

≈ **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**

Why not Weighting Method

Selecting a wife (maximization problem):

	beauty	cooking	house-wifery	tidi-ness	
Mary	1	10	10	10	
Jane	5	5	5	5	
Carol	10	1	1	1	

Idea originally from Prof. Pekka Korhonen

Why not Weighting Method

Selecting a wife (maximization problem):

	beauty	cooking	house-wifery	tidi-ness	
Mary	1	10	10	10	
Jane	5	5	5	5	
Carol	10	1	1	1	
weights	**0.4**	**0.2**	**0.2**	**0.2**	

Why not Weighting Method

Selecting a wife (maximization problem):

	beauty	cooking	house-wifery	tidi-ness	**results**
Mary	1	10	10	10	**6.4**
Jane	5	5	5	5	**5**
Carol	10	1	1	1	**4.6**
weights	0.4	0.2	0.2	0.2	

Weighting Method cont.

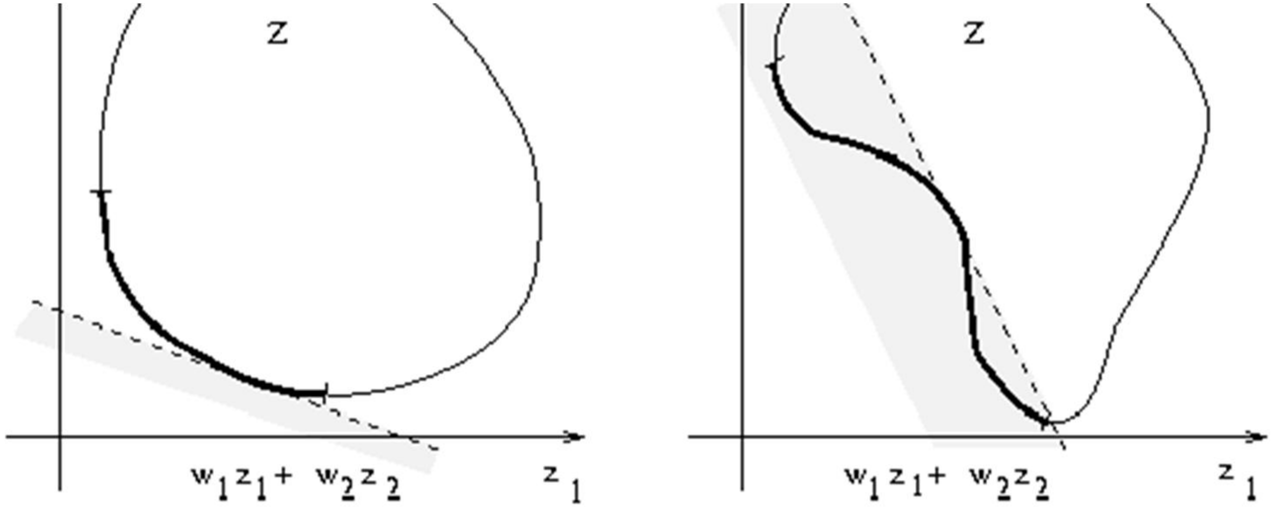


Figure 3. Convex and nonconvex problems.

- **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, $j \neq l$) 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 $S = \{x \in R^n \mid g(x) = (g_1(x),\ldots, g_m(x))^T \le 0\}$**
- ***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**

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

and

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

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 $p<\infty$
✿

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

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 ρ>0 sufficiently small.

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

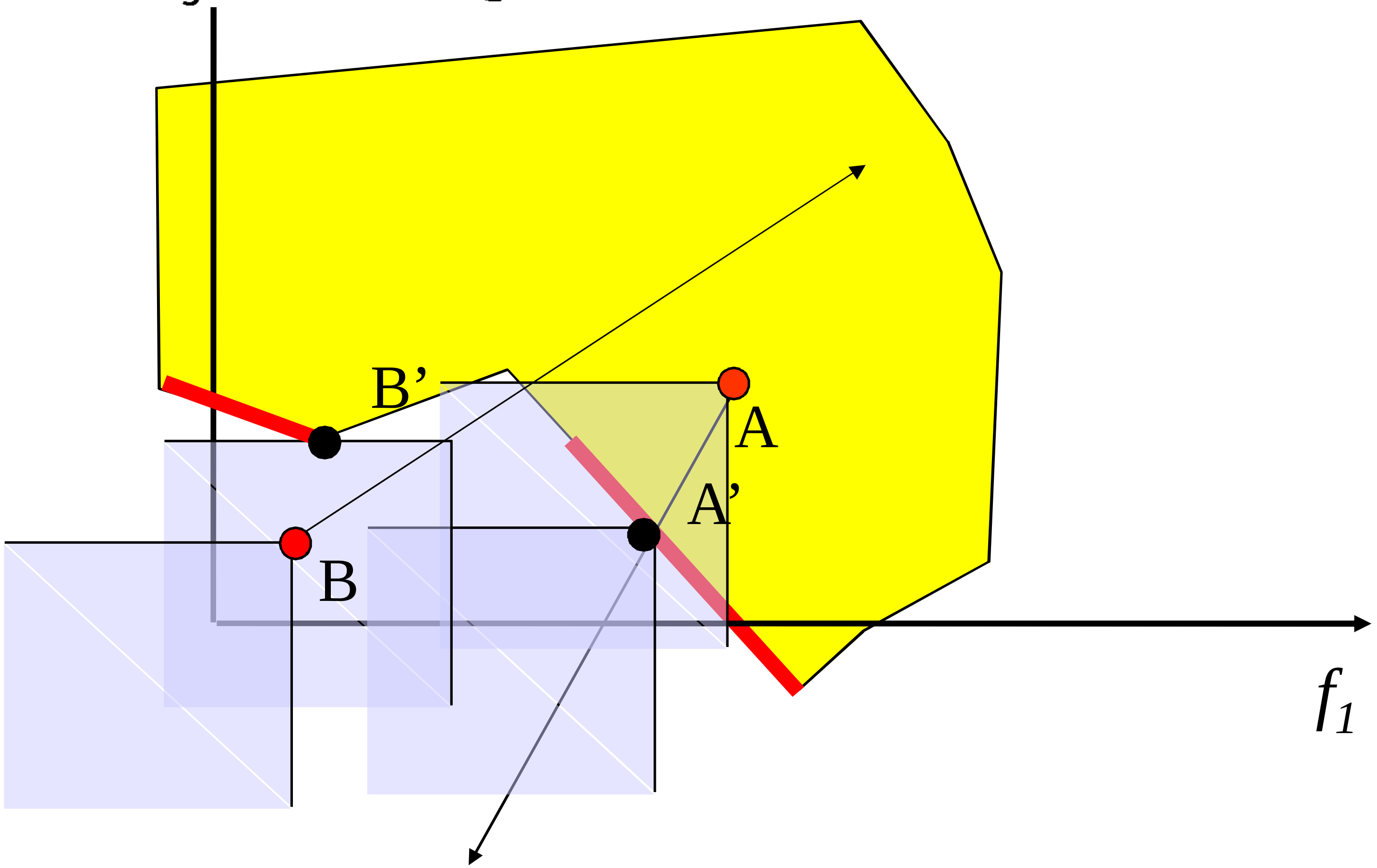
Achievement Scalar. Fun. cont.

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

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

Achievement Scalarizing Function:

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



Two Worlds

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 within the approximation
- Not too much emphasis on DM's preferences so far
- Guaranteeing actual optimality not always clear
- E.g. nonconvexity and discontinuity cause no difficulties
- Background in applications
- Many benchmark problems

Multiobj. Evolutionary Algorithms

- Many different approaches
- VEGA, RWGA, MOGA, NSGA, NSGA II, DPGA, etc.
- Goals: maintaining diversity and guaranteeing Pareto optimality – how to measure?
- Special operators have been introduced, fitness evaluated in many different ways etc.
- Works best if k=2

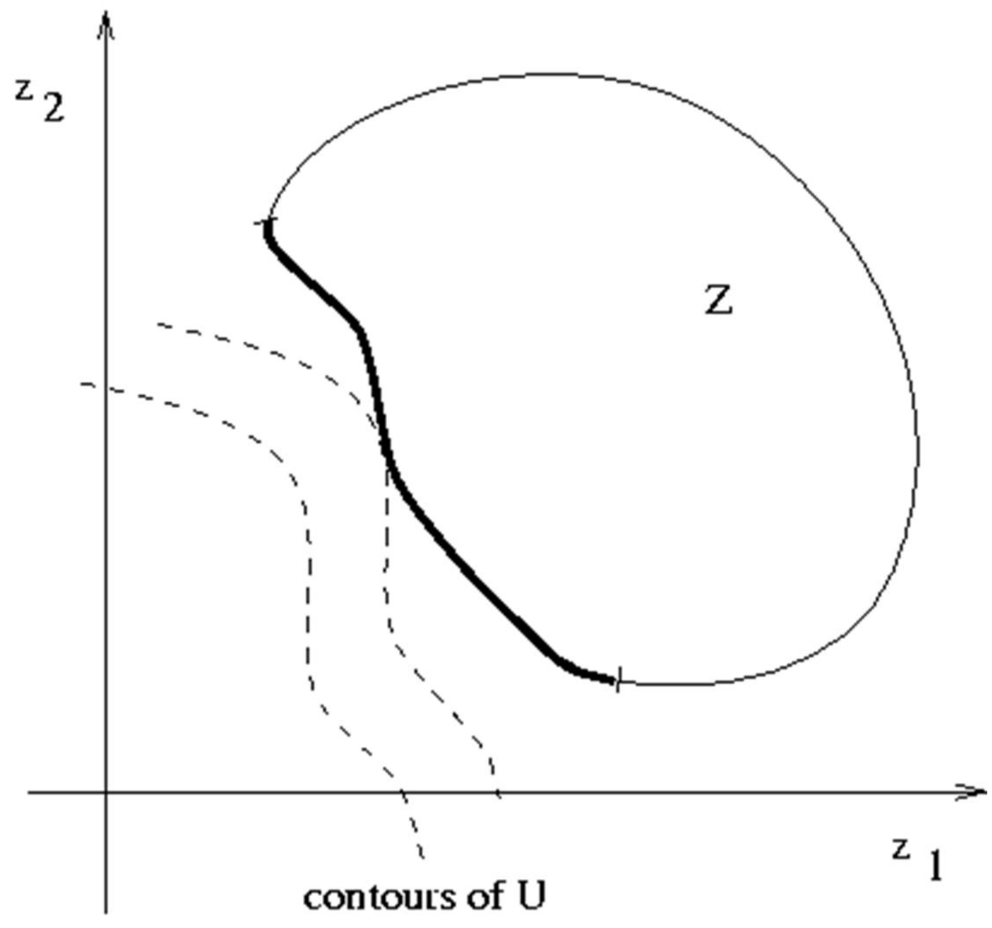
A Priori Methods

- **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**

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



Value Function Method cont.

- + If U represents the global preference structure of the DM, the solution obtained is the ``best´´
- + The solution is PO if U is strongly decreasing
- – It is very difficult for the DM to specify the mathematical formulation of her or his U
- – Existence sets consistency and other requirements
- – Even if the explicit U were known, the DM may have doubts or change preferences
- – U can not represent intransitivity/incomparability
- + Implicit value functions are important for theoretical convergence results of many methods

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**

Goal Programming (Charnes, Cooper)

- The DM must specify an aspiration level $\check{z}_i$ for each objective function
- f_i + an aspiration level = a *goal*. Deviations from aspiration levels are minimized
- The deviations can be represented as overachievements $\delta_i = \max\,[0, f_i(x) - \check{z}_i]$

➔ **Weighted approach:** with x and δ_i (i=1,…,k) as variables.

$$\begin{aligned} &\text{minimize} && \sum_{i=1}^{k} w_i \delta_i \\ &\text{subject to} && f_i(\mathbf{x}) - \delta_i \le \bar{z}_i, \quad i = 1, \ldots, k, \\ &&& \delta_i \ge 0, \quad i = 1, \ldots, k, \\ &&& \mathbf{x} \in S \end{aligned}$$

- Weights from the DM

- Not always PO

Goal Programming cont.

- <u>Lexicographic approach</u>: **the deviational variables are minimized lexicographically**
- <u>Combination</u>: **a weighted sum of deviations is minimized in each priority class**

+ **The solution is Pareto optimal if the reference point is or the deviations are all positive**
+ **Goal programming is widely used for its simplicity**

– **The solution may not be PO if the aspiration levels are not selected carefully**

– **Specifying weights or lex. orderings may be difficult**

– **Implicit assumption: it is equally easy for the DM to let something increase a little if (s)he has got little of it and if (s)he has got much of it**

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!

Examples of Forms of Interaction

- Opinions about trade-off rates, marginal rates of substitution
- Selecting one from a sample of PO solutions
- Reference point
- Classification
- Luque et al., OR Spectrum (2011), Ruiz et al, Annals of OR (2012)

Interactive Surrogate Worth Trade-Off (ISWT) Method (Chankong, Haimes)

- **Idea: Approximate (implicit) U by surrogate worth values using trade-offs of the ε-constraint method**
- **Assumptions:**
  - **continuously differentiable U is implicitly known**
  - **functions are twice continuously differentiable**
  - **S is compact and trade-off information is available**
- **KKT multipliers $\lambda_{li} > 0$ for all *i* are partial trade-off rates between f_l and f_i**
- **For all *i* the DM is told: ``If the value of f_l is decreased by λ_{li}, the value of f_i is increased by one unit or vice versa while other values are unaltered.´´**
- **The DM must tell the desirability with an integer [10,-10] (or [2,-2]) called *surrogate worth* value**

ISWT Algorithm

1) Select f_l to be minimized and give upper bounds
2) Solve the ε-constraint problem.Trade-off information is obtained from the KKT-multipliers
3) Ask the opinions of the DM with respect to the trade-off rates at the current solution
4) If some stopping criterion is satisfied, stop. Otherwise, update the upper bounds of the objective functions with the help of the answers obtained in 3) and solve several ε-constraint problems to determine an appropriate step-size. Let the DM choose the most preferred alternative. Go to 3)

ISWT Method cont.

- Thus: direction of the steepest ascent of U is approximated by the surrogate worth values
- Non ad hoc method
- DM must specify surrogate worth values and compare alternatives

! The role of f_l is important and it should be chosen carefully

! The DM must understand the meaning of trade-offs well

! Easiness of comparison depends on k and the DM.

– It may be difficult for the DM to specify consistent surrogate worth values

\+ All the solutions handled are Pareto optimal

Geoffrion-Dyer-Feinberg (GDF) Method

- Idea: Maximize the DM's (implicit) value function with a suitable gradient method
- Local approximations of the value function are made using marginal rates of substitution m_i (DM gives)
- Assumptions
  - U is implicitly known, continuously differentiable and concave in S
  - objectives are continuously differentiable
  - S is convex and compact
- The gradient of U at x^h:

$$\nabla_x U(f_1(\mathbf{x}^h), \ldots, f_k(\mathbf{x}^h)) = \sum_{i=1}^{k} \left(\frac{\partial U}{\partial f_i}\right) \nabla_x f_i(\mathbf{x}^h),$$

- The direction of the gradient of U:

$$\sum_{i=1}^{k} -m_i \nabla_x f_i(\mathbf{x}^h),$$

GDF Algorithm

1) Ask the DM to select the reference function f_l. Choose a feasible starting point z^1. Set h=1.
2) Ask the DM to specify k-1 marginal rates of substitution between f_l and other objectives at z^h.
3) Solve the problem. Set the search direction d^h. If $d^h = 0$, stop.
4) Determine with the help of the DM the appropriate step-size into the direction d^h . Denote the corresponding solution by $z^{h+1.}$
5) Set h=h+1. If the DM wants to continue, go to 2). Otherwise, stop.

GDF Method cont.

- ! The role of the function f_l is significant.
- Non ad hoc method
- DM must specify marginal rates of substitution and compare alternatives
  - The solutions to be compared are not necessarily Pareto optimal
  - It may be difficult for the DM to specify the marginal rates of substitution (consistency)
  - Theoretical soundness does not guarantee easiness of use

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:**

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

- **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

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

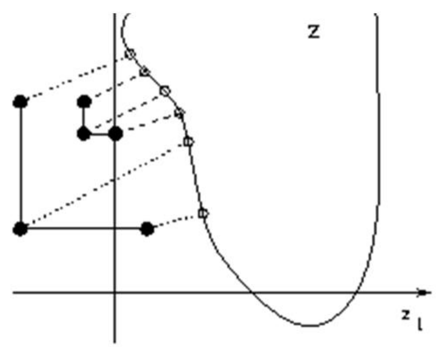


Figure 6. Altering the reference points.

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**

GUESS Method (Buchanan)

- **Idea: To make guesses $\check{z}^h$ and see what happens. (The search procedure is not assisted.)**
- **Assumptions: $z^{\star}$ and z^{nad} are available.**
- **Maximize the min. weighted deviation from z^{nad}.**
- **Each f_i(x) is normalized $\Rightarrow$ range is [0,1].**

$$\frac{z_i^{\text{nad}} - f_i(\mathbf{x})}{z_i^{\text{nad}} - z_i^{\star}}.$$

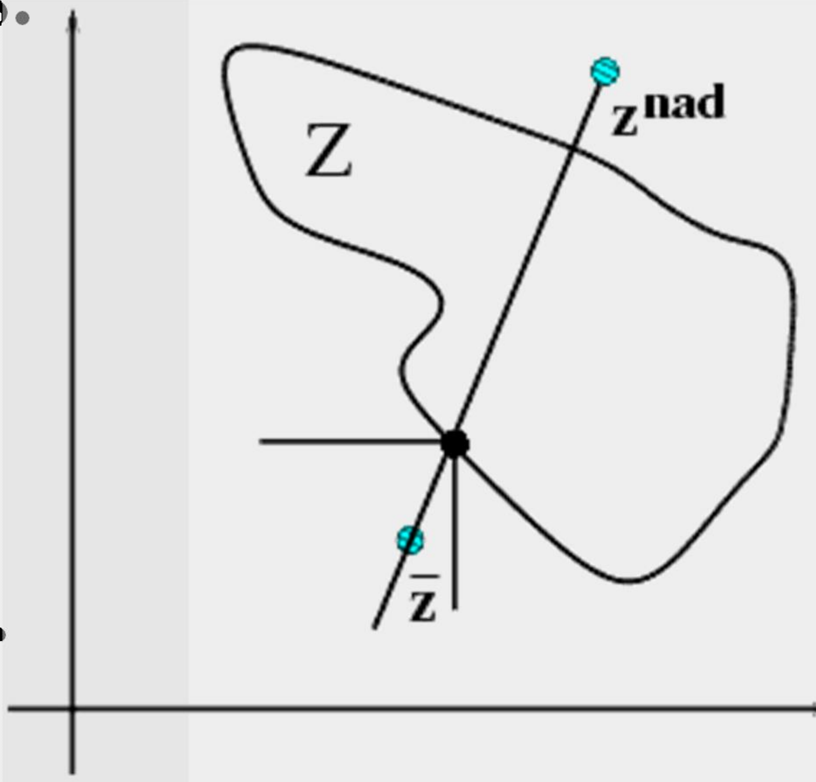
➔ <u>Problem</u>:

$$\begin{array}{ll} \text{maximize} & \min\limits_{1 \le i \le k} \left[\dfrac{z_i^{\text{nad}} - f_i(\mathbf{x})}{z_i^{\text{nad}} - \bar{z}_i^h} \right] \\ \text{subject to} & \mathbf{x} \in S. \end{array}$$

+ **Solution is weakly PO.**
+ **Any PO solution can be found.**

GUESS Algorithm

1) Present the ideal and the nadir objective vectors to the DM.
2) Let the DM give upper or lower bounds to the objective functions if (s)he so desires. Update the problem, if necessary.
3) Ask the DM to specify a reference point.
4) Solve the problem.
5) If the DM is satisfied, stop. Otherwise go to 2).



GUESS Method cont.

- Ad hoc method
- \+ Simple to use
- \+ No specific assumptions are set on the behaviour or the preference structure of the DM. No consistency is required
- \+ Good performance in comparative evaluations
- \+ Works for nondifferentiable problems
- – No guidance in setting new aspiration levels
- – Optional upper/lower bounds are not checked
- – Relies on the availability of the nadir point
- ! DMs are easily satisfied if there is a small difference between the reference point and the obtained 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**

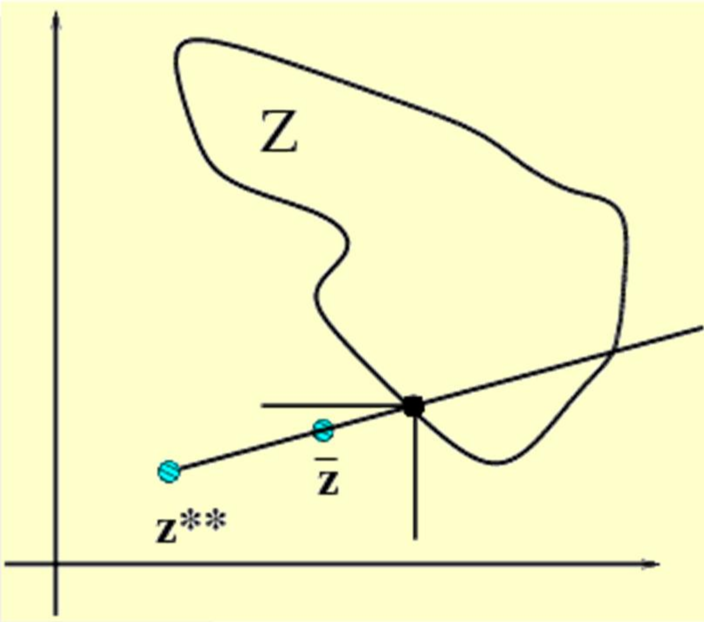
Satisficing Trade-Off Method cont.

➔ **Problem**

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

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 $\check{z}^{h} > z^{\blacklozenge\blacklozenge}$ 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).

Satisficing Trade-Off Method cont.

- **For linear and quadratic problems *exact trade-off* may be used to calculate how much objective values must be relaxed in order to stay in the PO set**
- **Ad hoc method**
- **Almost the same as the GUESS method if trade-off information is not available**

+ **The role of the DM is easy to understand: only reference points are used**
+ **Automatic or exact trade-off decrease burden on the DM**
+ **No consistency required**

– **The DM is not supported**

STEM

- Classification: $I^<$ and $I^\geq$ + ideal and nadir objective vectors

$$\begin{array}{ll} \text{minimize} & \max_{i=1,\ldots,k} \left[\frac{e_i}{\sum_{j=1}^k e_j} (f_i(\mathbf{x}) - z_i^\star) \right] \\ \text{subject to} & f_i(\mathbf{x}) \leq \varepsilon_i \quad \text{for all} \quad i \in I^\geq, \\ & f_i(\mathbf{x}) \leq f_i(\mathbf{x}^c) \quad \text{for all} \quad i \in I^<, \\ & \mathbf{x} \in S, \end{array}$$

- Solution is weakly Pareto optimal
- Benayon, Tergny, Larichev, Montgolfier

Light Beam Search (Slowinski, Jaszkiewicz)

- **Idea: To combine the reference point idea and tools of multiattribute decision analysis (ELECTRE)**
- **Minimize order-approximating achievement function (with an infeasible reference point).**

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

- **Assumptions**
  - **functions are continuously differentiable**
  - **$z^{\star}$ and z^{nad} are available**
  - **none of the objective functions is more important than all the others together**

Light Beam Search Algorithm

1) Get the best and the worst values of each f_i from the DM or calculate $z^{\star}$ and z^{nad}. Set $z^{\star}$ as reference point. Get indifference (preference and veto) thresholds.
2) Minimize the achievement function.
3) Calculate k PO additional alternatives and show them. If the DM wants to see alternatives between any two, set their difference as a search direction, take steps in that direction and project them. If desired, save the current solution.
4) The DM can revise the thresholds; then go to 3). If (s)he wants to change reference point, go to 2). If, (s)he wants to change the current solution, go to 3). If one of the alternatives is satisfactory, stop.

Light Beam Search cont.

- Establish *outranking relations* between alternatives. Alternative outranks another if it is at least as good as the latter
- DM gives *indifference thresholds*=intervals where indifference prevails. Hesitation between indifference and preference=*preference thresholds*. V*eto threshold* prevents compensating poor values in some objectives
- Alternatives near the current solution (based on the reference point) generated so that they outrank current one - no incomparable/indifferent solutions shown
- Ad hoc method

+ Versatile possibilities: specifying reference points, comparing alternatives and affecting the set of alternatives in different ways

– Specifying different thresholds may be demanding. They are important

+ The thresholds are not assumed to be global.

+ Thresholds should decrease the burden on the DM

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 – Which is the best? $\Rightarrow$ Synchronous NIMBUS®
- Different solutions are obtained using different scalarizations (Miettinen, Mäkelä, OR Spec (2002))
- Show them to the DM & let her/him choose the best
- **NB: DM is assumed to have knowledge about the problem in question, no deep understanding of optimization process or theory**
- Also intermediate solutions can be generated
- Versatile possibilities to direct solution process

Classification in NIMBUS

- **Form of interaction: Classification of objective functions into up to 5 classes**
- **Classification: 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 ($i \in I^\Diamond$)**
- **Functions in $I^\leq$ are to be minimized only till the specified level.**
- **DM must be willing to give up something**
- Miettinen, Mäkelä: Optim (1995), JORS (1999), Comp&OR (2000), EJOR (2006)

NIMBUS® Method cont.

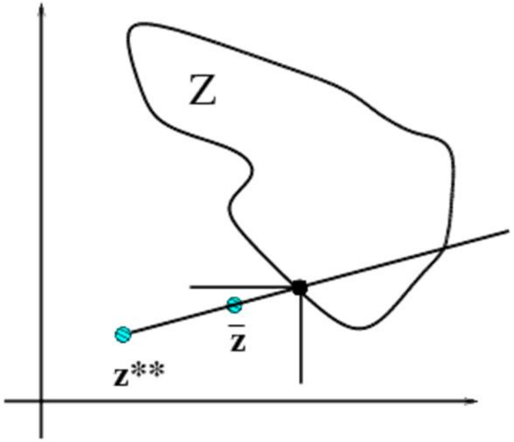
- Solve subproblem

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

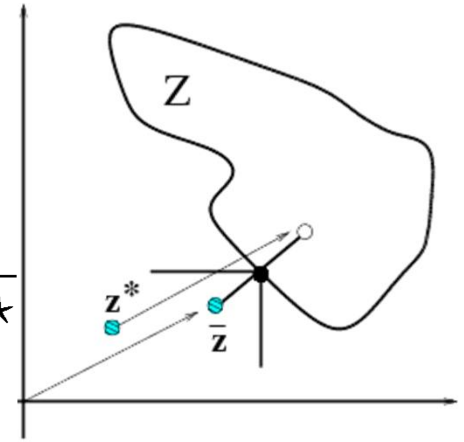
  where $\rho > 0$
- Appropriate single objective optimizer
- **Solution properly PO. Any PO solution can be found**
- Solution satisfies desires as well as possible – feedback of tradeoffs
- We have 3 more subproblems to get more solutions

3 More Subproblems

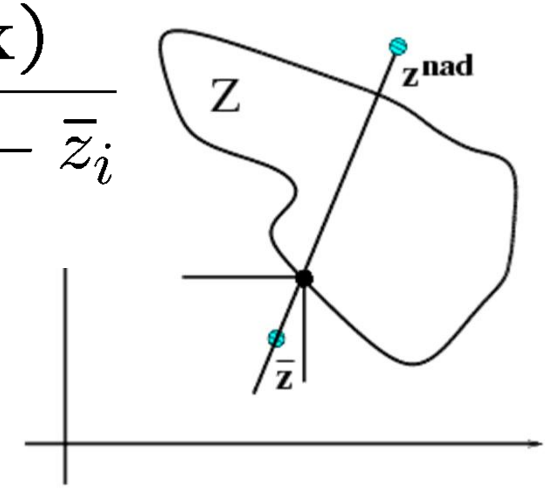
$$\begin{aligned} &\min \quad \max_{i=1,\dots,k} \left[\frac{f_i(\mathbf{x}) - z_i^{\star\star}}{\bar{z}_i - z_i^{\star\star}} \right] + \rho \sum_{i=1}^{k} \frac{f_i(\mathbf{x})}{\bar{z}_i - z_i^{\star\star}} \\ &\text{s.t.} \quad \mathbf{x} \in S \end{aligned}$$



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



$$\begin{aligned} &\min \quad \max_{i \notin I^{\diamond}} \left[\frac{f_i(\mathbf{x}) - z_i^{\mathsf{nad}}}{z_i^{\mathsf{nad}} - \bar{z}_i} \right] + \rho \sum_{i=1}^{k} \frac{f_i(\mathbf{x})}{z_i^{\mathsf{nad}} - \bar{z}_i} \\ &\text{s.t.} \quad \mathbf{x} \in S \end{aligned}$$



$\Rightarrow$ Different Pareto optimal solutions

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 $t_j=j/(P+1)$**
- **Only different solutions are shown**
- **Search iteratively around the PO set – learning-oriented**
- **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 demanding questions are posed to the DM**

\+ **Classification and comparison of alternatives are used in the extent the DM desires**

\+ **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.

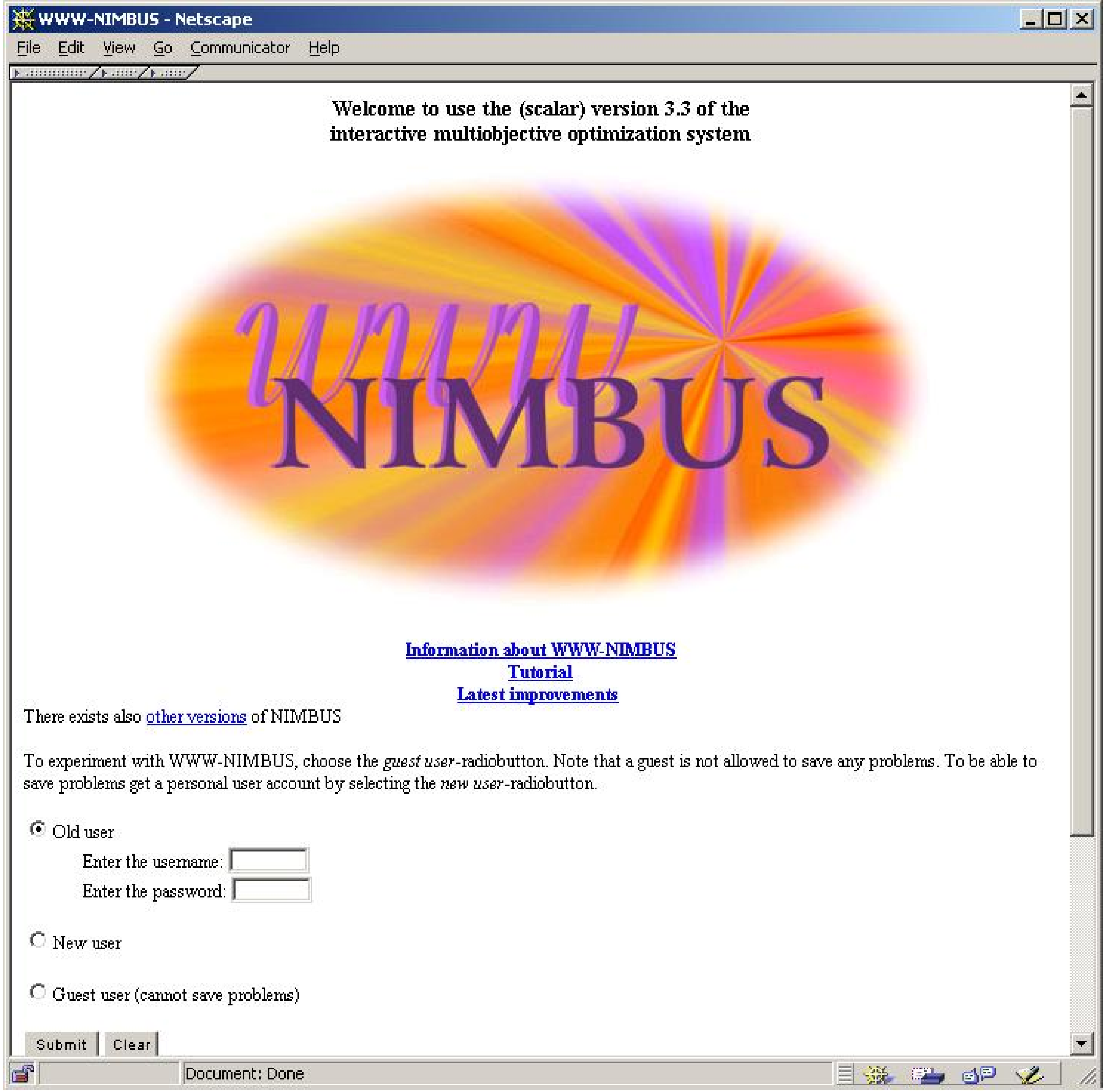
WWW-NIMBUS® since 1995

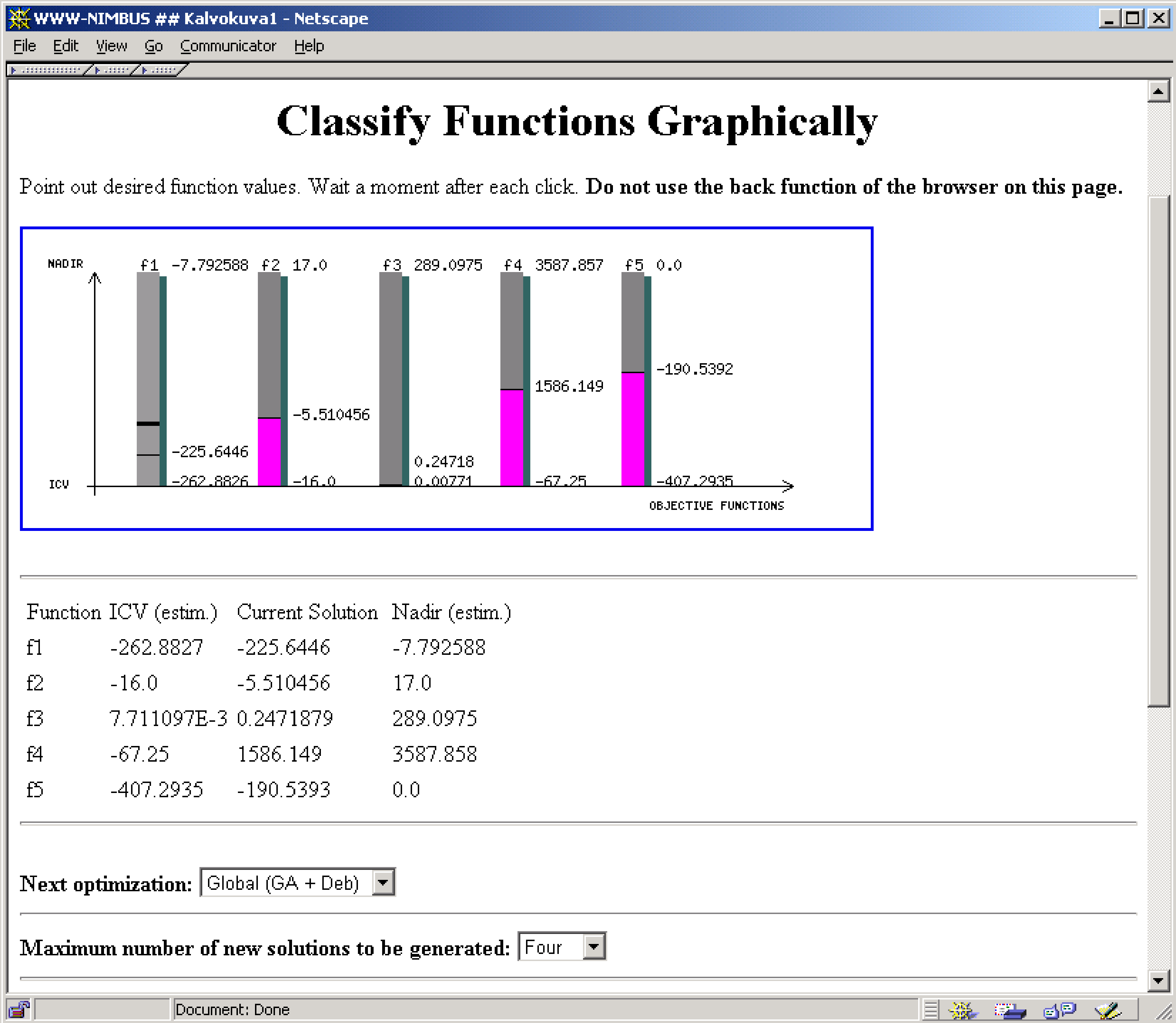
- The first, unique interactive optimization system on the Internet
- Centralized computing (server in Jyväskylä) & distributed interface
- No special requirements for computers: No computing capacity nor compilers needed
- Latest version always available
- Graphical user-interface via WWW
- Personal username and password
- Even for nonconvex and nondifferentiable problems and integer-valued variables
- Symbolic (sub)differentiation
- Available to any academic Internet user for free
- Tutorial and online help
- ➢ Miettinen, Mäkelä, Comp & OR (2000), EJOR (2006)

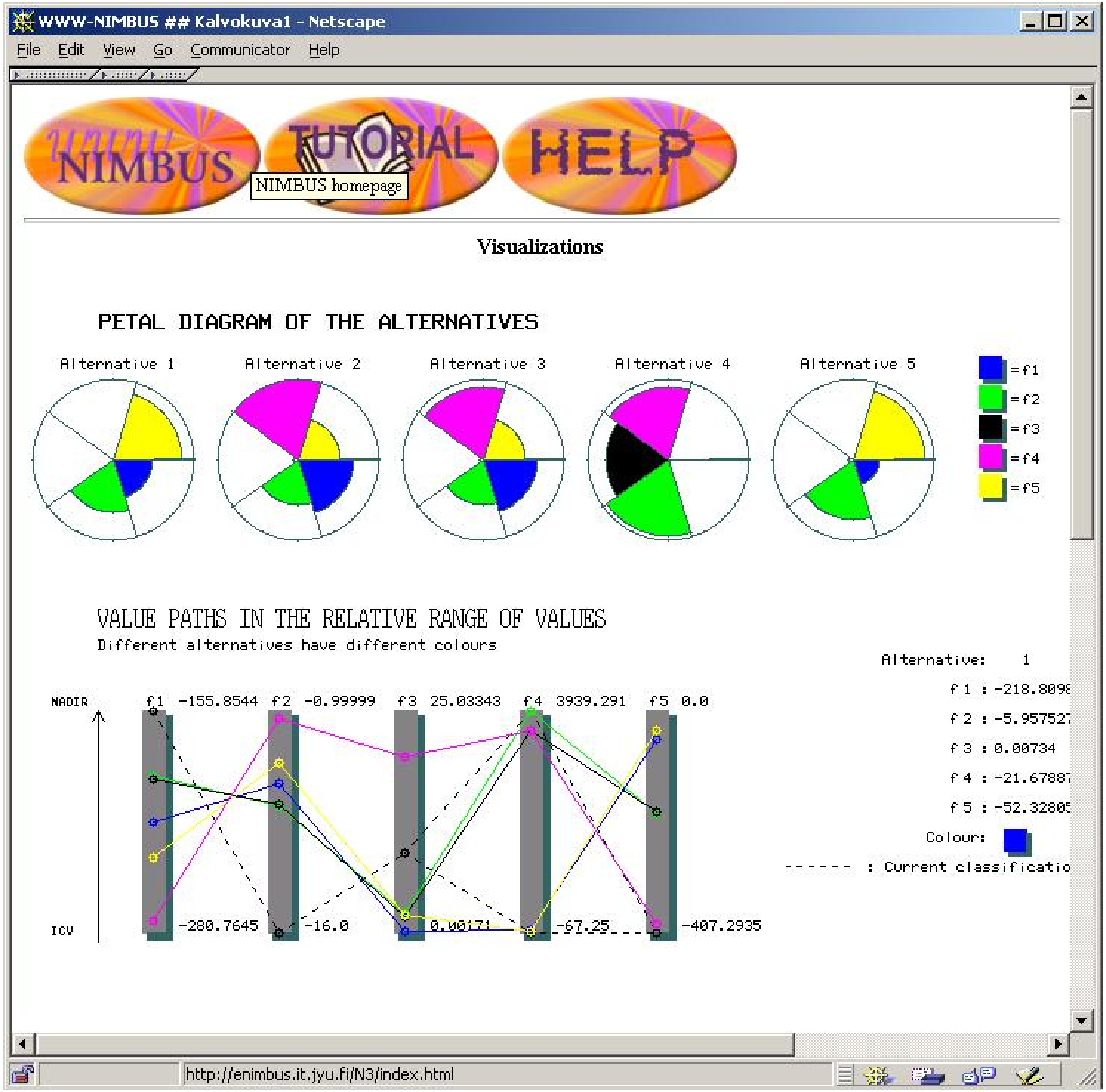
http://nimbus.it.jyu.fi/

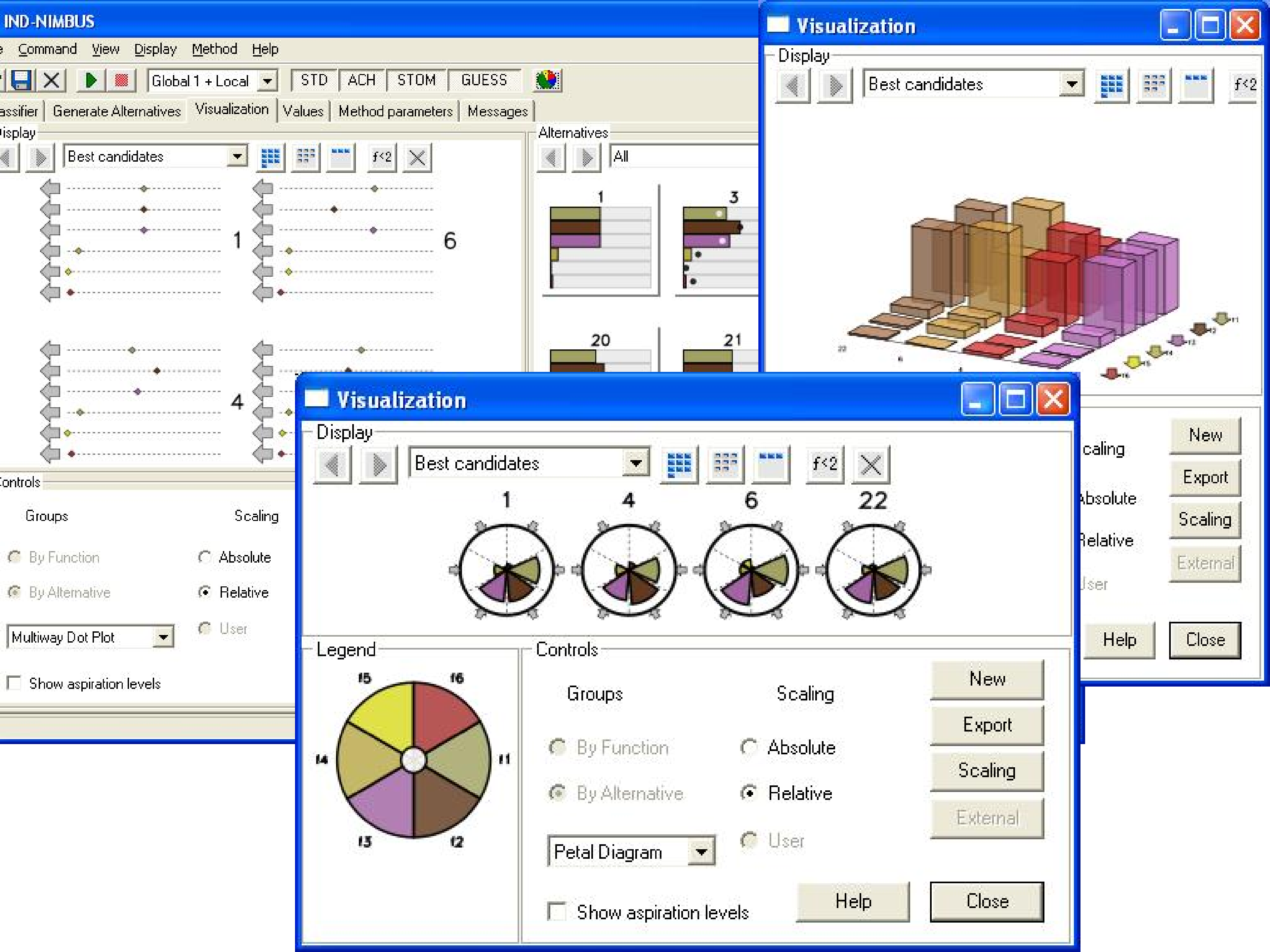
WWW-NIMBUS Version 4

- Synchronous algorithm
  - Several scalarizing functions based on the same user input
- Minimize/maximize objective functions
- Linear/nonlinear inequality/equality and/or box constraints
- Continuous or integer-valued variables
- Local and and global single-objective solvers and hybrids
- Different constraint-handling methods
- Problem formulation and results available in a file
- Possible to
  - change solver at every iteration or change parameters
  - edit/modify the current problem
  - save different solutions and return to them (visualize, intermediate) using database


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.
NADIR
ICV
f1 -7.792588
-225.6446
-262.8826
f2 17.0
-5.510456
-16.0
f3 289.0975
0.24718
0.00771
f4 3587.857
1586.149
-67.25
f5 0.0
-190.5392
-407.2935
OBJECTIVE FUNCTIONS
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
NADIR
ICV
f1 -155.8544
f2 -0.99999
f3 25.03343
f4 3939.291
f5 0.0
-280.7645
-16.0
0.00171
-67.25
-407.2935
Alternative: 1
f 1 : -218.8098
f 2 : -5.957527
f 3 : 0.00734
f 4 : -21.67887
f 5 : -52.32805
Colour:
------ : Current classificatio
http://enimbus.it.jyu.fi/N3/index.html


IND-NIMBUS
Command View Display Method Help
Global 1 + Local
STD ACH STOM GUESS
Generate Alternatives Visualization Values Method parameters Messages
Alternatives
All
Best candidates
Controls
Groups
Scaling
By Function
By Alternative
Absolute
Relative
User
Multiway Dot Plot
Show aspiration levels
Visualization
Display
Legend
Petal Diagram
New
Export
Scaling
External
Help
Close

IND-NIMBUS®

- For MS-Windows and Linux operating systems
- Synchronous NIMBUS, Pareto Navigator, PAINT
- Minimize/maximize objective functions
- Linear/nonlinear inequality/equality and/or box constraints
- Continuous or integer-valued variables
- Local solvers and global solvers and their **hybrid**
- User can change solver at every iteration
- User can change parameters of solvers
- Has been connected with e.g.
  - BALAS® and APROS® process simulators by (VTT) and GPS-X simulator
  - different modelling and simulation tools (Matlab, GAMS)

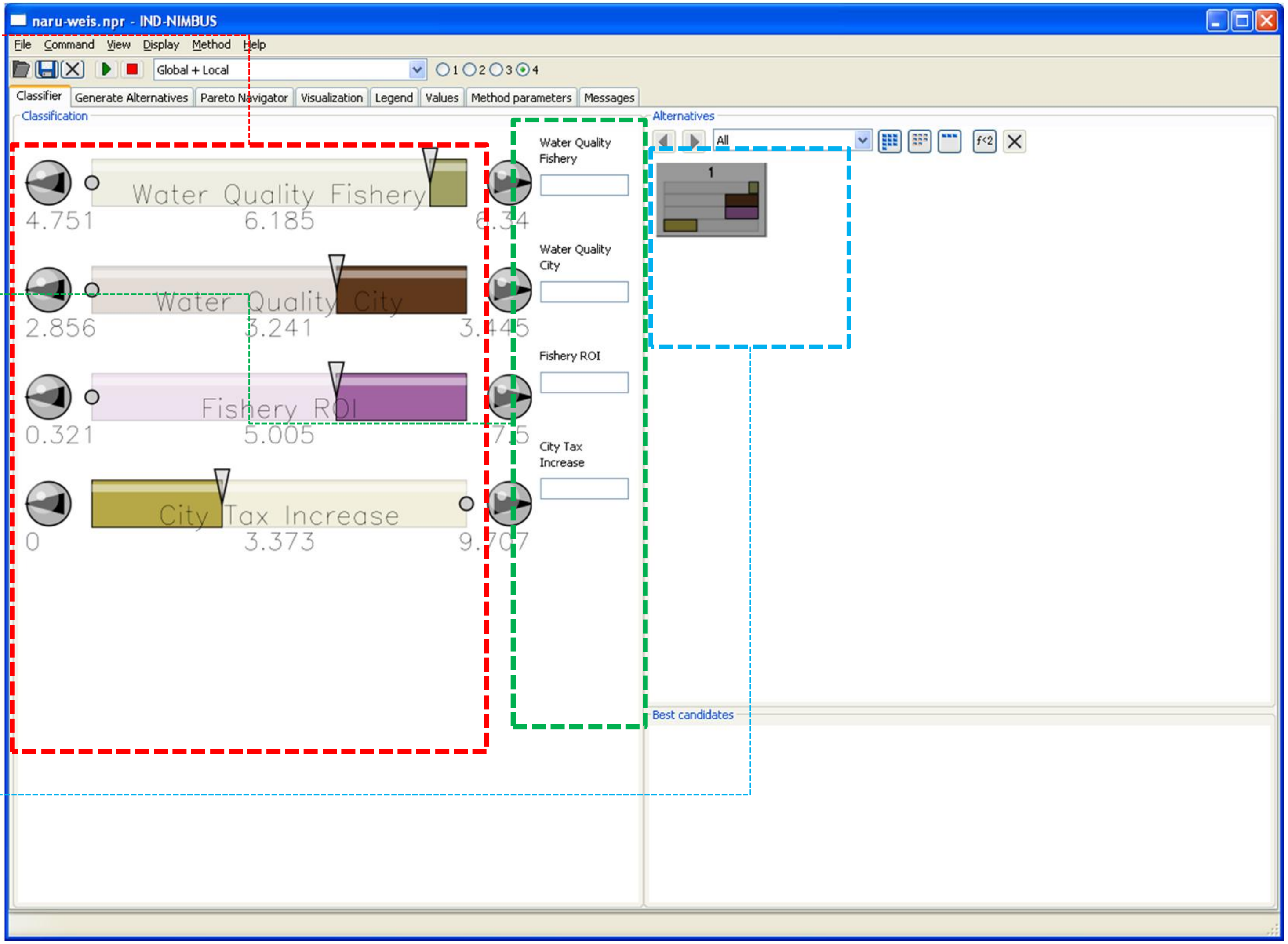
http://ind-nimbus.it.jyu.fi/

IND-NIMBUS® Views

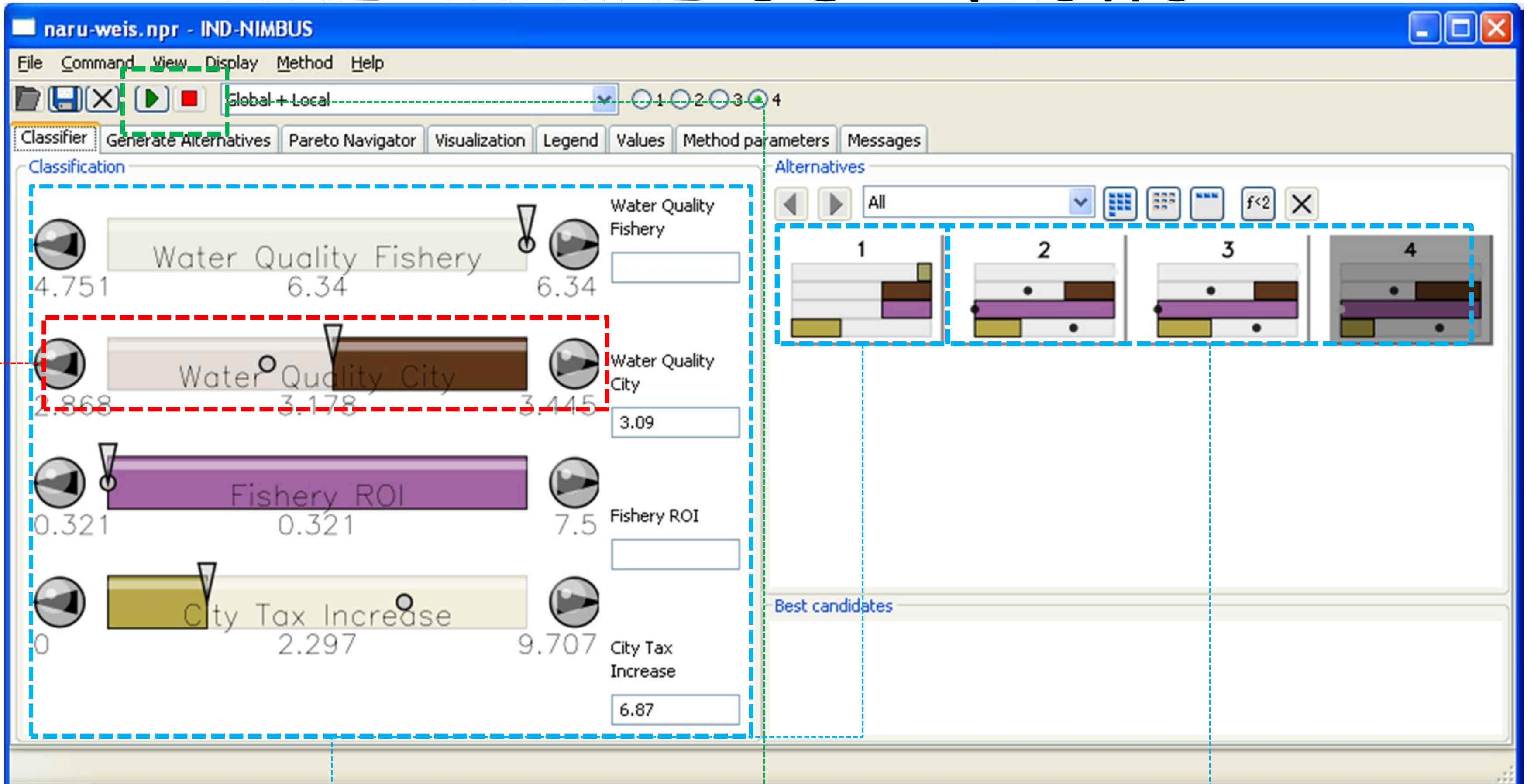
Objective function values and classification

Numerical classification boundaries

Initial PO solution



IND-NIMBUS® Views



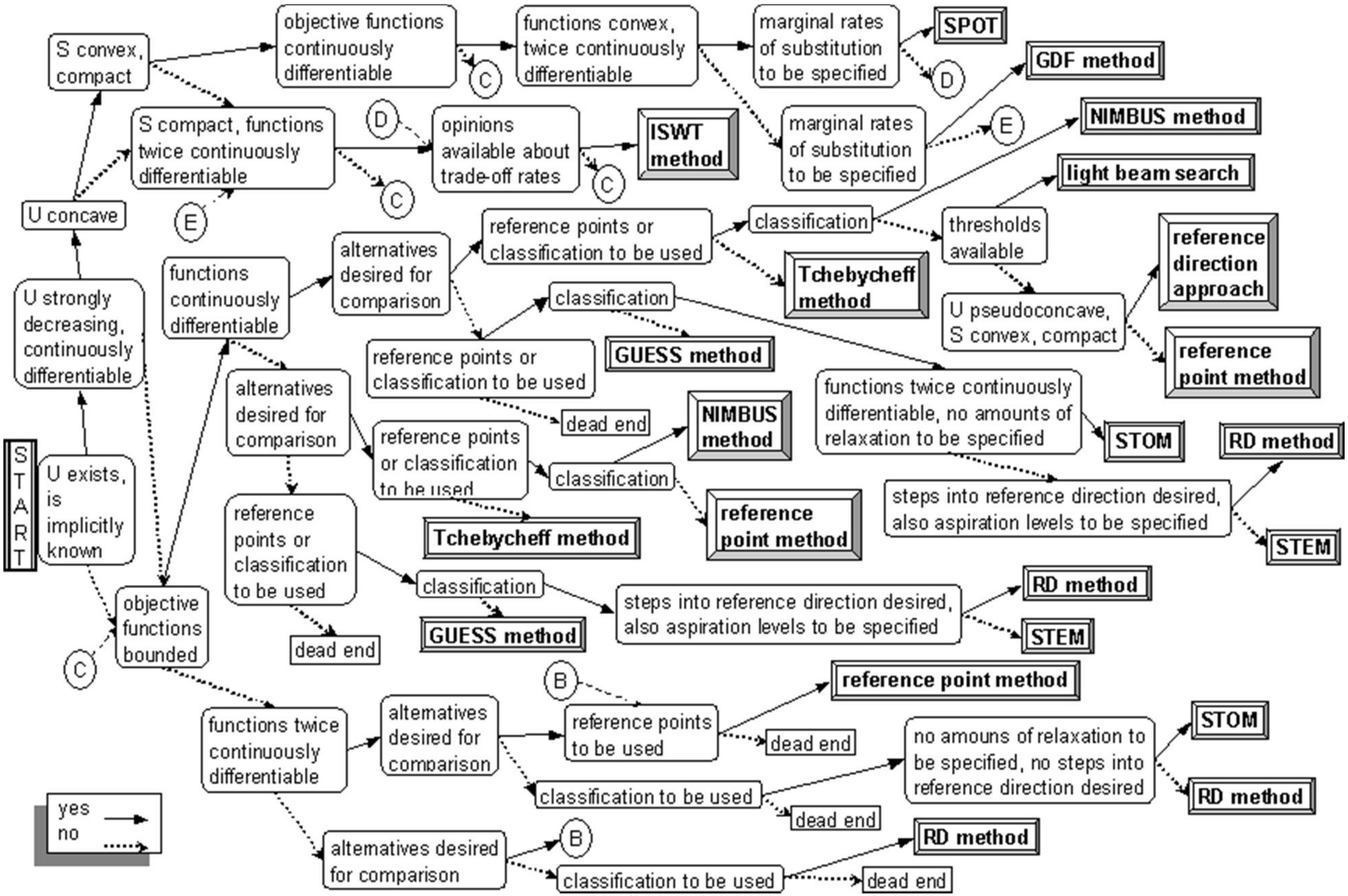
NIMBUS classification by clicking objective bar

Initial PO solution

New solutions calculated with a play button

New PO solutions

Tree Diagram of Methods



Computational Challenges

of complex simulation-based optimization

- We need tools for handling complexity
- Computational cost
  - Objective and constraint functions depend on output of simulation models – may be time-consuming
- Multiple conflicting objectives
  - Identifying most preferred solution requires preferences of decision maker – methodological support needed
- Black-box models
  - Global optimization needed -> computational cost
- Stochasticity
  - Model output is random vector with unknown distribution
  - Sampling the output increases computational cost

Pareto Navigator

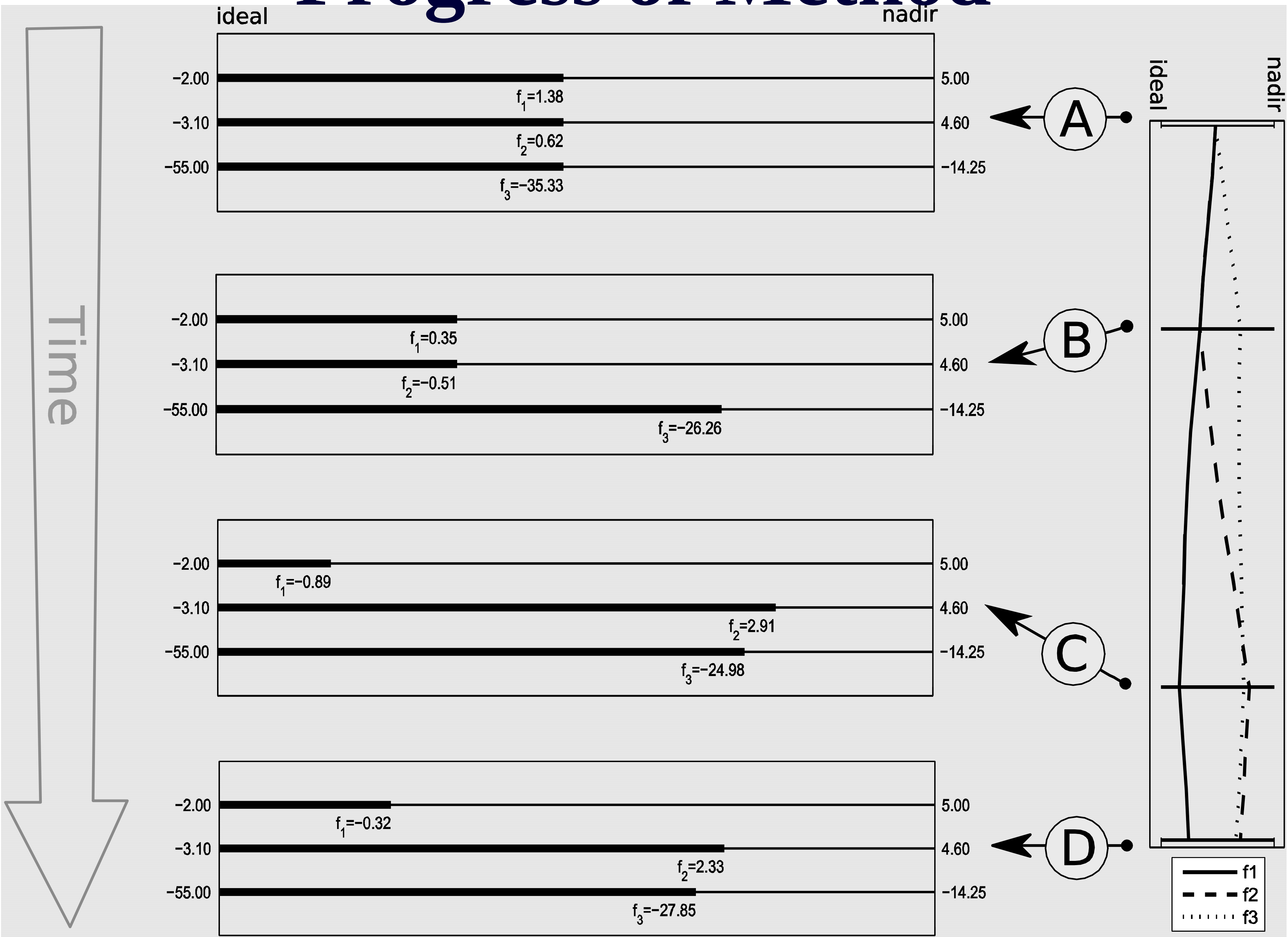
Eskelinen et al., OR Spectrum (2010)

- **Challenge: computational expense (convex)**
- Background & motivation
  - How to support DM?
  - I Learning phase   II Decision phase
  - Computationally costly problems
- Pareto optimal set = **actual PO set**
- Learning-oriented interactive method
- Instead of approximating objective functions we directly approximate PO set
- Hybrid method which combines a posteriori and interactive methods
- Move around approximation – then project to actual PO set

Pareto Navigator, cont.

- Initialization phase
  - (relatively small) set of Pareto optimal solutions
  - polyhedral approximation of Pareto optimal set (convex hull of PO solutions available) in objective space – ***approximated PO set***
- Navigation phase
  - DM specifies how current solution should be improved, e.g. reference point ž. Set search direction $d = ž – z^c$
  - dynamic real-time movement into desired direction
  - information of whole PO set – possibilities and limitations
  - active participation of DM: DM can **learn** about problem, trade-offs, interdependencies and adjust one's hopes
  - DM can concentrate on interesting solutions
  - computationally inexpensive (parametric LP)
  - more accurate approximation can be generated of part of PO set (Klamroth, Miettinen, Oper Res)

Progress of Method

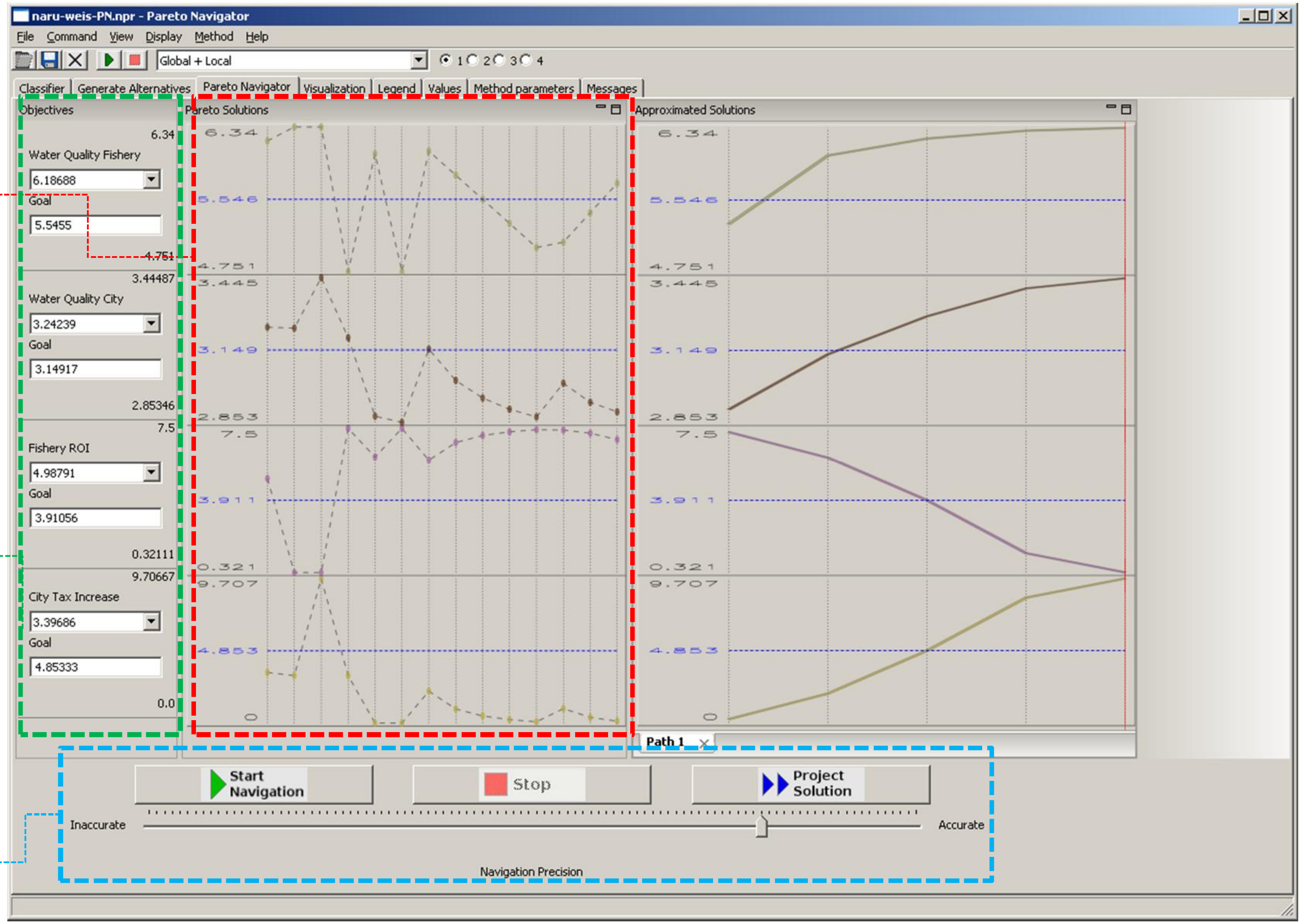


Pareto Navigator Views

Pareto optimal solutions

Objective function values and goals

Pareto Navigator controls

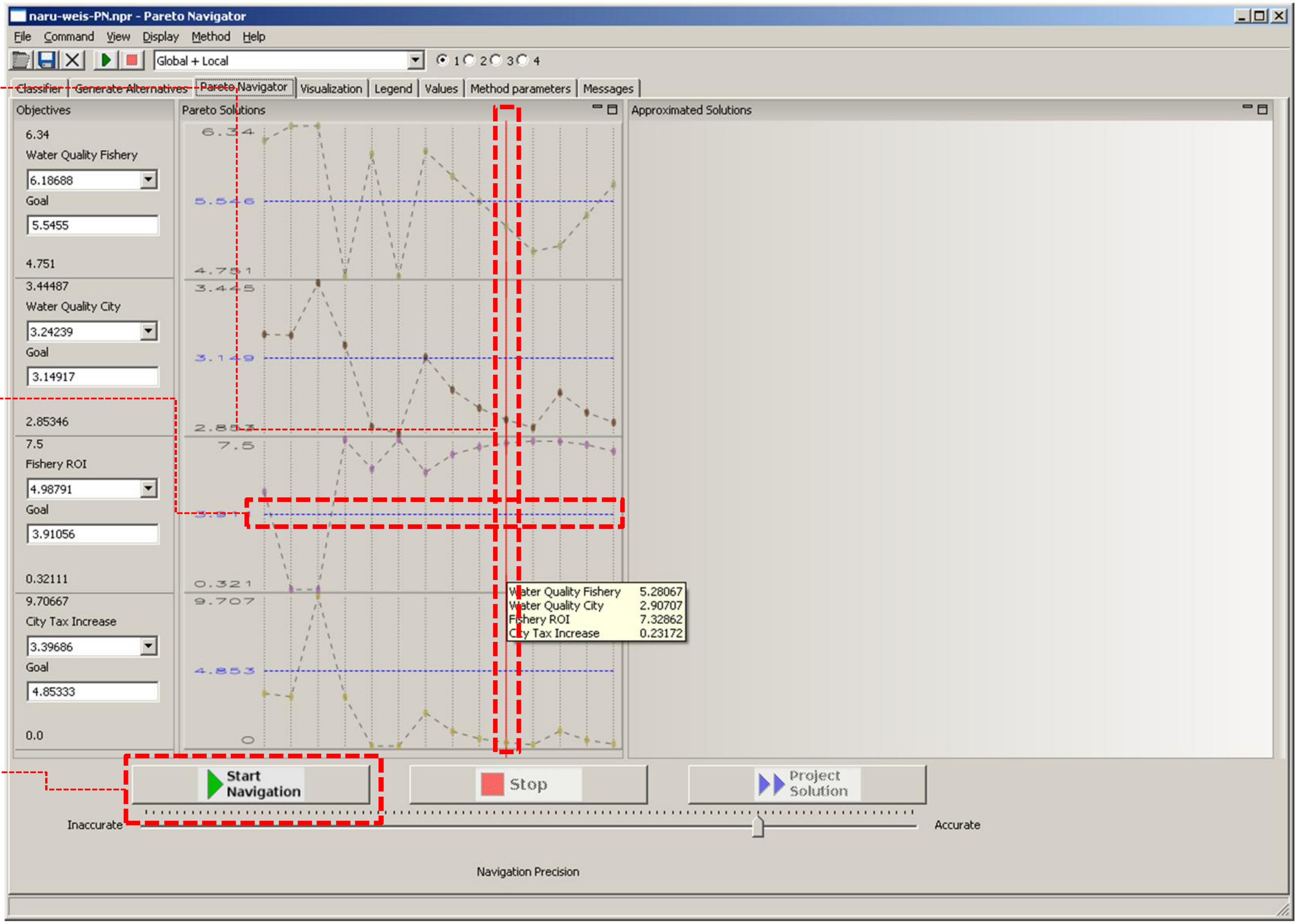


Pareto Navigator Views

First, the DM selects PO solution where navigation starts

Next, the DM sets goals for objectives

Then, the DM starts to navigate along the direction of goals



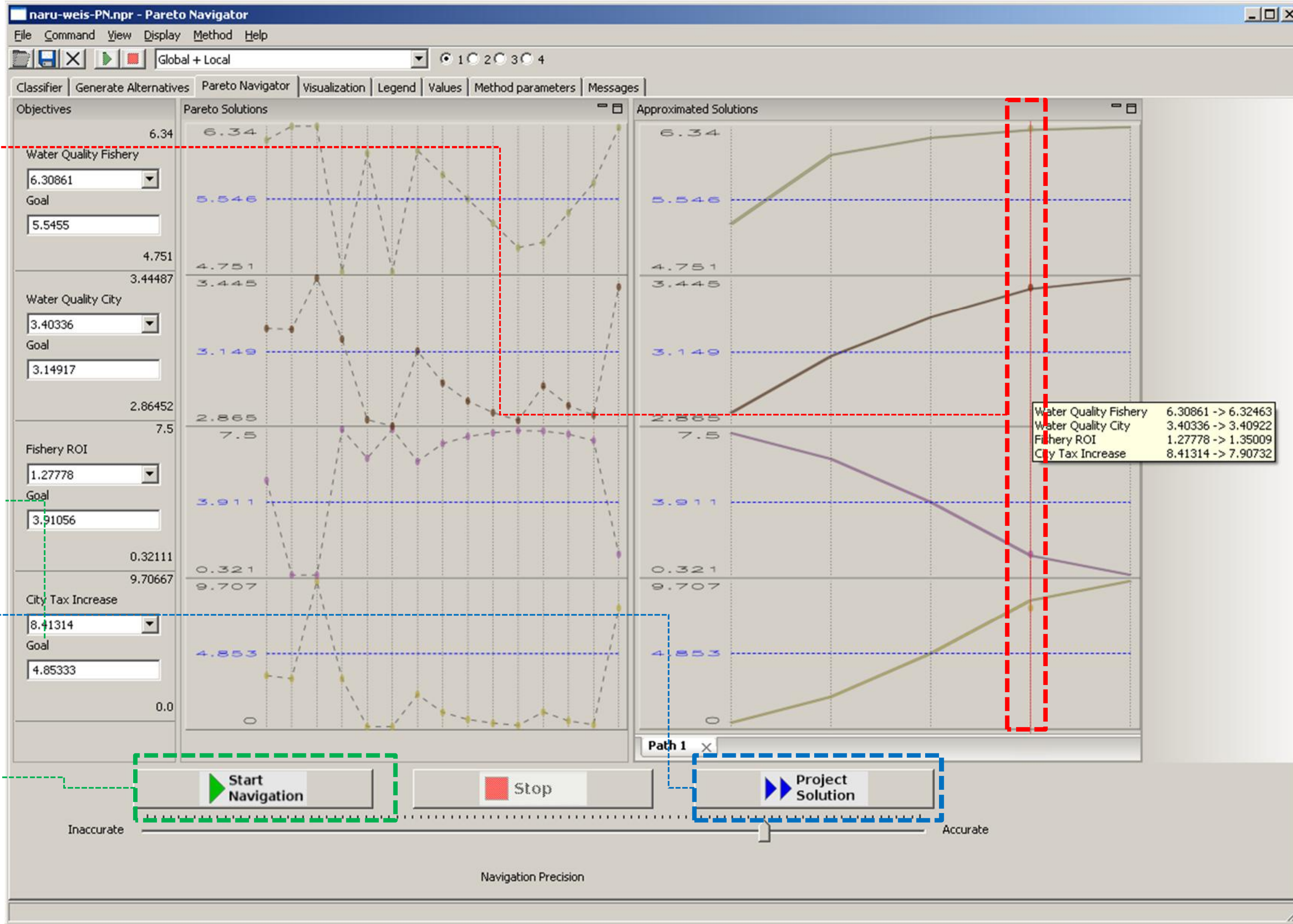
Pareto Navigator Views

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

Approximated solutions can be used

to project them to real PO solutions

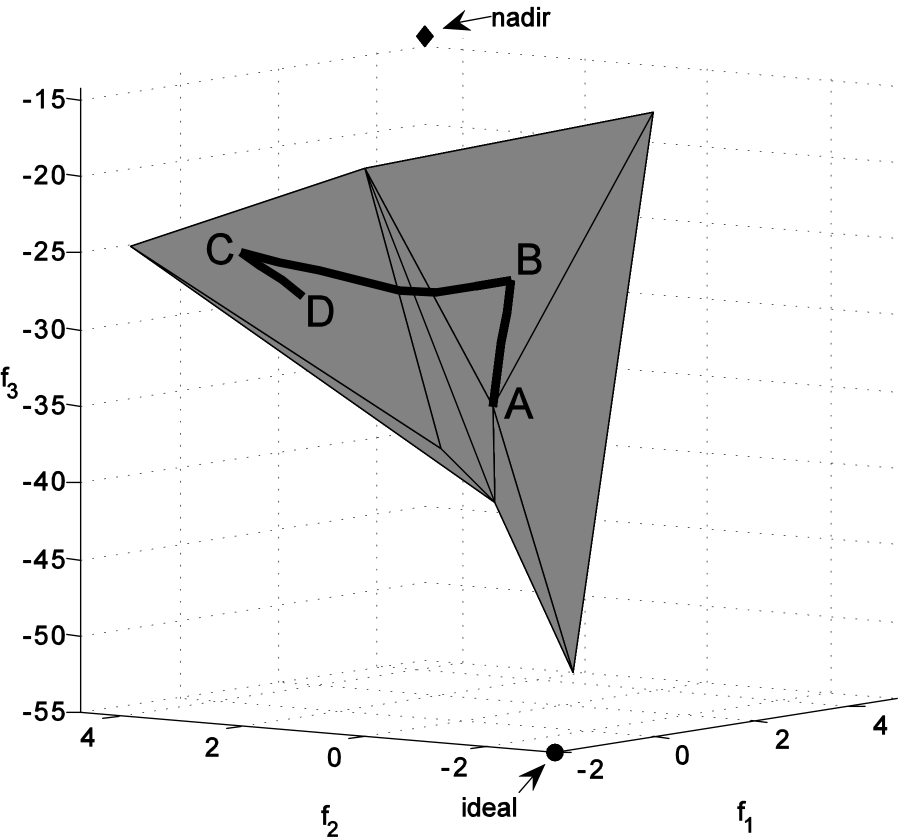
or as a starting point for new navigation

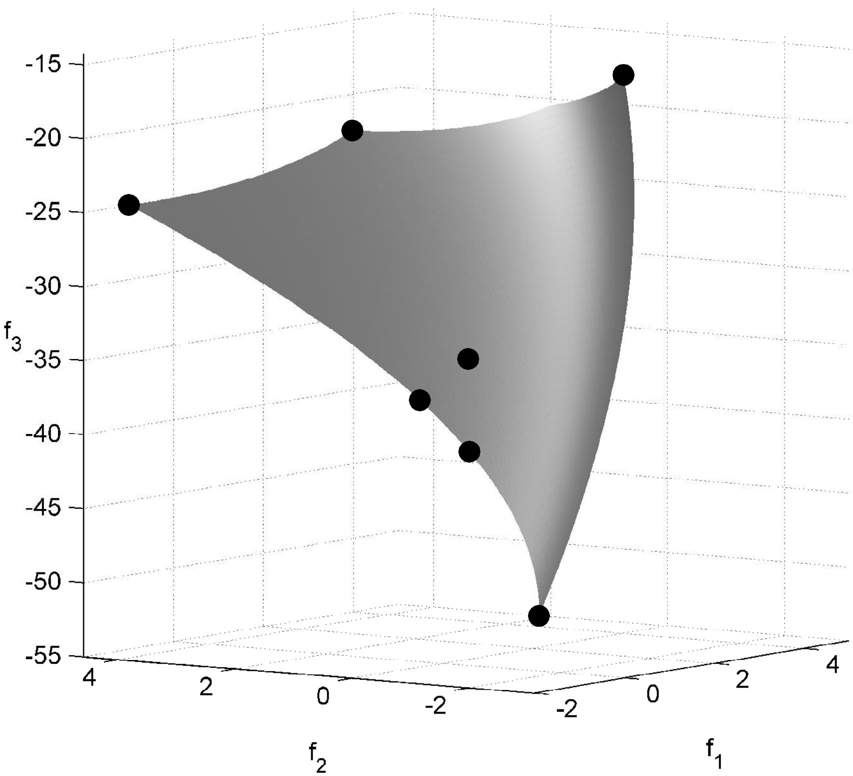


Example in 3D

- We can see what happenes in objective space during the solution process

  (polyhedral approximation and actual PO set)





Comments on Pareto Navigator

- If approximated solution desirable, it is projected to actual PO set
- Stop if DM is satisfied with projected solution
  - DM can continue navigation
  - solution can be included in the approximation and approximation regenerated
  - more accurate approximation can be generated of a part of PO set (Klamroth, Miettinen, Oper Res, 2008)
  - DM can continue with some interactive method to fine-tune
- Enables convenient and real-time navigation in the approximated PO set
  - DM can move in directions of promising solutions and learn
- For computationally demanding problems
- Instead of approximating objective functions we directly approximate PO set

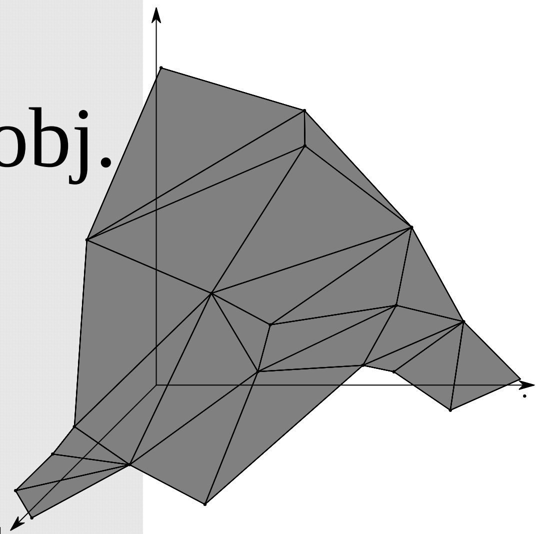
PAINT

Hartikainen et al., Comp Opt & Appl (2012)

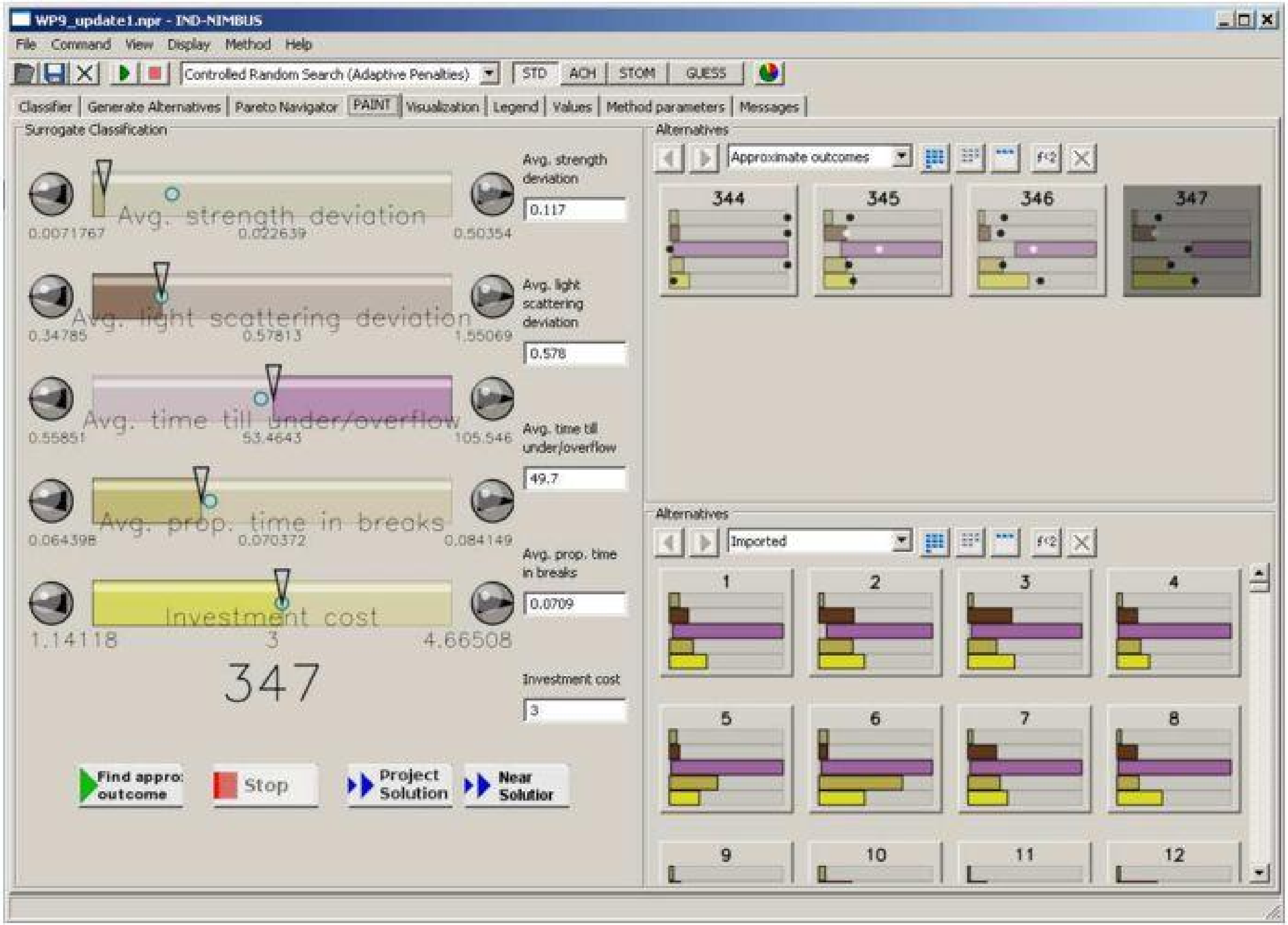
- **Challenge: computational expense (also nonconvex problems)**
- Background & motivation
  - Combine a posteriori and interactive methods
    - Support learning
  - For computationally expensive problems
    - Avoid long waiting times in interactive methods when generating new PO solutions
- Idea
  - Compute a small set of Pareto optimal solutions
  - Create approximated multiobj.optimization problem
  - Use any interactive method to find the best approximate solution on the approximation
  - Find closest Pareto optimal solution in original problem

PAINT = PAreto front INTerpolation

- Both for convex and nonconvex problems
- Approximates PO set with Delaunay triangulations
- Forms mixed-integer linear multiobj. opt. problem as a surrogate to the original one
  - computationally inexpensive
- E.g. NIMBUS can be applied with no waiting times (e.g. CPLEX)



Screen Shot: PAINT+NIMBUS



Nautilus – Background

Miettinen et al., EJOR (2010)

- 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
- We do not react symmetrically to gains and losses
  - Requirement of trading off may hinder DM's willingness to move from the current Pareto optimal solution
- Typically low number of iterations in interactive methods
  - 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
  - ➢ Those negotiators easily anchor at starting Pareto optimal solution if it is advantageous for their interests

Prospect Theory

- Kahneman and Tversky (1979)
- Critique of expected utility theory as a descriptive model of decision making under risk
- Our attitudes to losses loom larger than gains
  - Pleasure of gaining some money seems to be lower than the dissatisfaction of losing the same amount of money
- People underweight outcomes that are merely probable in comparison with outcomes that are obtained with certainty

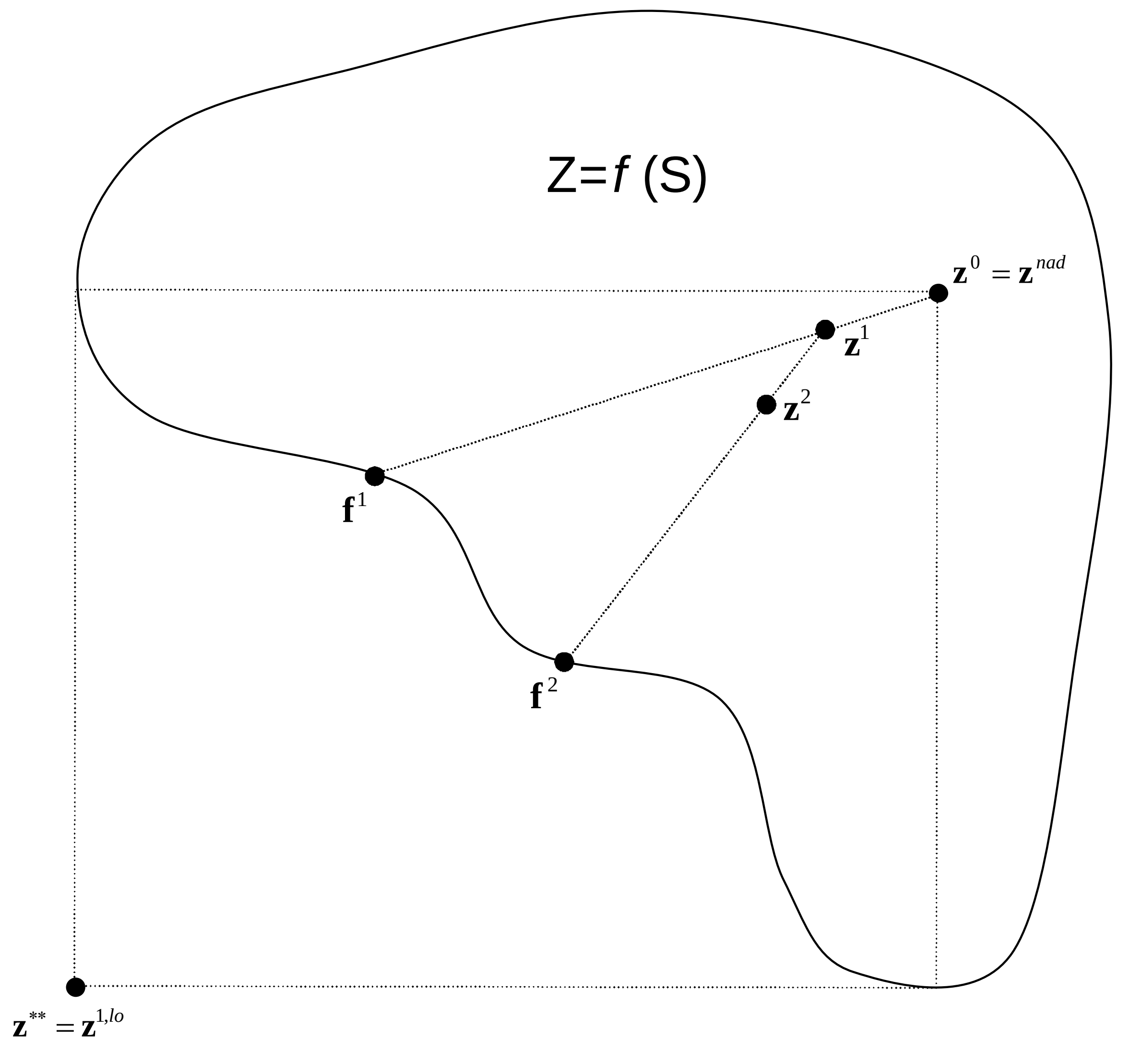
  -> risk aversion in choices involving sure gains

  -> risk seeking in choices involving sure losses

  -> inconsistent preferences for situations presented in different forms
- The past and present context of experience defines an adaptation level, or reference point, and stimuli are perceived in relation to this reference point
  - If we first see a very unsatisfactory solution, a somewhat better solution is more satisfactory than otherwise
- Location of reference point and the manner in which choice problems are coded and edited emerge as critical factors in the analysis of decisions

Nautilus

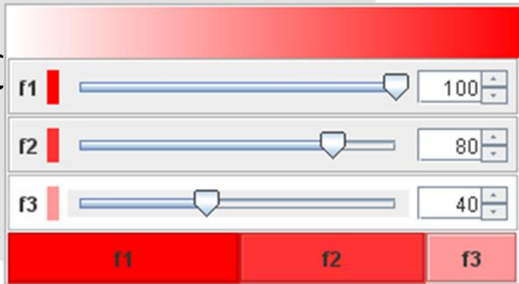
- Learning-oriented interactive method
- DM starts from the worst i.e. nadir objective vector and moves towards PO set
  - Improvement in each objective at each iteration
  - Possible to gain at every iteration – no need for impairment
- At each iteration, objective vector obtained dominates the previous one
- Only the final solution is Pareto optimal
- DM can always go backwards if desired
- The method allows the DM to approach the part of the PO set (s)he wishes



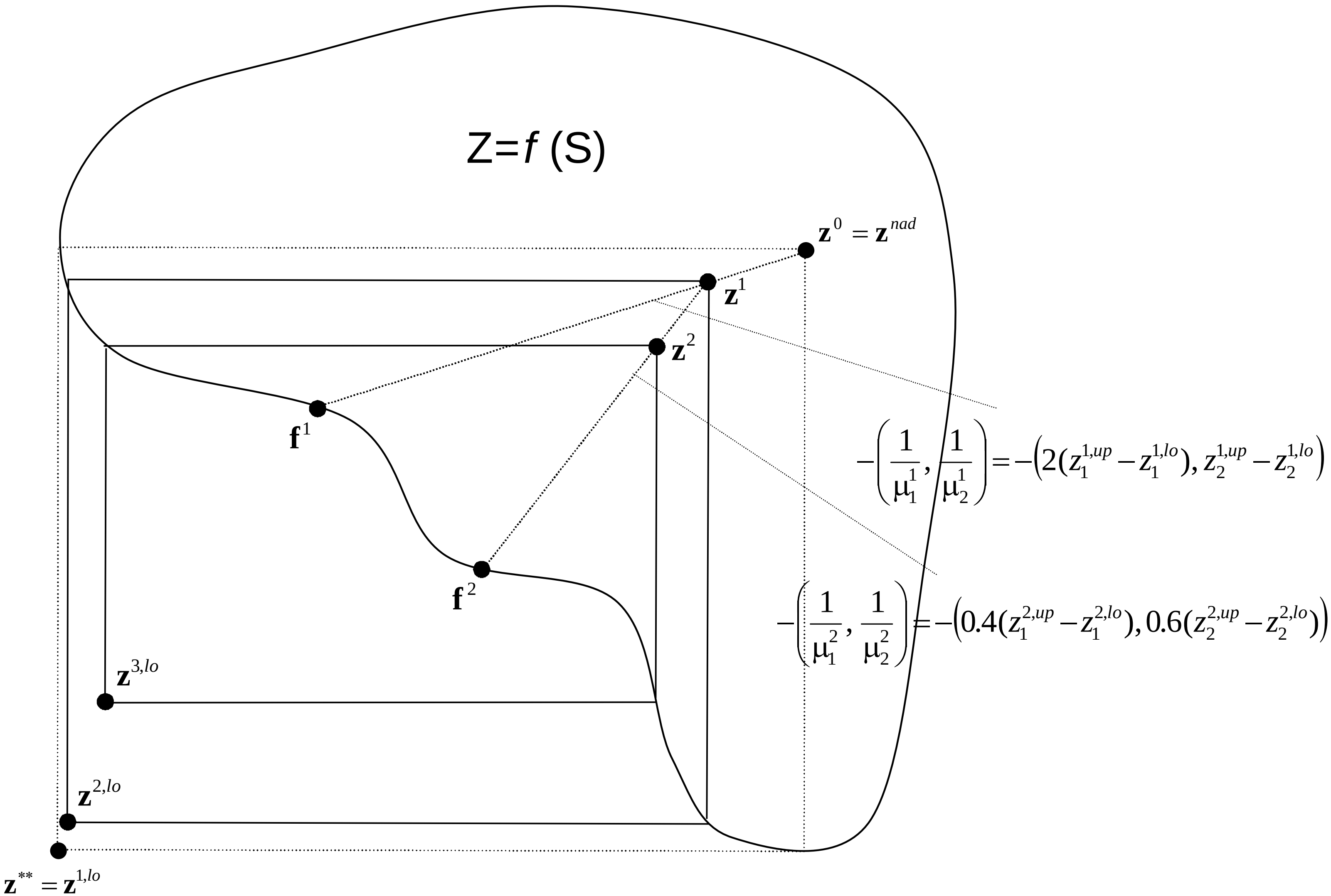
Z=f (S)
$\mathbf{z}^0 = \mathbf{z}^{nad}$
$\mathbf{z}^1$
$\mathbf{z}^2$
$\mathbf{f}^1$
$\mathbf{f}^2$
$\mathbf{z}^{**} = \mathbf{z}^{1,lo}$

Nautilus cont.

- Main underlying tool: achievement function based on a reference point
- Given the current values $\mathbf{z}^h$, there are two possibilities for preference information:
  - Rank relative importance of improving each current value (the higher rank, the more important improvement is)
  - How would you distribute 100 points among the current obj.values: the more points you allocate, the more improvement is desired
- DM sets number of steps to be taken (can be changed) and preferences related to nadir obj. vector
- We calculate number of iterations left
- We calculate the solution of achievement minimization, and take a step towards it
- At the last step we get PO solution



- At each iteration, range of reachable obj.values shrinks
- We calculate also how close current obj.vector is to the PO set



Example

	min	low	current	max
f1	-6.34	-6.34	-4.07	-4.07
f2	-3.44	-3.44	-2.83	-2.83
f3	-7.50	-7.50	-0.32	-0.32
f4	0.00	0.00	9.71	9.71

	min	low	current	max
f1	-6.34	-6.30	-4.30	-4.07
f2	-3.44	-3.40	-2.95	-2.83
f3	-7.50	-7.50	-2.71	-0.32
f4	0.00	0.33	7.47	9.71

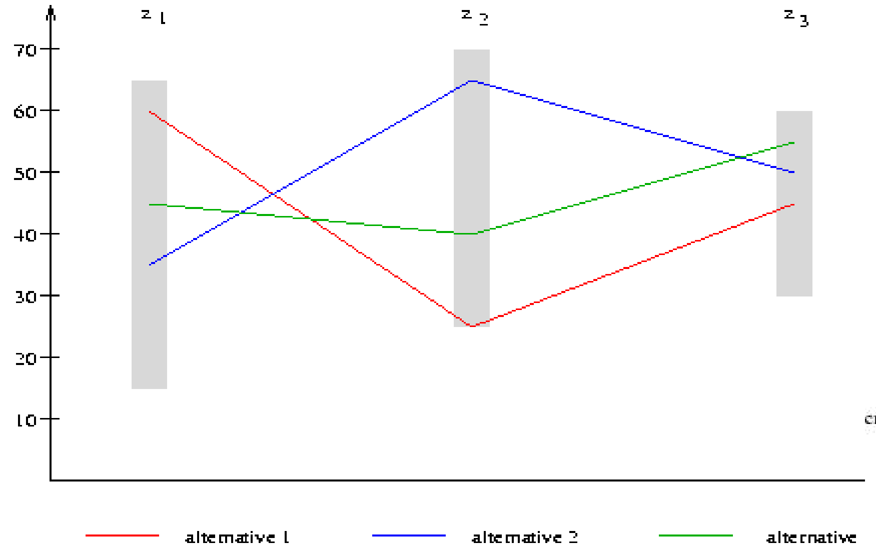
	min	low	current	max
f1	-6.34	-6.18	-4.52	-4.07
f2	-3.44	-3.33	-3.07	-2.83
f3	-7.50	-7.50	-5.11	-0.32
f4	0.00	1.23	5.24	9.71

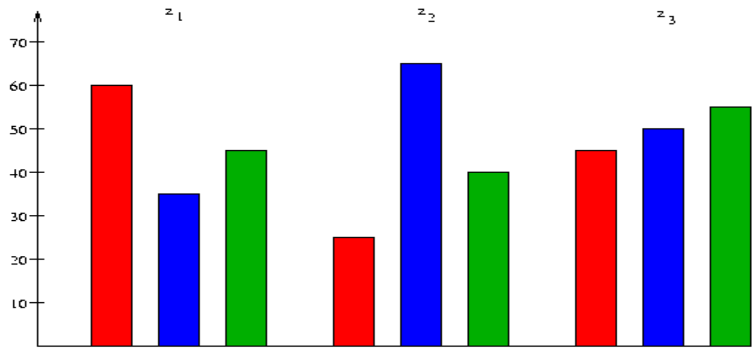
	min	low	current	max
f1	-6.34	-6.01	-6.01	-4.07
f2	-3.44	-3.21	-3.21	-2.83
f3	-7.50	-6.22	-6.22	-0.32
f4	0.00	2.99	2.99	9.71

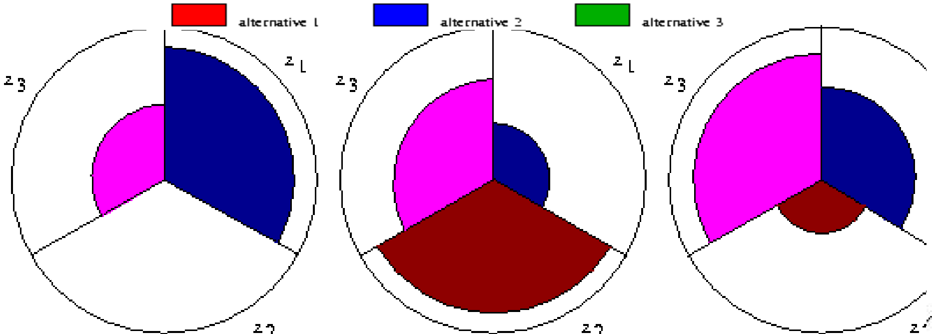
Nautilus - Remarks

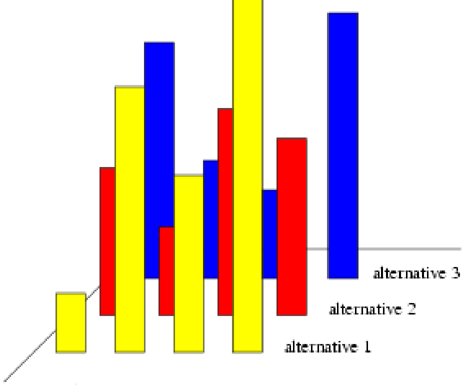
- During the solution process, connection to decision variable space is temporarily lost
  - Iteration points generated are only defined in objective space
  - We *know* that a feasible solution and corresponding obj.vector **better than** the current vector exist
- Nautilus allows free search
  - Nautilus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

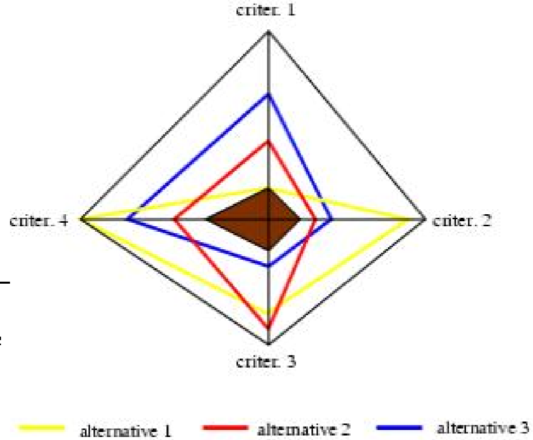
Graphical Illustrations

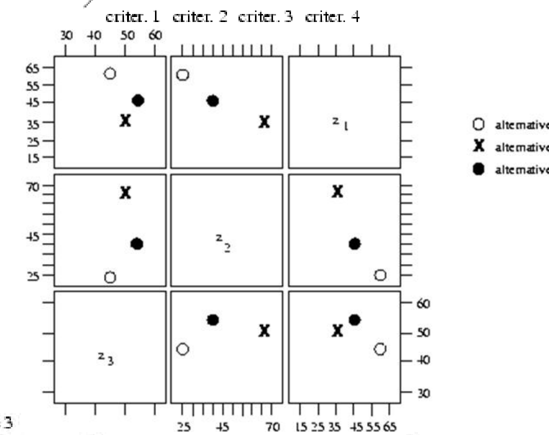


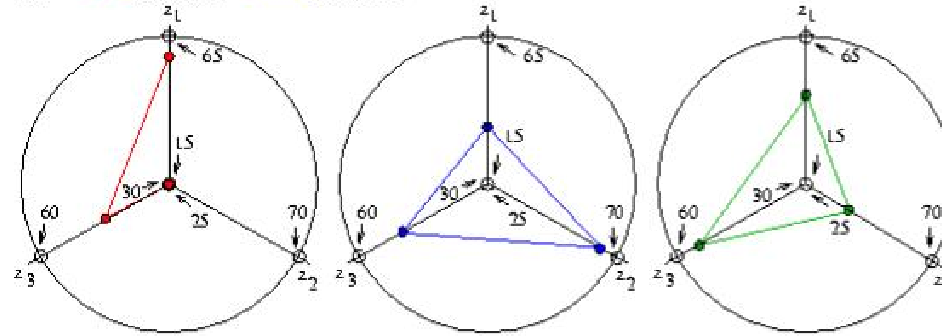


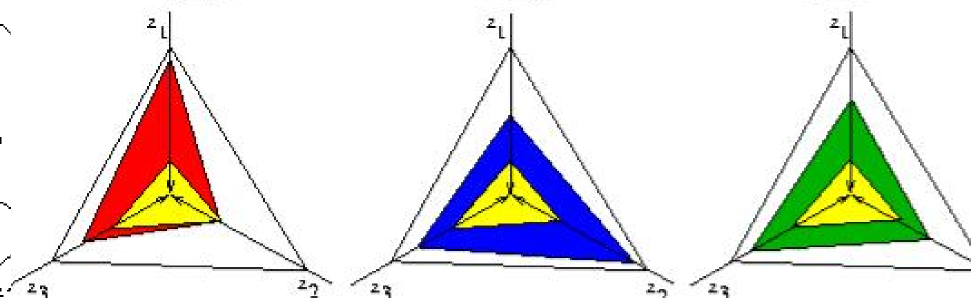












On Visual Illustration

- The decision maker (DM) is often asked to compare several alternatives
  - part of interactive methods (GDF, ISWT, Tchebycheff, reference point method, light beam search, NIMBUS)
- 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

On Applications

Gain important feedback for method development

Gain new ideas for decision support

- We have applied IND-NIMBUS®
  - application independent
- 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

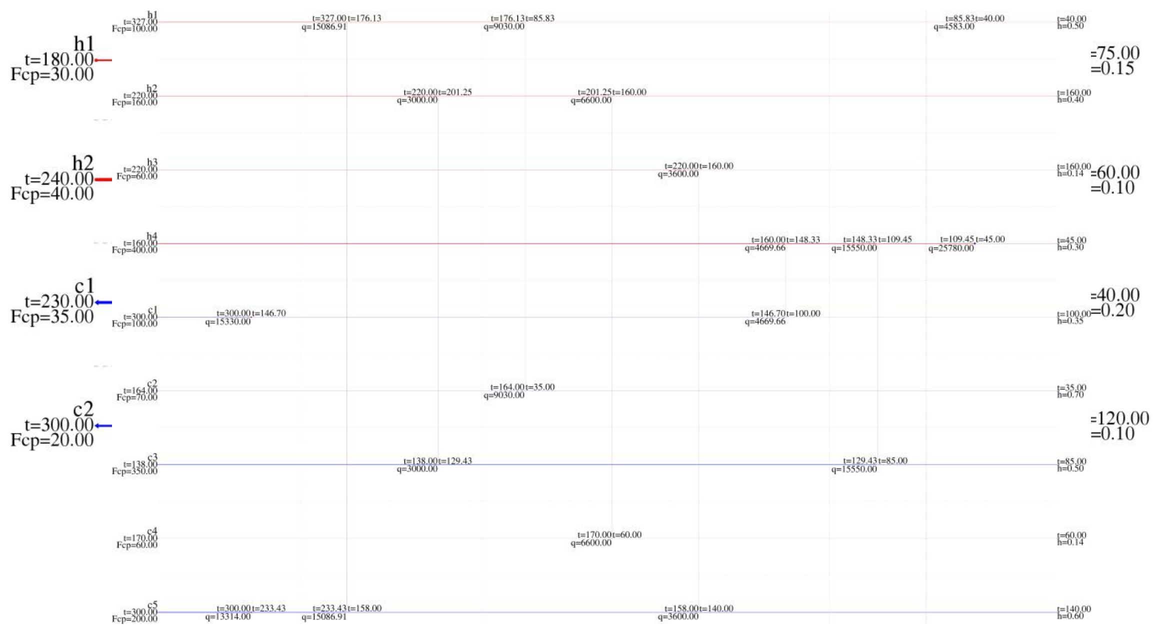
Some Applications

- Continuous casting of steel
  - Miettinen et al., Comput Opt & Appl (1998)
  - Miettinen, Mater & Manuf Processes (2007)
- Headbox design for paper machines
  - Hämäläinen et al., JOTA (2003)
- Paper quality in paper machine design
  - Madetoja et al., Eng with Comp (2006)
- Ultrasonic transducer design
  - Heikkola et al., Ultrasonics (2006)
- Chemical process design
  - Hakanen et al., JMCDA (2005)
  - Hakanen et al., Appl Therm Eng (2006)
- Simulated moving bed processes
  - Hakanen et al., Cont & Cyb (2007)
- Heat exchange network synthesis
  - Laukkanen et al., Comp Chem Eng (2010)
- Wastewater treatment system planning
  - Hakanen et al. DSS(2011), Env Mod (2013)
- Brachytherapy and IMRT
  - Ruotsalainen et al., Phys Med Biol (2010)
- Paper machine: both design and operation
  - Steponavice et al., Comp-Aided Design (2014)

Heat Exchanger Network Synthesis

- Simultaneous heat exchanger network synthesis model solved as a true multiobjective problem
- Objectives: utility cost, fixed cost of units and the cost related to the size of the heat exchangers
- **GAMS** model solved using interactive NIMBUS method
  - single objective optimizers of GAMS available

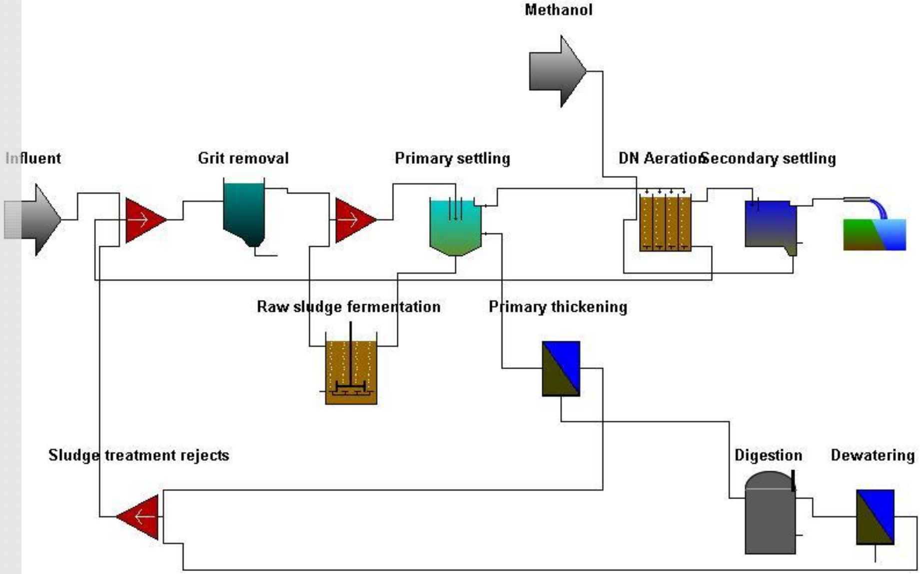
➢ Laukkanen et al., Computers and Chem Eng (2010)



Collaboration with Aalto University of Technology and Sciences

Optimization of Wastewater Treatment by Process Modelling and Simulation

- Challenges
  - operational requirements (e.g. effluent limits of nitrogen and phosphorus) getting more stringent
  - economical efficiency (e.g. min plant footprint, consumption of chemicals and energy)
  - operational reliability
- Conflict: quality of the treated wastewater vs. operational costs
- Interactive tool for designers combining commercial simulator and interactive decision making
- Advantages
  - conflicting objectives considered simultaneously
  - easier to formulate obj.functions
  - novel perspectives for designers
- ➢ Hakanen et al., DSS (2011), Env Model & Softw (2013)



Collaboration with PÖYRY

Paper Machine Headbox Design

- 100-150 meters long, width up to 11 meters
- Four main components
  - headbox
  - former
  - press
  - drying
- In addition, finishing

- Objectives
  - qualitative properties
  - save energy
  - use cheaper fillers and fibres
  - produce as much as possible
  - save environment

- First design problem: Headbox outlet height control
- Then chain of unit process models: virtual paper machine
- Optimize e.g. gloss, roughness, basis weight, fibre orientat.

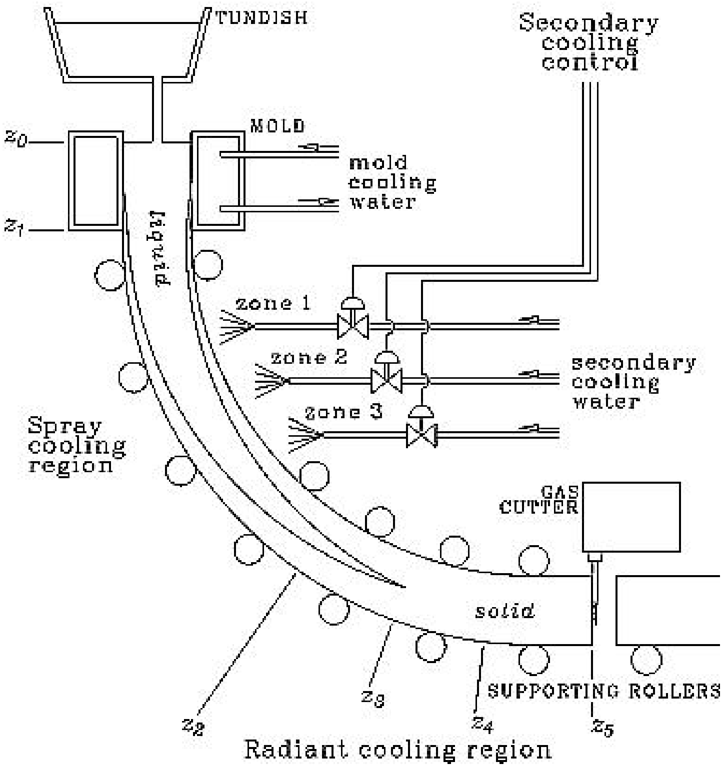
Collaboration with Metso Paper

Headbox Design cont.

Earlier

- Weighting method
  - how to select the weights?
  - how to vary the weights?
- Genetic algorithm
  - two objectives
  - computational burden
- First model with NIMBUS
  - turned out: model did not represent the actual goals
  - thus, it was difficult for the DM to specify preference information

Continuous Casting of Steel

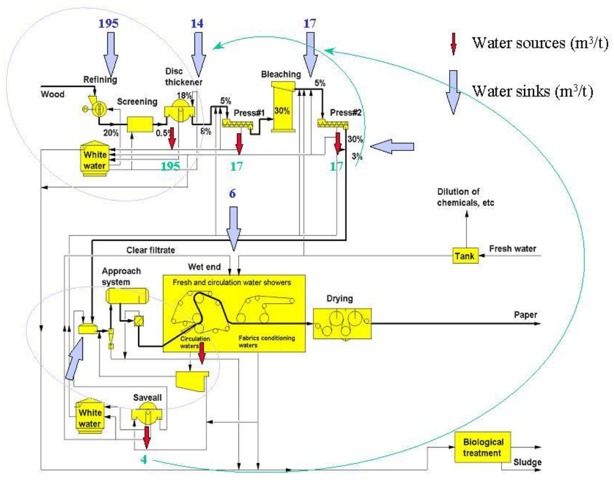


- Control of secondary cooling; intensity of water sprays affects solidification rate of steel
- Quality of steel depends on behavior of surface temperature and solidification front in time
- Originally, empty feasible region
- Constraints into objectives: minimize constraint violations
  - Keep the surface temperature near a desired temperature
  - Keep the surface temperature between some upper and lower bounds
  - Avoid excessive cooling or reheating on the surface
  - Restrict the length of the liquid pool
  - Avoid too low temperatures at the yield point

➢ Miettinen et al., Comput Opt & Appl (1998), Miettinen, Mater & Manuf Processes (2007)

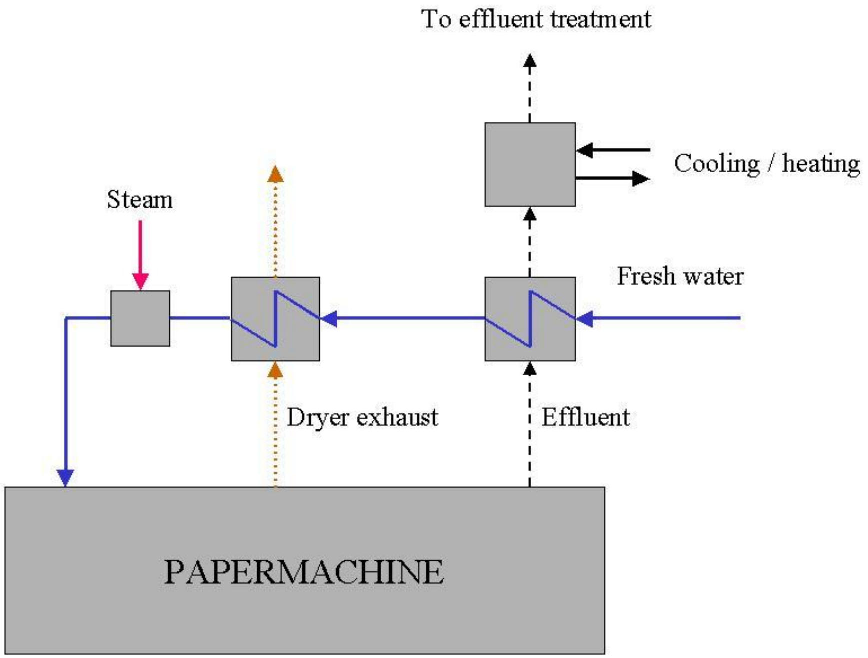
Process Simulation in Chemical Engineering

- Using BALAS® process simulator (product of VTT Finland)
- Flowsheet of process designed with BALAS® provides a simulation model to be optimized with IND-NIMBUS
  - Heat recovery: organize heat management taking seasonal changes in climate into account (typically single objective of annualized energy and investment costs, estimated amortization time and interest rate for capital)
  - Water allocation (recycle water in the process)
  - Collaboration with VTT



Heat Recovery System

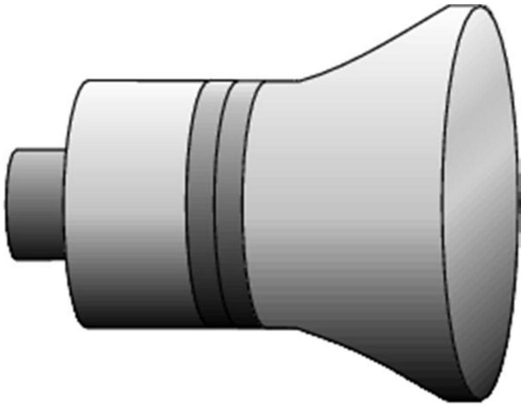
- Heat recovery system design for process water system of a paper mill
- Main trade-off between running costs, i.e., energy and investment costs
- 4 objective functions
  - steam needed for heating water for summer conditions
  - steam needed for heating water for winter conditions
  - estimation of area for heat exchangers
  - amount of cooling or heating needed for effluent
- 3 decision variables
  - area of the effluent heat exchanger
  - approach temperatures of the dryer exhaust heat exchangers for both summer and winter operations

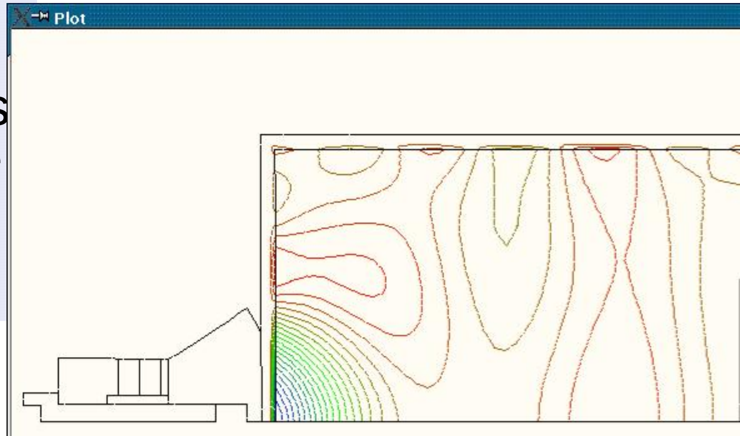


$$
\begin{array}{ll}
\text{minimize} & \{f_1(\mathbf{y}(\mathbf{x})), \ldots, f_4(\mathbf{y}(\mathbf{x}))\} \\
\text{subject to} & \mathbf{F}(\mathbf{y}(\mathbf{x})) = \mathbf{y}(\mathbf{x}) - \tilde{\mathbf{y}}(\mathbf{y}(\mathbf{x})) = 0 \\
 & \mathbf{x} \in S,
\end{array}
$$

Ultrasonic Transducer

- High-power ultrasonics creates strong ultrasonic vibration fields in solids and fluids, low ultrasonic frequencies (20-100 kHz)
- Vibrations cause intense effects: cavitation, steaming&heating needed in sonochemistry, cleaning, welding, etc.
- 3 objectives
  - **Minimize axial vibration** (attachment point vibrates as little as possible)
  - **Minimize electric impedance** (implying less power loss & less interference with other electric equipment)
  - **Minimize acoustic pressure** near transducer front
    (reduce strong cavitation effects at the container wall)
- Even the starting solution of IND-NIMBUS was better than current design (in terms of all three objectives)
- Need for global single obj. optimizer evident

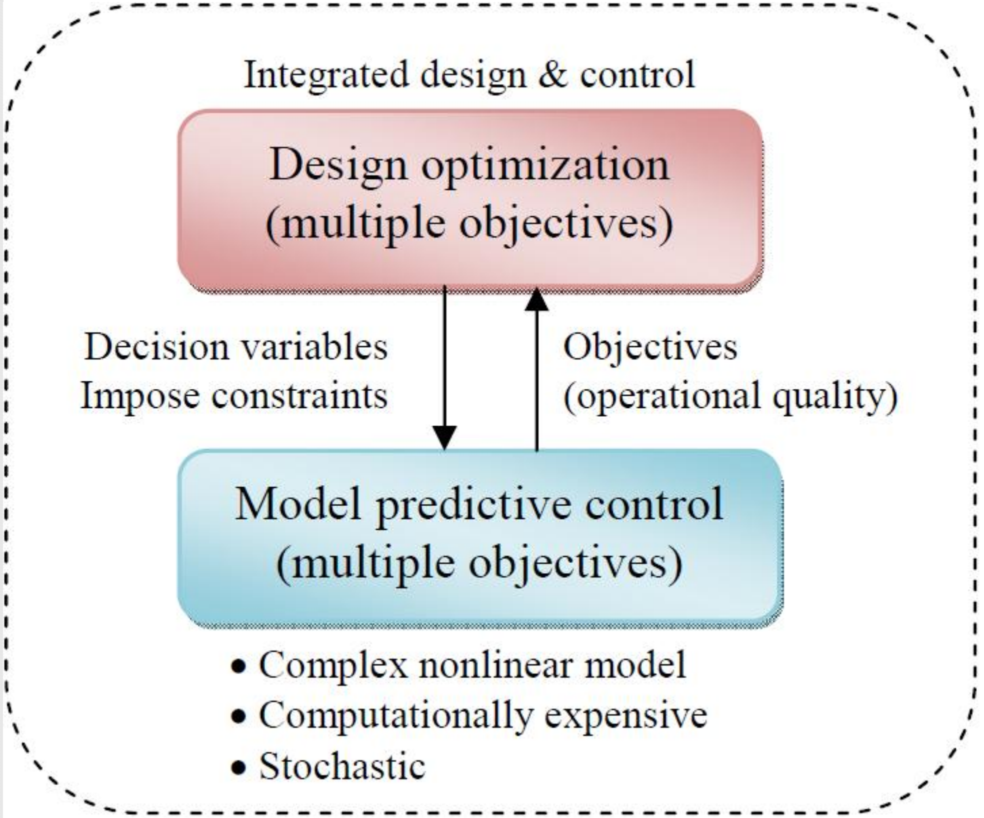


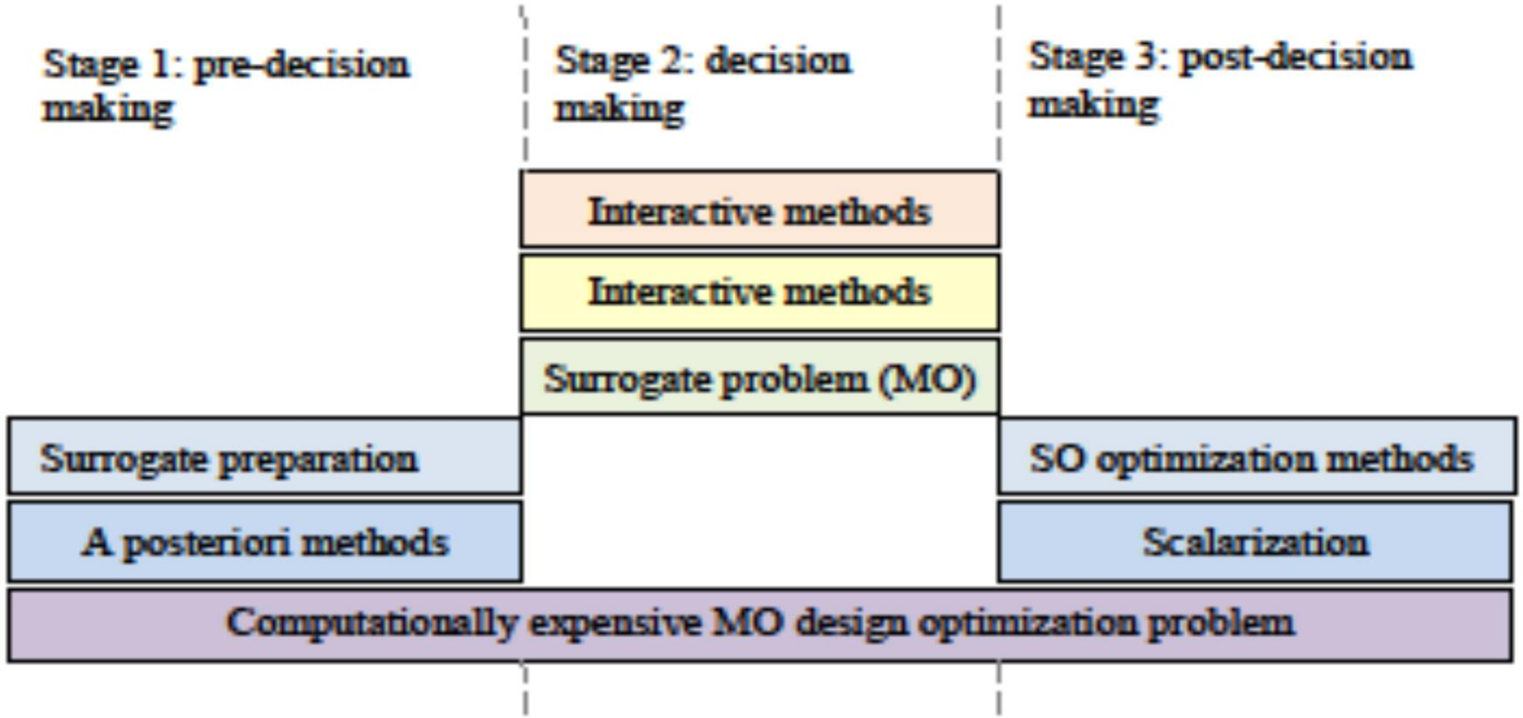


numerola
numerical software development

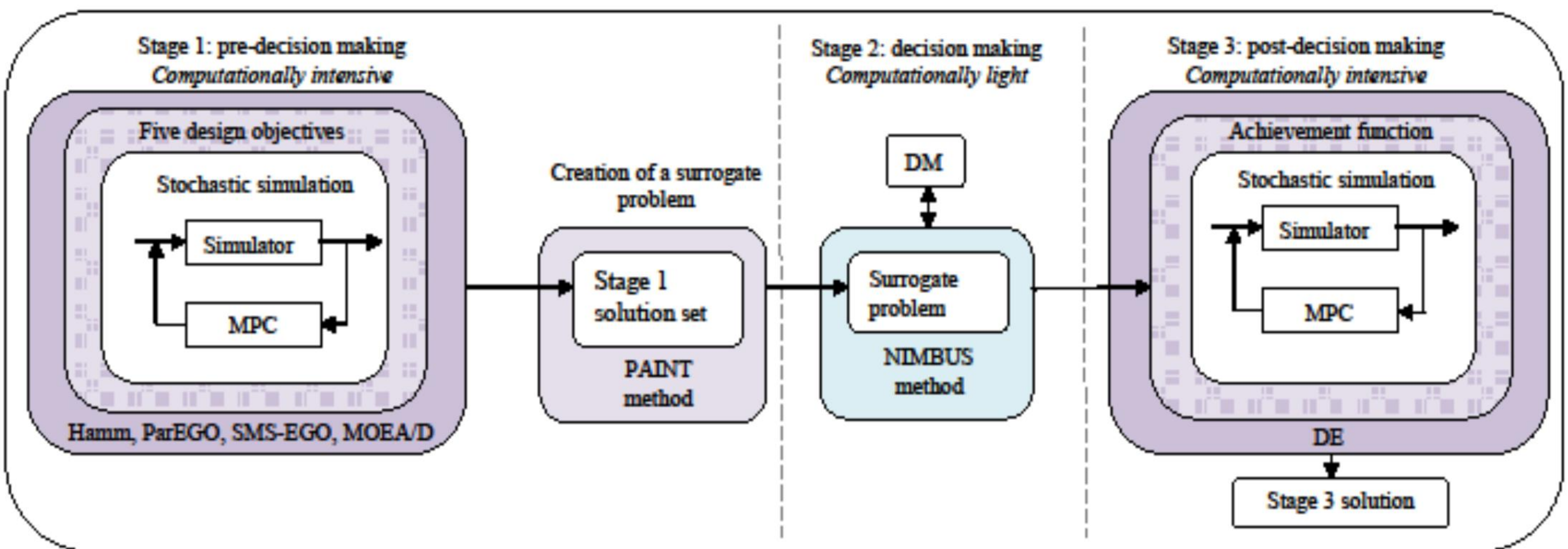
Desing and Operation of Paper Machine

- Hierarchical structure
  - design problem on upper level
  - operation optim. on lower level
  - ➢ multiple objective on both levels
- Targets
  - ❖ on design level
    - ❖ decrease investment cost
    - ❖ increase quality of paper
  - ❖ on operational level
    - ❖ guarantee runnability and stability of production system
- Design level objectives
  - Min long term averages of operational objectives (variations in filler content, basis weight, and paper strength)
  - Max runtime
  - Min investment cost i.e. tower volumes





Stage 1: pre-decision making
Stage 2: decision making
Stage 3: post-decision making
Interactive methods
Interactive methods
Surrogate problem (MO)
Surrogate preparation
SO optimization methods
A posteriori methods
Scalarization
Computationally expensive MO design optimization problem



Stage 1: pre-decision making
Computationally intensive
Five design objectives
Stochastic simulation
Simulator
MPC
Hamm, ParEGO, SMS-EGO, MOEA/D
Creation of a surrogate problem
Stage 1 solution set
PAINT method
Stage 2: decision making
Computationally light
DM
Surrogate problem
NIMBUS method
Stage 3: post-decision making
Computationally intensive
Achievement function
Stochastic simulation
Simulator
MPC
DE
Stage 3 solution

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

Method Development Challenges

- Complex problems
  - High dimensions (n and k)
  - Computational cost (metamodels vs. interpolation, new methods)
  - Uncertainty (scenarios, distributions)
  - Stochasticity
  - Robustness
  - Bi- and multilevel problems
  - Model predictive control
- User interface design - usability
- Better decision support (strengths of humans vs computers) – new devices and platforms
- Automatic decision support
  - How to build intelligent systems that learn DM’s preferences (incomplete, uncertain, verbal information)

Conclusions

- Multiobjective optimization problems **can** be solved!
- Complex problems as a whole – not only suboptima
- New insight of complex phenomena – no simplifications
- Role of DM emphasized: is in control, gets decision support and **learns**
- We can find solutions that could not have been found otherwise
- Applications everywhere
- Selecting a method: features of problem, opinions of DM, practical applicability
- **Compromise *is better* than optimum!**

Acknowledgements

- Collaboration: collaborators in different projects and Industrial Optimization Group http://www.mit.jyu.fi/optgroup/
- Funding: Tekes, Finnish Funding Agency for Technology and Innovation, Academy of Finland, companies

International Society on Multiple Criteria Decision Making

- http://www.mcdmsociety.org/
- About 1500 members in 100 countries
- No membership fees
- Electronic newsletter (2 times/year)
- Contact secretary@mcdmsociety.org if you wish to join
- International Conferences every two years
- MCDM2015 in August in Hamburg, Germany