(Nonlinear) Multiobjective Optimization

Kaisa Miettinen

miettine@hse.fi

Helsinki School of Economics

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

Motivation

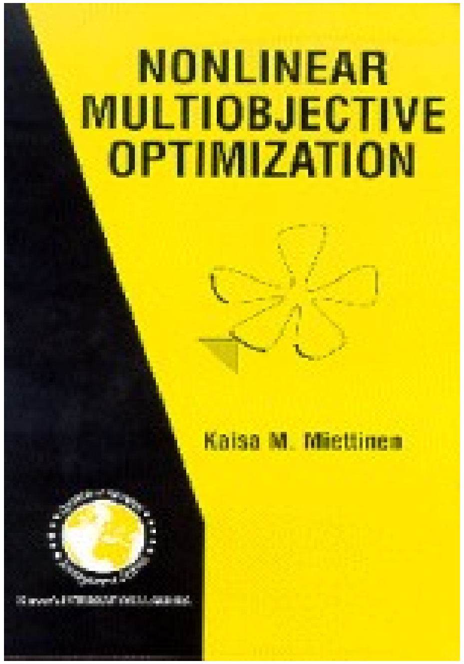
- Optimization is important
  - Not only what-if analysis or trying a few solutions and selecting the best of them
- Most real-life problems have several conflicting criteria to be considered simultaneously
- Typical approaches
  - convert all but one into constraints in the modelling phase or
  - invent weights for the criteria and optimize the weighted sum
  - but this simplifies the consideration and we lose information
- Genuine multiobjective optimization
  - Shows the real interrelationships between the criteria
  - Enables checking the correctness of the model
- Very important: less simplifications are needed and the true nature of the problem can be revealed
- The feasible region may turn out to be empty $\Rightarrow$ we can continue with multiobjective optimization and minimize constraint violations

Problems with Multiple Criteria

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

→ Nonlinear multiobjective optimization

Contents



- *Nonlinear Multiobjective Optimization* by Kaisa M. Miettinen, Kluwer Academic Publishers, Boston, 1999
- Concepts
- Optimality
- Methods (in 4 classes)
- Tree diagram of methods
- Graphical illustration
- Applications
- Concluding remarks

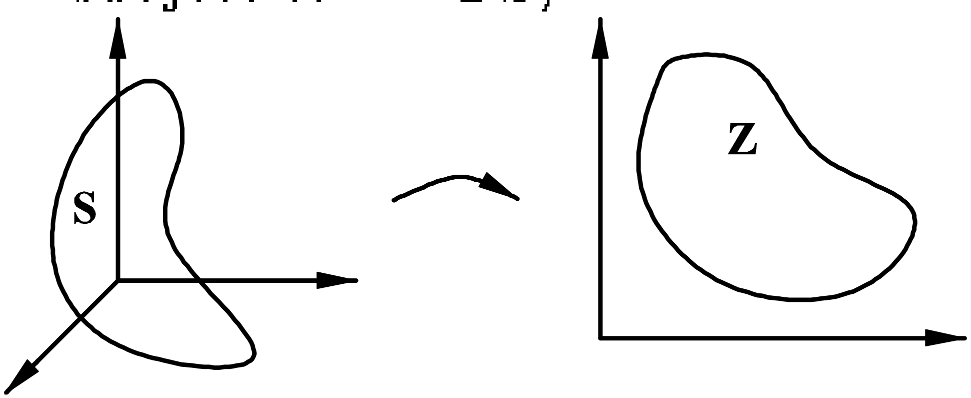
Concepts

We consider multiobjective optimization problems

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

in other words

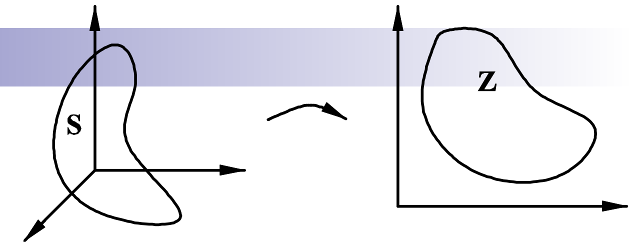
$$\begin{array}{ll} \text{minimize} & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\} \\ \text{subject to} & \mathbf{x} \in S, \end{array}$$



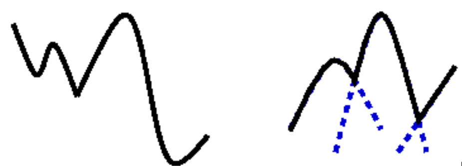
where

- **f_i : $R^n \rightarrow R$ = *objective function***
- **k (≥ 2) = number of (conflicting) objective functions**
- **x = *decision vector* (of n *decision variables* x_i)**
- **S $\subset$ R^n = *feasible region* formed by *constraint functions* and**
- **\`\`minimize´´ = minimize the objective functions simultaneously**

Concepts cont.



- S consists of linear, nonlinear (equality and inequality) and box constraints (i.e. lower and upper bounds) 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 $z \in Z$
- Remember: maximize $f_i(x)$ = - minimize - $f_i(x)$
- We call a function *nondifferentiable* if it is locally Lipschitzian



<u>Definition</u>:

If all the (objective and constraint) functions are linear, the problem is *linear* (*MOLP)*. If some functions are nonlinear, we have a *nonlinear multiobjective optimization problem (MONLP)*. The problem is *nondifferentiable* if some functions are nondifferentiable and *convex* if all the objectives and S are convex

Optimality

- **Contradiction and possible incommensurability $\Rightarrow$**
- **<u>Definition</u>: A point $x^* \in S$ is (globally) *Pareto optimal* (PO) if there does not exist another point $x \in S$ 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. An objective vector $z^* \in Z$ is Pareto optimal if the corresponding point x^* is Pareto optimal.**

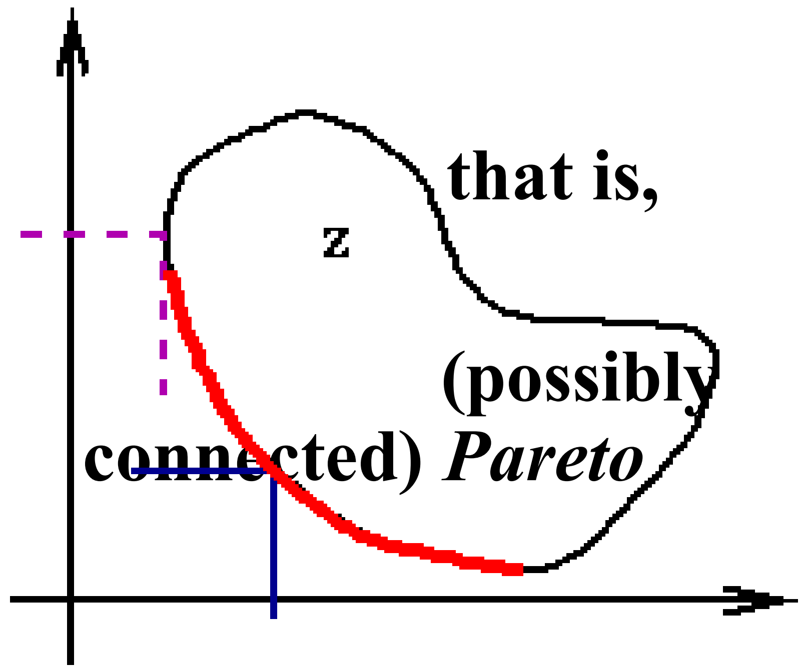
In other words,

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

that is,

$$(z^* - R^k_+) \cap Z = z^*$$

- **Pareto optimal solutions form (possibly nonconvex and non-connected) *Pareto optimal set***



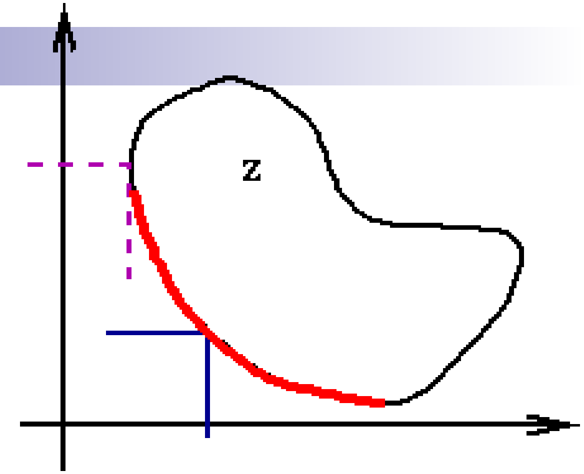
Theorems

- **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 (KKT) (necessary and sufficient) optimality conditions can be formed as a natural extension to single objective optimization for both differentiable and nondifferentiable problems**

Optimality cont.

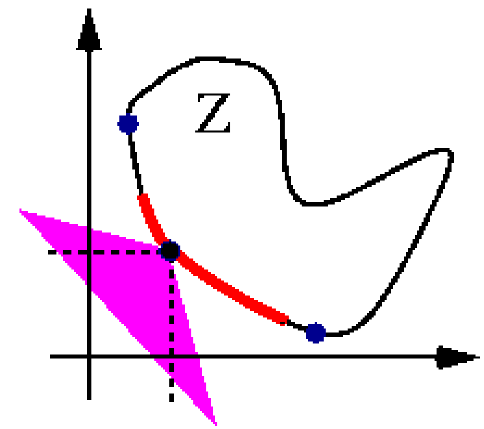
- **Paying attention to the Pareto optimal set and forgetting other solutions is acceptable only if we know that no unexpressed or approximated objective functions are involved!**
- **A point $x^* \in S$ is *locally Pareto optimal* if it is Pareto optimal in some environment of x^***
- **Global Pareto 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**

Optimality cont.

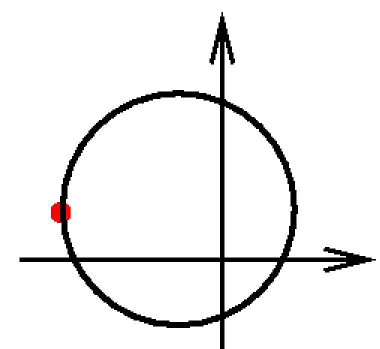


- **Definition: A point $x^* \in S$ is *weakly Pareto optimal* if there does not exist another point $x \in S$ such that $f_i(x) < f_i(x^*)$ for all $i = 1,\ldots,k$. That is,**

  $(z^* - \text{int } R^k_+) \cap Z = \varnothing$



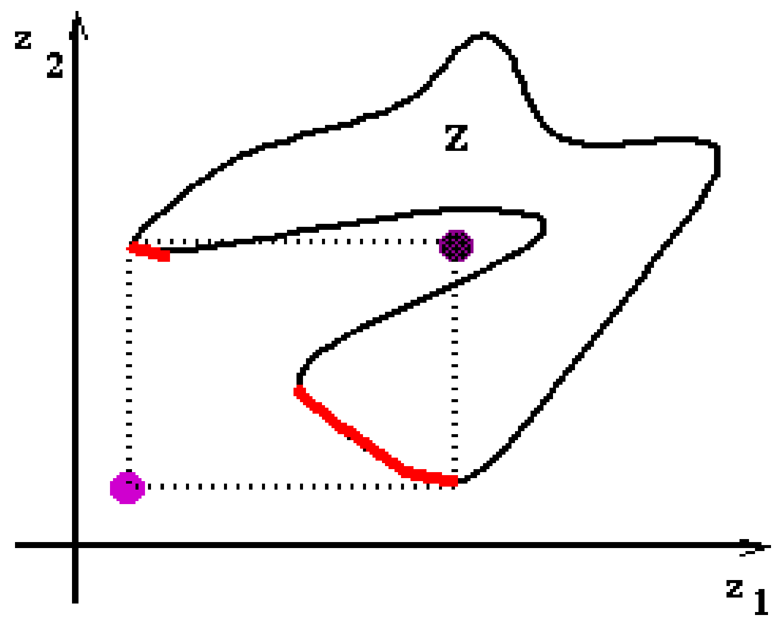
- **Pareto optimal points can be *properly* or improperly PO**
- **Properly PO: unbounded trade-offs are not allowed. Several definitions... Geoffrion:**



$\mathbf{x}^* \in S$ is properly PO if it is PO and $\exists\ M > 0$ s.t. $\forall f_i, \mathbf{x} \in S$ satisfying $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ $\exists\ f_j$ such that $f_j(\mathbf{x}^*) < f_j(\mathbf{x})$ and $\frac{f_i(\mathbf{x}^*) - f_i(\mathbf{x})}{f_j(\mathbf{x}) - f_j(\mathbf{x}^*} \leq M$.

Concepts cont.

- A *decision maker* (DM) is needed to identify a final Pareto optimal solution. (S)he has insight into the problem and can express preference relations
- 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* $z^{\star}$ and approximated *nadir objective vector* z^{nad}
- *Ideal objective vector* = individual optima of each f_i
- Utopian objective vector $z^{\star\star}$ is strictly better than $z^{\star}$
- Nadir objective vector can be approximated from a payoff table but this is problematic



Concepts cont.

- Value function $U:R^k \rightarrow R$ may represent preferences and sometimes DM is expected to be maximizing value (or utility)
- 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 in methods
- 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* $\check{z} \in R^k$
- Problems are usually solved by *scalarization*, where a real-valued objective function is formed (depending on parameters). Then, single objective optimizers can be used!

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,…,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***
- **Definition: 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 $\forall\ 0 \leq \alpha \leq \alpha^*$, then λ_{ij} is a *partial trade-off rate***

Marginal Rate of Substitution

- **Remember: 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

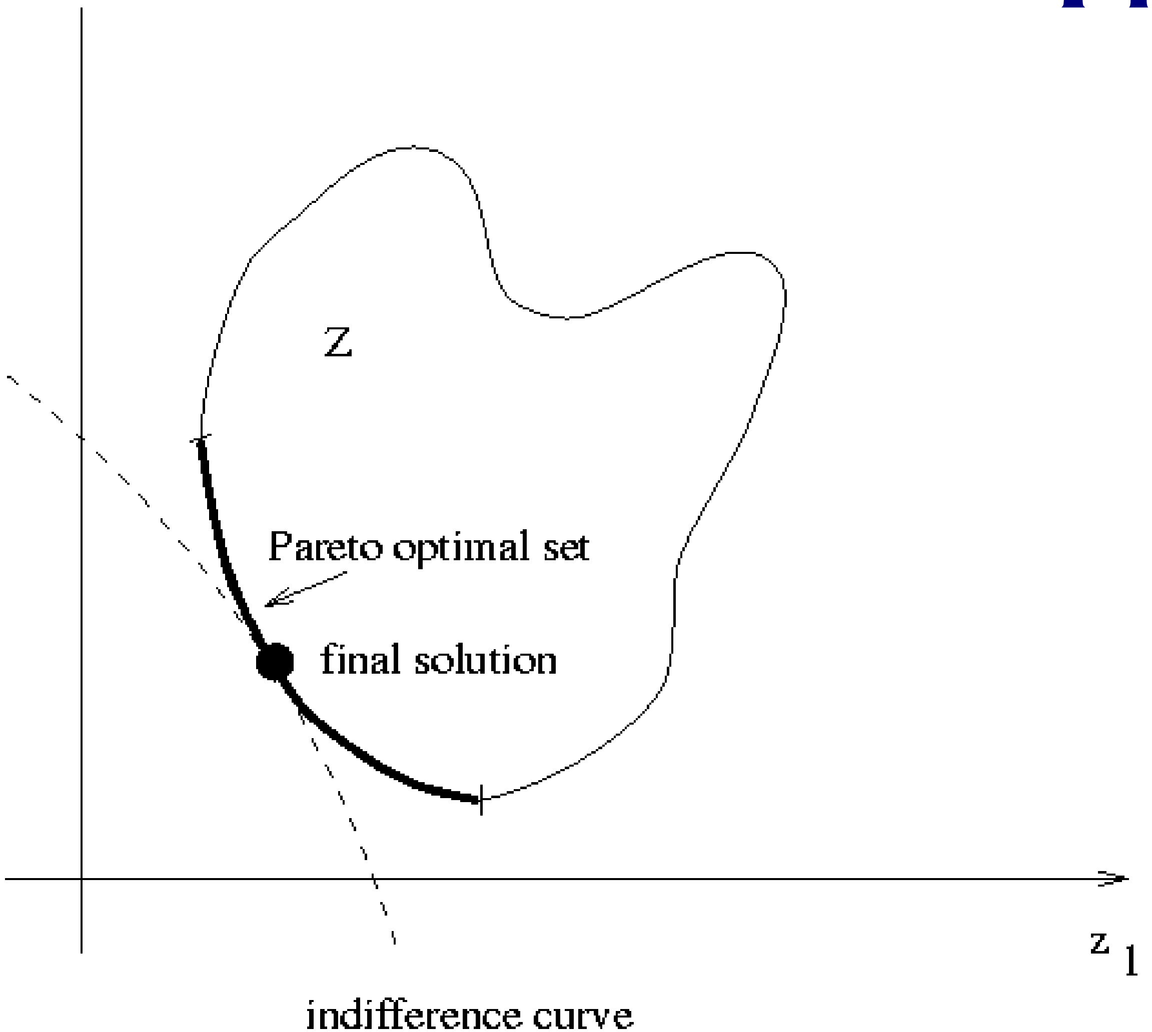


Figure 1. The final solution.

Testing Pareto Optimality (Benson)

- **x* is Pareto optimal if and only if**

$$
\begin{array}{ll}
\text{maximize} & \sum_{i=1}^{k} \varepsilon_i \\
\text{subject to} & f_i(\mathbf{x}) + \varepsilon_i \le f_i(\mathbf{x}^*) \text{ for all } i = 1, \dots, k \\
 & \varepsilon_i \ge 0 \text{ for all } i = 1, \dots, k \\
 & \mathbf{x} \in S
\end{array}
$$

value 0. Otherwise, the solution obtained is PO

Methods

- **Solution = best possible compromise**
- **Decision maker (DM) is responsible for final solution**
- **Finding a Pareto optimal set or a representation of it = *vector optimization***
- **Method differ, for example, in: What information is exchanged, how scalarized**
- **Two criteria**
  - **Is the solution generated PO?**
  - **Can any PO solution be found?**
- ***Classification***
  - **according to the role of the DM:**
    - **no-preference methods**
    - **a posteriori methods**
    - **a priori methods**
    - **interactive methods**
  - **based on the existence of a value function:**
    - **ad hoc: U would not help**
    - **non ad hoc: U helps**

Methods cont.

- No-preference methods
  - Meth. of Global Criterion
- A posteriori methods
  - Weighting Method
  - ε-Constraint Method
  - Hybrid Method
  - Method of Weighted Metrics
  - Achievement Scalarizing Function Approach
- A priori methods
  - Value Function Method
  - Lexicographic Ordering
  - Goal Programming
- Interactive methods
  - Interactive Surrogate Worth Trade-Off Method
  - Geoffrion-Dyer-Feinberg Method
  - Tchebycheff Method
  - Reference Point Method
  - GUESS Method
  - 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:**

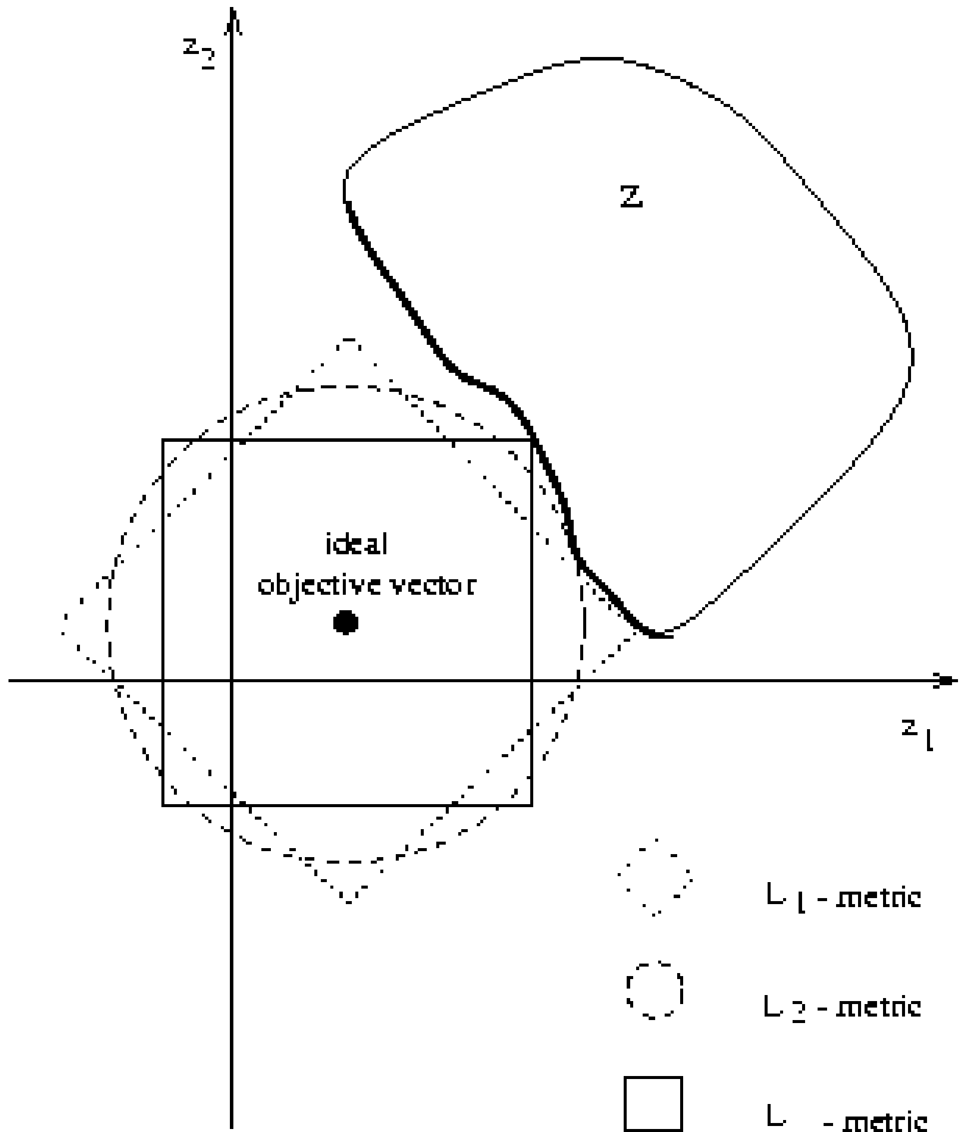
$$\begin{array}{ll} \text{minimize} & \max_{1 \le i \le k} [f_i(\mathbf{x}) - z_i^\star] \\ \text{subject to} & \mathbf{x} \in S. \end{array}$$

- **Differentiable form of the latter:**

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

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 (or a part 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? (Difficult to present the PO set)

Weighting Method (Gass, Saaty)

- **Problem**

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

- ≈ **Solution is weakly PO**
- \+ **Solution is PO if it is unique or $w_i > i$**
- \+ **Convex problems: any PO solution can be found**
- − **Nonconvex problems: some of the PO solutions may fail to be found**

Weighting Method cont.

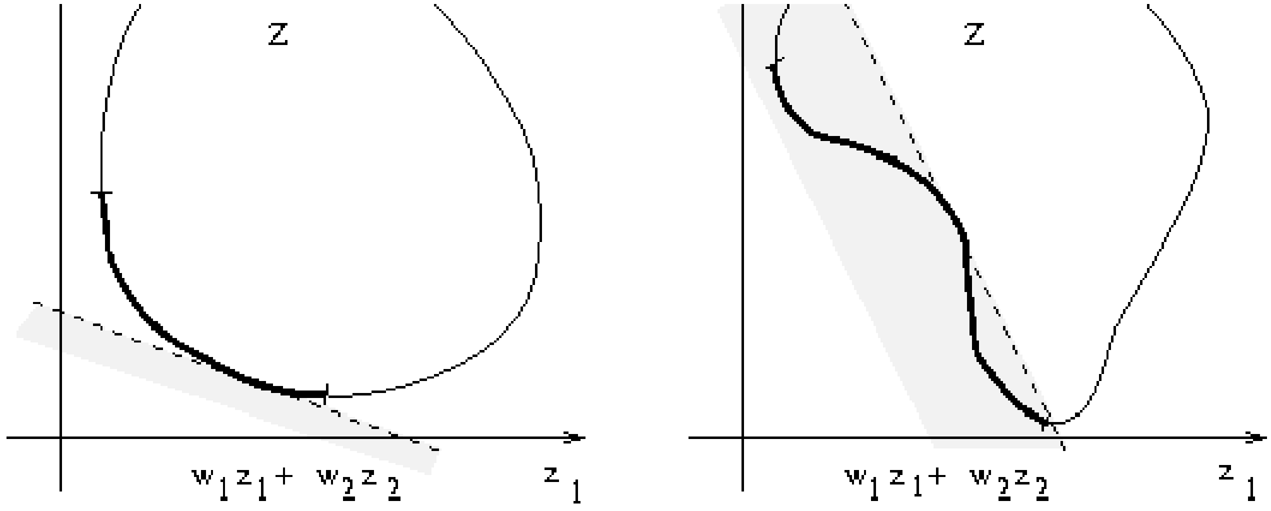


Figure 3. Convex and nonconvex problems.

- 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

Why not Weighting Method

Selecting a wife (maximization problem):

	beaut y	cookin g	house - wifer y	tidi - nes s	
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	

ε-Constraint Method (Haimes et al)

- **Problem**

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

≈ The solution is weakly Pareto optimal

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

+ A unique solution is PO

+ Any PO solution can be found

– 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 $\lambda_j = \lambda_{\ell j}$ are (partial or total) trade-off rates**

Hybrid Method (Wendell et al)

- Combination: weighting + ε-constraint methods
- <u>Problem</u>: where $\forall\ i=1,\ldots,k$

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \\ \text{subject to} \quad & f_j(\mathbf{x}) \le \varepsilon_j \quad \text{for all} \quad j = 1, \ldots, k, \\ & \mathbf{x} \in S, \end{aligned}$$

+ The solution is PO any ε
+ Any PO solution can be found
- The PO set can be found by solving the problem with methods for parametric constraints (where the parameter is ε). Thus, the weights do not have to be altered
+ Positive features of the two methods are combined
− The specification of parameter values may be difficult

Method of Weighted Metrics (Zeleny)

- **Weighted metric formulations are**

- **Absolute values may be needed**

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

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 $\exists$ 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 Scalarizing Functions

- *Achievement (scalarizing) functions* $s_{ž}: Z \to R$, where ž is any reference point. In practice, we minimize in S
- Definition: $s_{ž}$ is *strictly increasing* if $z_i^1 < z_i^2 \ \forall \ i=1,\ldots,k \Rightarrow s_{ž}(z^1) < s_{ž}(z^2)$. It is *strongly increasing* if $z_i^1 \leq z_i^2$ for $\forall \ i$ and $z_j^1 < z_j^2$ for some $j \Rightarrow s_{ž}(z^1) < s_{ž}(z^2)$
- $s_{ž}$ is *order-representing* under certain assumptions if it is strictly increasing for any ž
- $s_{ž}$ is *order-approximating* under certain assumptions if it is strongly increasing for any ž
- Order-representing $s_{ž}$: solution is weakly PO $\forall$ ž
- Order-approximating $s_{ž}$: solution is PO $\forall$ ž
- If $s_{ž}$ is order-representing, any weakly PO or PO solution can be found. If $s_{ž}$ is order-approximating 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 Scalarizing Function: MOLP

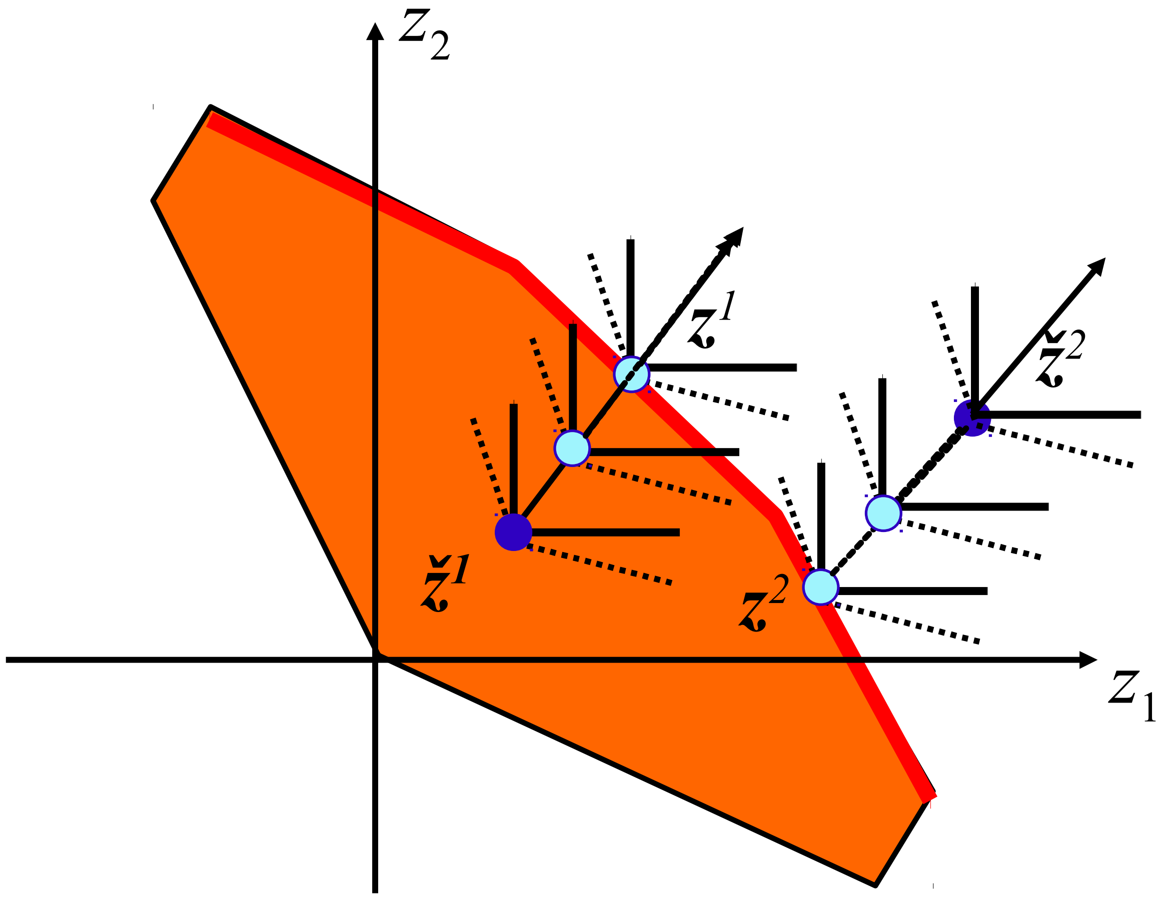


Figure from Prof. Pekka Korhonen

Achievement Scalarizing Function: MONLP

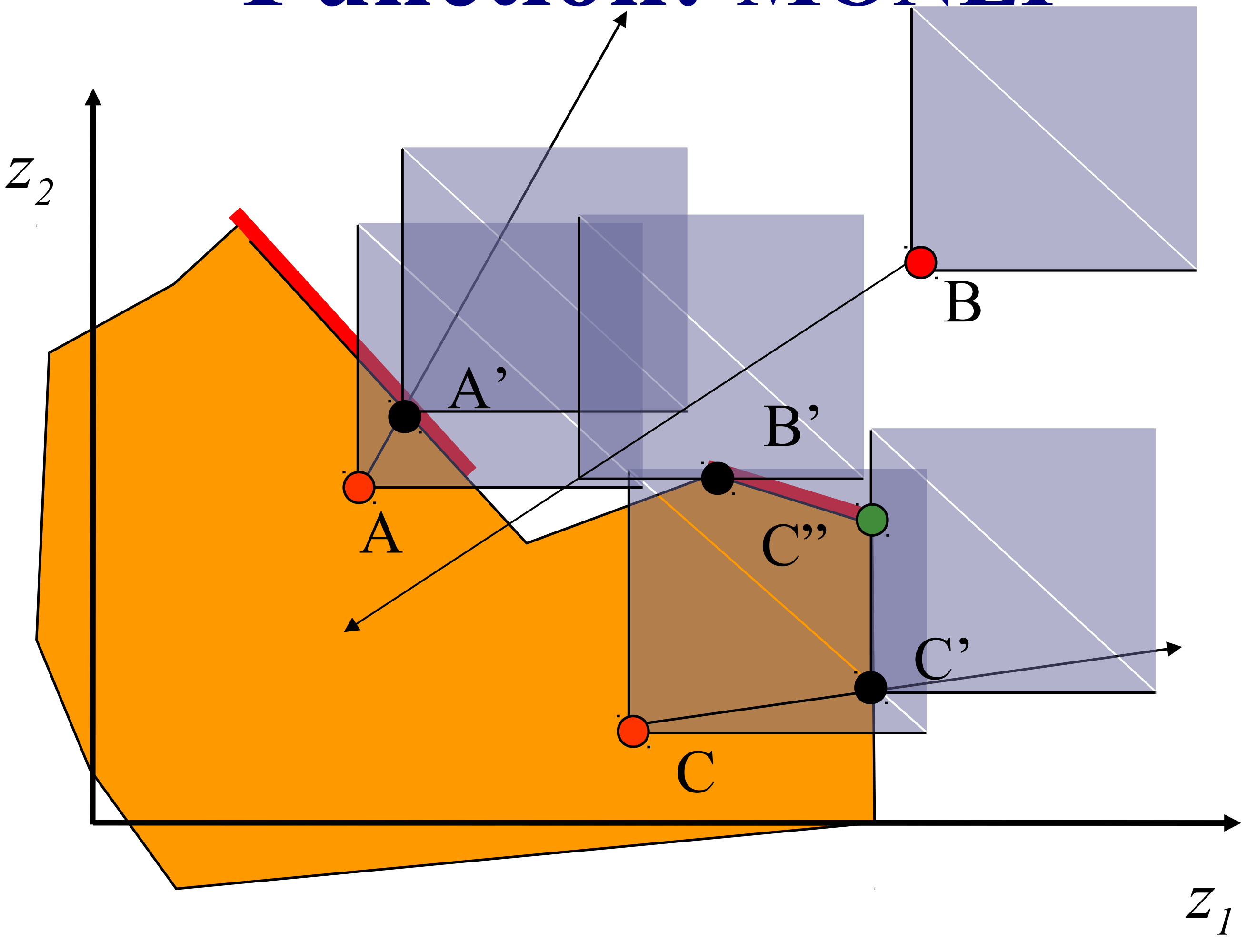


Figure from Prof. Pekka Korhonen

Multiobjective Evolutionary Algorithms

- Many different approaches
- VEGA, RWGA, MOGA, 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.
- Problem: with real problems, it remains unknown how far the solutions generated are from the true PO solutions

NSGA II (Deb et al)

- Includes elitism and explicit diversity-preserving mechanism
- Nondominated sorting – fitness=nondomination level (1 is the best)

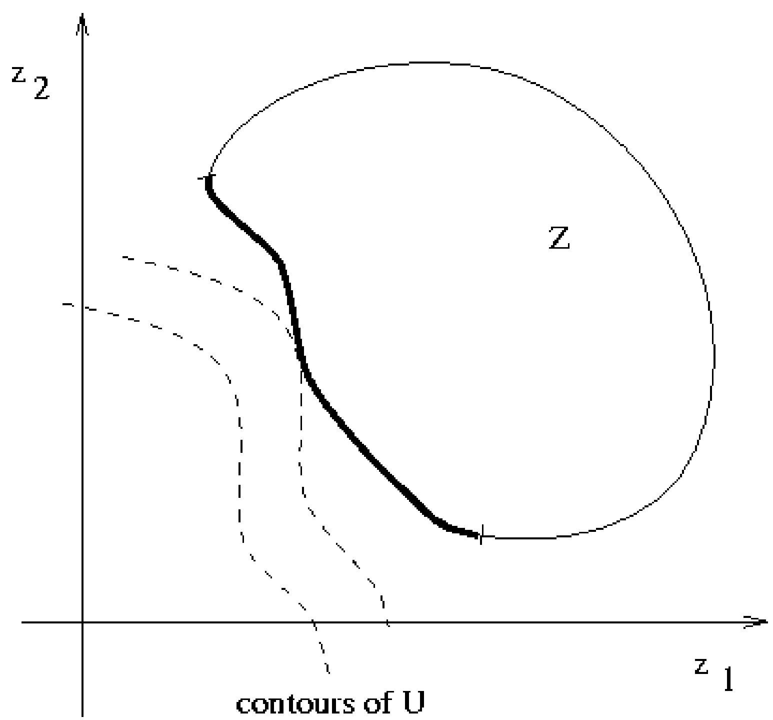
1. Combine parent and offspring populations (2N individuals) and perform nondominated sorting to identify different fronts F_i (i=1, 2, …)
2. Set new population = ;. Include fronts < N members.
3. Apply special procedure to include most widely spread solutions (until N solutions)
4. Create offspring population

A Priori Methods

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

Value Function Method (Keeney, Raiffa)

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



Variable, Objective and Value Space

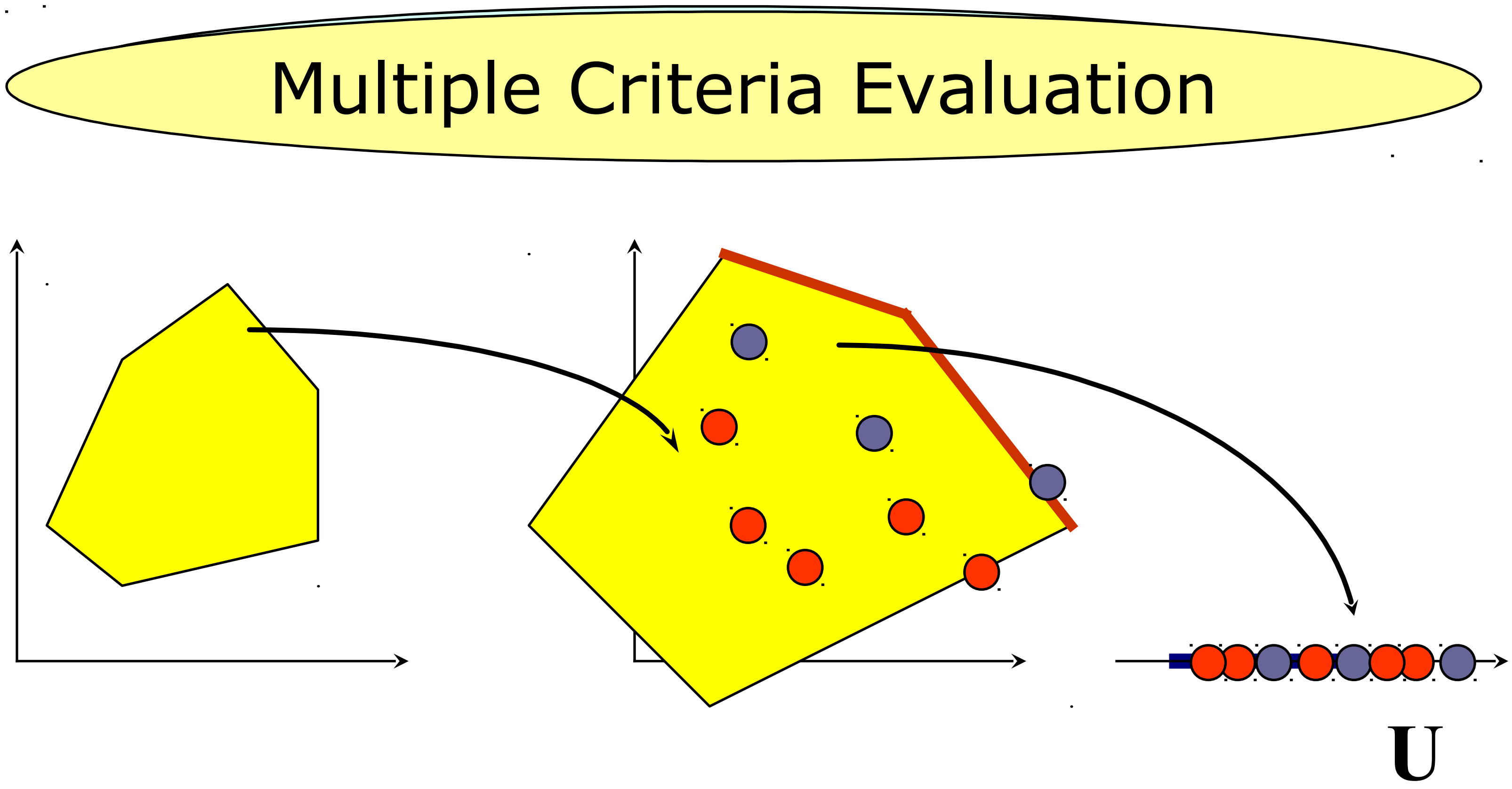


Figure from Prof. Pekka Korhonen

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/his U
- − Existence of U sets consistency and comparability requirements
- − Even if the explicit U was 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 PO**
  + **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 & aspiration level = a *goal*. Deviations from aspiration levels are minimized ($f_i(x) - \delta_i = \check{z}_i$)
- The deviations can be represented as overachievements $\delta_i > 0$
- <u>Weighted approach</u>:

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

  - Weights from the DM

Goal Programming cont.

- Lexicographic approach: the deviational variables are minimized lexicographically
- Combination: 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

- A solution pattern is formed and repeated
- Only some PO points are generated
- Solution phases - loop:
  - Computer: Initial solution(s)
  - DM: evaluate preference information – stop?
  - Computer: Generate solution(s)
- Stop: DM is satisfied, tired or stopping rule fulfilled
- DM can learn about the problem and interdependencies in it

Interactive Methods cont.

- Most developed class of methods
- DM needs time and interest for co-operation
- DM has more confidence in the final solution
- No global preference structure required
- DM is not overloaded with information
- DM can specify and correct preferences and selections as the solution process continues
- Important aspects
  - what is asked
  - what is told
  - how the problem is transformed

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 $\lambda_{\iota i} > 0$ $\forall i$ are partial trade-off rates between f_ι and f_i**
- **For all i the DM is told: ``If the value of f_ι is decreased by $\lambda_{\iota i}$, 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

- **Well-known method**
- **Idea: Maximize the DM's (implicit) value function with a suitable (Frank-Wolfe) gradient method**
- **Local approximations of the value function are made using marginal rates of substitution that the DM gives describing her/his preferences**
- **Assumptions**
  - **U is implicitly known, continuously differentiable and concave in S**
  - **objectives are continuously differentiable**
  - **S is convex and compact**

GDF Method cont.

- **The gradient of U at x^h:**

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

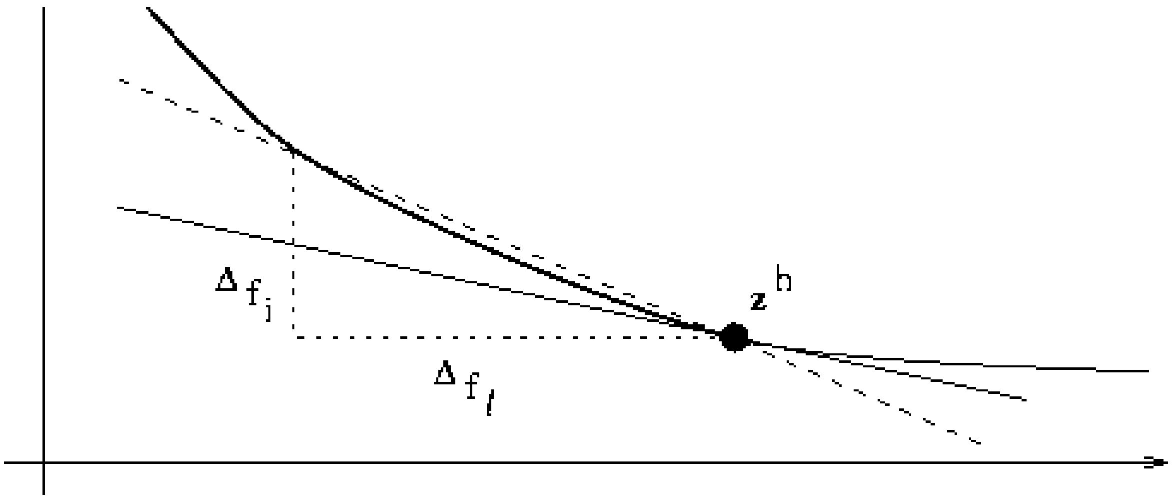
- **The direction of the gradient of U:**

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

where m_i is the marginal rate of substitution involving f_ι and f_i at x^h $\forall$ i, ($i \neq \iota$). They are asked from the DM as such or using auxiliary procedures

GDF Method cont.

- Marginal rate substitution is the slope of the tangent



- The direction of steepest ascent of U:

$$\text{maximize} \quad \mathbf{y}^T \sum_{i=1}^{k} -m_i \nabla_x f_i(\mathbf{x}^h)$$
$$\mathbf{y} \in S.$$

- Step-size problem: How far to move (one variable). Present to the DM objective vectors with different values for t in $f_i(x^h + td^h)$ ($i=1,\ldots,k$) where $d^h = y^h - x^h$

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_i 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:**

$$\text{lex min} \quad \max_{i=1,\ldots,k} \left[w_i(f_i(\mathbf{x}) - z_i^{\star\star}) \right], \sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^{\star\star})$$

$$\text{subj. to } \mathbf{x} \in S.$$

- **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 $z^{\star\star}$. 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: To direct the search by reference points using achievement functions (no 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.

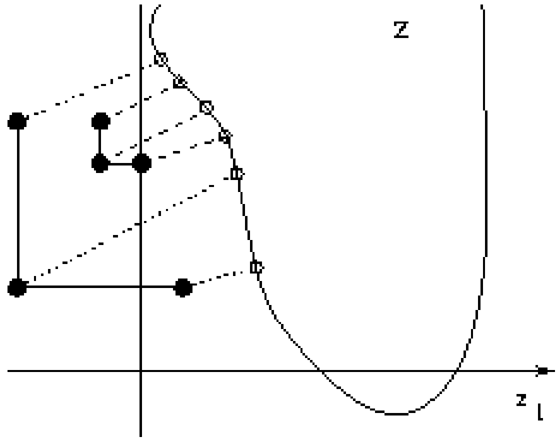


Figure 6. Altering the reference points.

- Ad hoc method both)
- DIDAS software

+ Easy for the DM to (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 $\bar{z}^h$ 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]

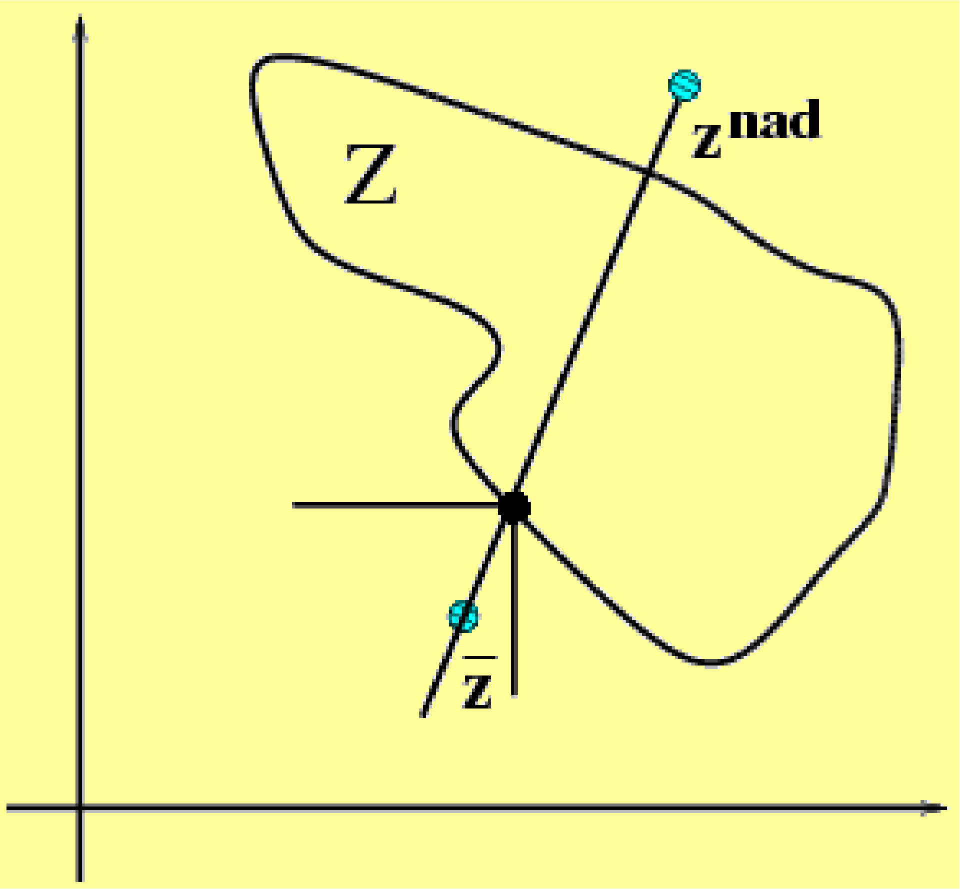
$$\frac{z_i^{\mathrm{nad}} - f_i(\mathbf{x})}{z_i^{\mathrm{nad}} - z_i^{\star}}.$$

→ <u>Problem</u>:

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

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

GUESS cont.



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:**

  minimize

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

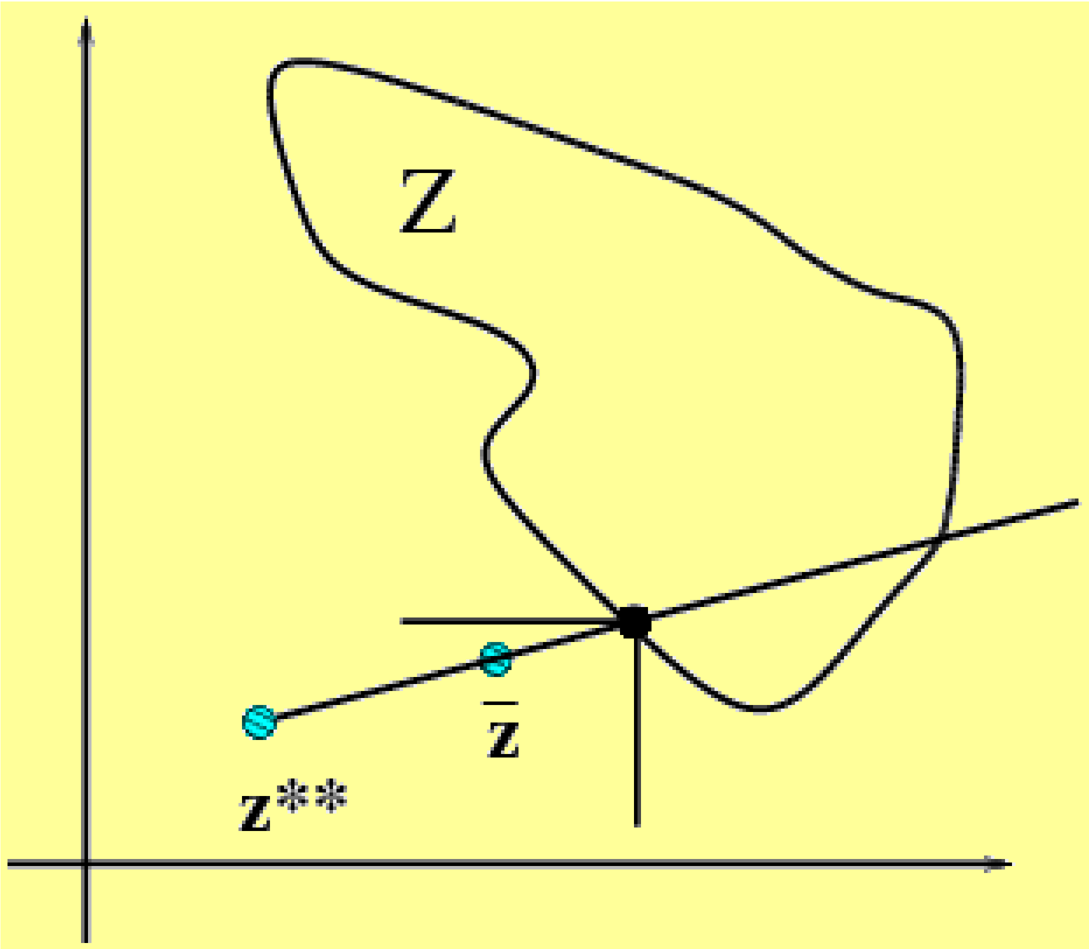
or

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

where žh > z◆◆ and ρ>0

- **Partial trade-off rate information can be obtained from optimal KKT multipliers of the differentiable counterpart problem**

Satisficing Trade-off Method cont.



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**

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)

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

- 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 cont.

- Establish *outranking relations* between alternatives. One alternative outranks the other if it is at least as good as the latter
- DM gives (for each objective) *indifference thresholds* = intervals where indifference prevails. Hesitation between indifference and preference = *preference thresholds*. A *veto threshold* prevents compensating poor values in some objectives
- Additional alternatives near the current solution (based on the reference point) are generated so that they outrank the current one

⇒ No incomparable/indifferent solutions shown

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.

- 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

NIMBUS Method (Miettinen, Mäkelä)

- Idea: move around Pareto optimal set
- How can we support the learning process?
- The DM should be able to direct the solution process
- Goals: easiness of use
  - What can we expect DMs to be able to say?
  - No difficult questions
  - Possibility to change one’s mind
- Dealing with objective function values is understandable and straightforward

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 $\varepsilon_i^h > f_i(x^h)$ ($i \in I^>$) and
  - are allowed to change freely ($i \in I^\diamond$)
- Functions in $I^\leq$ are to be minimized only till the specified level
- Assumption: ideal objective vector available
- DM must be willing to give up something

NIMBUS Method cont.

- **Problem**

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

where $\rho > 0$

- **Solution properly PO. Any PO solution can be found**
- **Any nondifferentiable single objective optimizer**
- **Solution satisfies desires as well as possible – feedback of tradeoffs**

Latest Development

- **Scalarization is important and contains preference information**
- **Normally method developer selects one scalarization**
- **But scalarizations based on same input give different solutions – Which one is the best? ⇒ Synchronous NIMBUS**
- **Different solutions are obtained using different scalarizations**
- **A reference point can be obtained from classification information**
- **Show them to the DM and let her/him choose the best**
- **In addition, intermediate solutions**

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.**

NIMBUS Method cont.

- 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

NIMBUS Software

- Mainframe version
  - + Applicable for even large-scale problems
  - − No graphical interface ⇒ difficult to use
  - − Trouble in delivering updates
- WWW-NIMBUS http://nimbus.it.jyu.fi/
  - ! Centralized computing & distributed interface
  - + Graphical interface with illustrations via WWW
  - + Applicable for even large-scale problems
  - + Latest version is always available
  - + No special requirements for computers
    - No computing capacity
    - No compilers
  - Available to any academic Internet user for free
  - + Nonsmooth local solver (proximal bundle)
  - + Global solver (GA with constraint-handling)

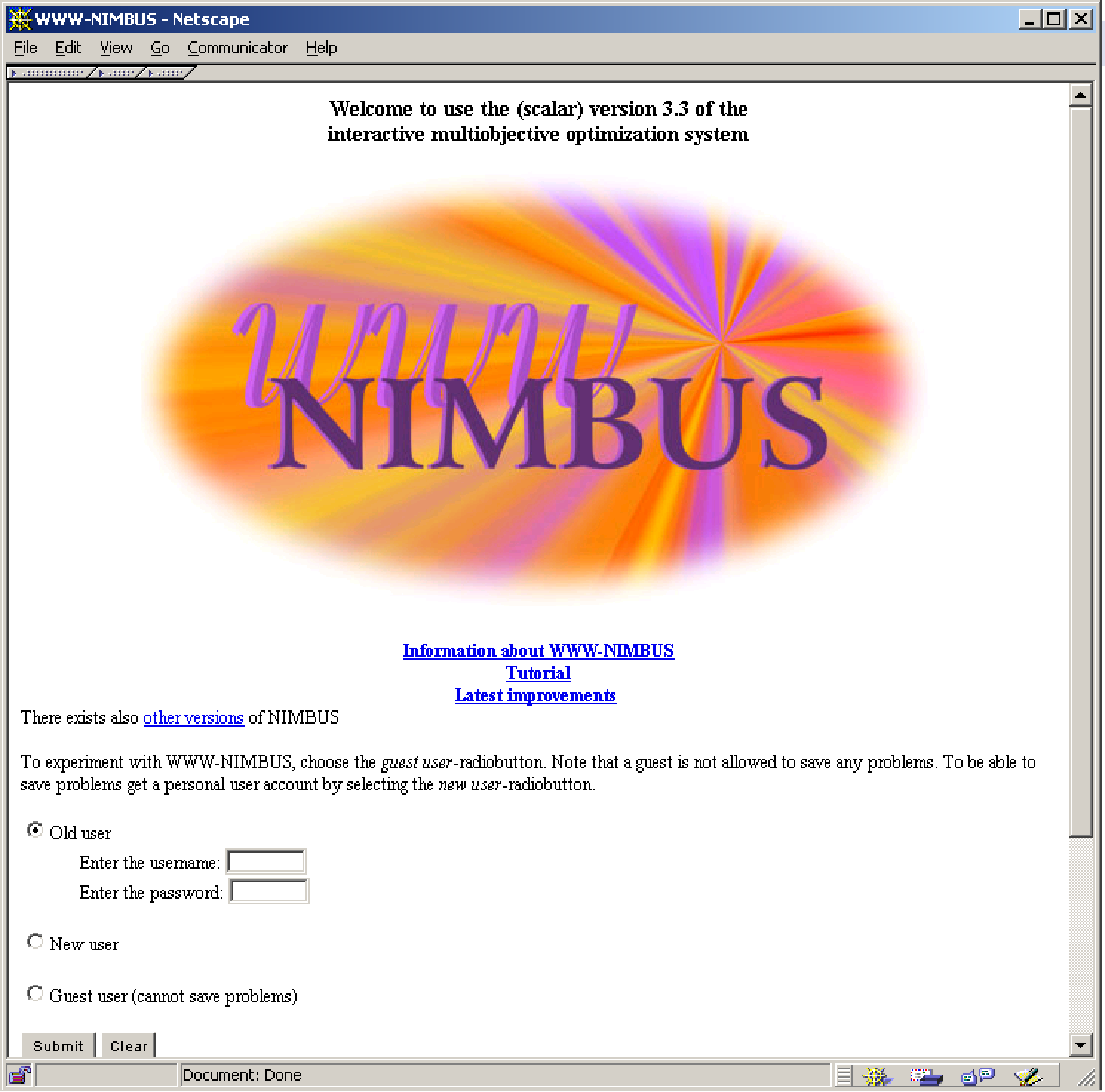
WWW-NIMBUS since 1995

- First, unique interactive system on the Internet
- Personal username and password
- Guests can visit but cannot save problems
- Form-based or subroutine-based problem input
- Even nonconvex and nondifferentiable problems, integer-valued variables
- Symbolic (sub)differentiation
- Graphical or form-based classification
- Graphical visualization of alternatives
  - Possibility to select different illustrations and alternatives to be illustrated
- Tutorial and online help
- Server computer in Jyväskylä

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

WWW-NIMBUS Version 4.1

- 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
- Nonsmooth local solver (proximal bundle) and global solver (GA with constraint-handling)
- Two different constraint-handling methods available for GA (adaptive penalties & parameter free penalties)
- 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

NIMBUS TUTORIAL HELP

Input Problem

Fill in all the fields, please. Note that the function and the variable names are used when displaying results, and the variable names x1, x2, x3, etc. must be used when defining the problem.

Objective functions to be minimized

name	:	expression	ICV	Nadir
f1	:	-2*x1^2-x2^2-48*x1-40*x2		
f2	:	x1^2-x2^2		
f3	:	0.01*(x1-1)^2+(x2-x1^2)^2		
f4	:	x1^2+x2^2+2*x3^2-5*x1-5*x2-21*x3		
f5	:	-x1*x2*x3		

Standard functions available

Variables

☐ Use autofill

		lower bound	<=	starting point	<=	upper bound	Integer values
x1	:	0	<=	0	<=	1000	☐
x2	:	0	<=	0	<=	1000	☑
x3	:	0	<=	0	<=	1000	☑

Linear constraints

Ax <= b

Document: Done

WWW-NIMBUS ## Kalvokuva1 - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

Classify Functions

Please, save the problem if you wish to use it another time. Specify the direction of change from the current criterion vector by selecting an appropriate class for each function. **To use graphical classification instead, select the appropriate operation below.**

Function	Current solution	Class:	<	<=	=	>=	<>
f1	-262.8827		○	○	○	●	○
f2	-2.630336		●	○	○	○	○
f3	11.37786		○	○	○	○	●
f4	1363.751		○	●	○	○	○
f5	-242.2867		○	●	○	○	○

Function	ICV (estim.)	Nadir (estim.)
f1	-262.8827	-7.792588
f2	-16.0	17.0
f3	7.711097E-3	289.0975
f4	-67.25	3587.858
f5	-407.2935	0.0

Next optimization: Global (GA + Deb)

Maximum number of new solutions to be generated: Four

Select operation

Document: Done

WWW-NIMBUS ## Kalvokuva1 - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

NIMBUS TUTORIAL HELP

Enter Classification Parameters

Aspiration functions (<=)

Enter aspiration level(s)

Function	ICV (estim.)	<=	Aspiration level	<=	Current solution
f4	-67.25	<=	500.0	<=	1363.751
f5	-407.2935	<=	-320.0	<=	-242.2867

Upper bound functions (>=)

Enter upper bound values

Function	Current solution	<=	Upper bound	<=	Nadir (estim.)
f1	-262.8827	<=	-150.0	<=	-7.792588

Select operation

- ○ Change the problem
- ○ Save the current problem
- ○ Remove a saved problem
- ● Calculate a new solution (continue)
- ○ Correct ICV or Nadir
- ○ Show the whole problem

Document: Done

WWW-NIMBUS ## Kalvokuva1 - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

Warning: Notice that only 3 new alternatives could be generated.

Analyse Results

Select one alternative for continuation (classification), generate intermediate alternatives between the first and the last alternative, compare the alternatives visually or stop.

☐ **Alternative1**	:	f1 :	-262.8827
		f2 :	-2.630336
		f3 :	11.37786
		f4 :	1363.751
		f5 :	-242.2867
☐ **Alternative2**	:	f1 :	-225.6446
		f2 :	-5.510456
		f3 :	0.2471879
		f4 :	1586.149
		f5 :	-190.5393
☐ **Alternative3**	:	f1 :	-178.8686
		f2 :	-15.99645
		f3 :	15.98043
		f4 :	2496.705
		f5 :	-9.776596
☐ **Alternative4**	:	f1 :	-239.2094
		f2 :	-4.546019
		f3 :	2.126391

Document: Done

WWW-NIMBUS ## Kalvokuva1 - Netscape

Select visualizations

- ☐ Value paths in the absolute range of values
- ☑ Value paths in the relative range of values
- ☐ Bar charts arranged by function (Absolute values)
- ☐ Bar charts arranged by function (Relative values)
- ☐ Bar charts arranged by alternative (Absolute values)
- ☐ Bar charts arranged by alternative (Relative values)
- ☑ Petal diagrams (Relative values)
- ☐ 3D diagrams (Relative values)

Select the alternatives to continue with

- ☑ Alternative1 (Criterion vector: (-262.8827, -2.630336, 11.37786, 1363.751, -242.2867))
- ☑ Alternative2 (Criterion vector: (-225.6446, -5.510456, 0.2471879, 1586.149, -190.5393))
- ☑ Alternative3 (Criterion vector: (-178.8686, -15.99645, 15.98043, 2496.705, -9.776596))
- ☑ Alternative4 (Criterion vector: (-239.2094, -4.546019, 2.126391, 1472.902, -208.9341))

○ Change the problem ○ Return to result analysis

◉ Visualize the alternatives (continue) ○ Show the whole problem

Submit Clear

---

f1 : -2*x1^2-x2^2-48*x1-40*x2
f2 : x1^2-x2^2
f3 : 0.01*(x1-1)^2+(x2-x1^2)^2
f4 : x1^2+x2^2+2*x3^2-5*x1-5*x2-21*x3
f5 : -x1*x2*x3

Send comments and bug reports to nimbus@mit.jyu.fi

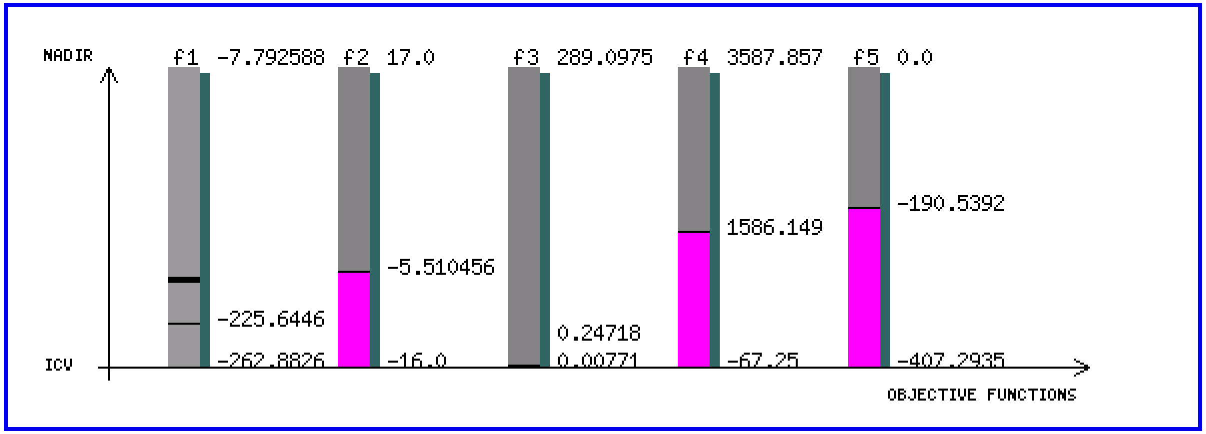
Document: Done

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.**

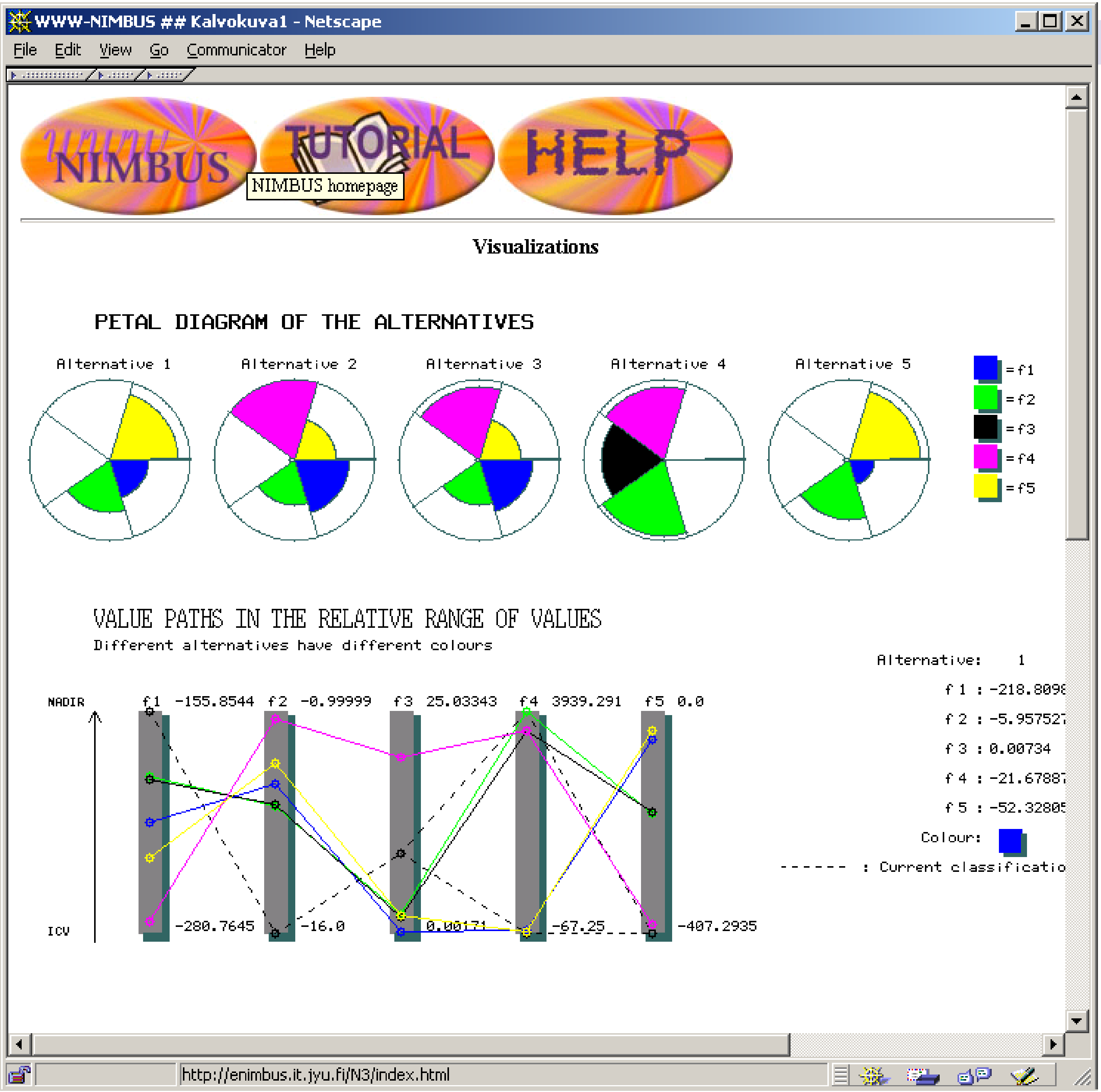


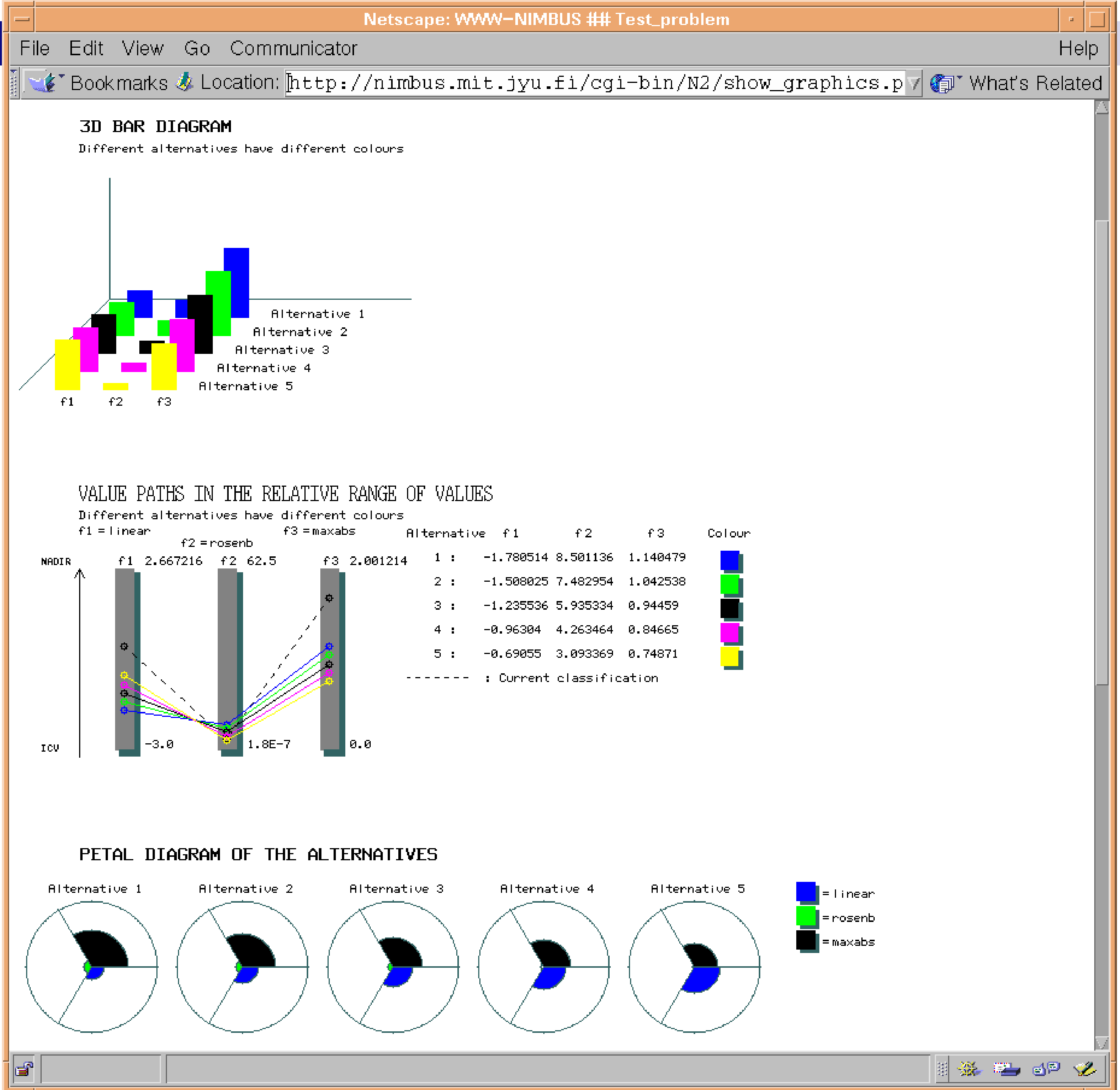
Function	ICV (estim.)	Current Solution	Nadir (estim.)
f1	-262.8827	-225.6446	-7.792588
f2	-16.0	-5.510456	17.0
f3	7.711097E-3	0.2471879	289.0975
f4	-67.25	1586.149	3587.858
f5	-407.2935	-190.5393	0.0

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
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



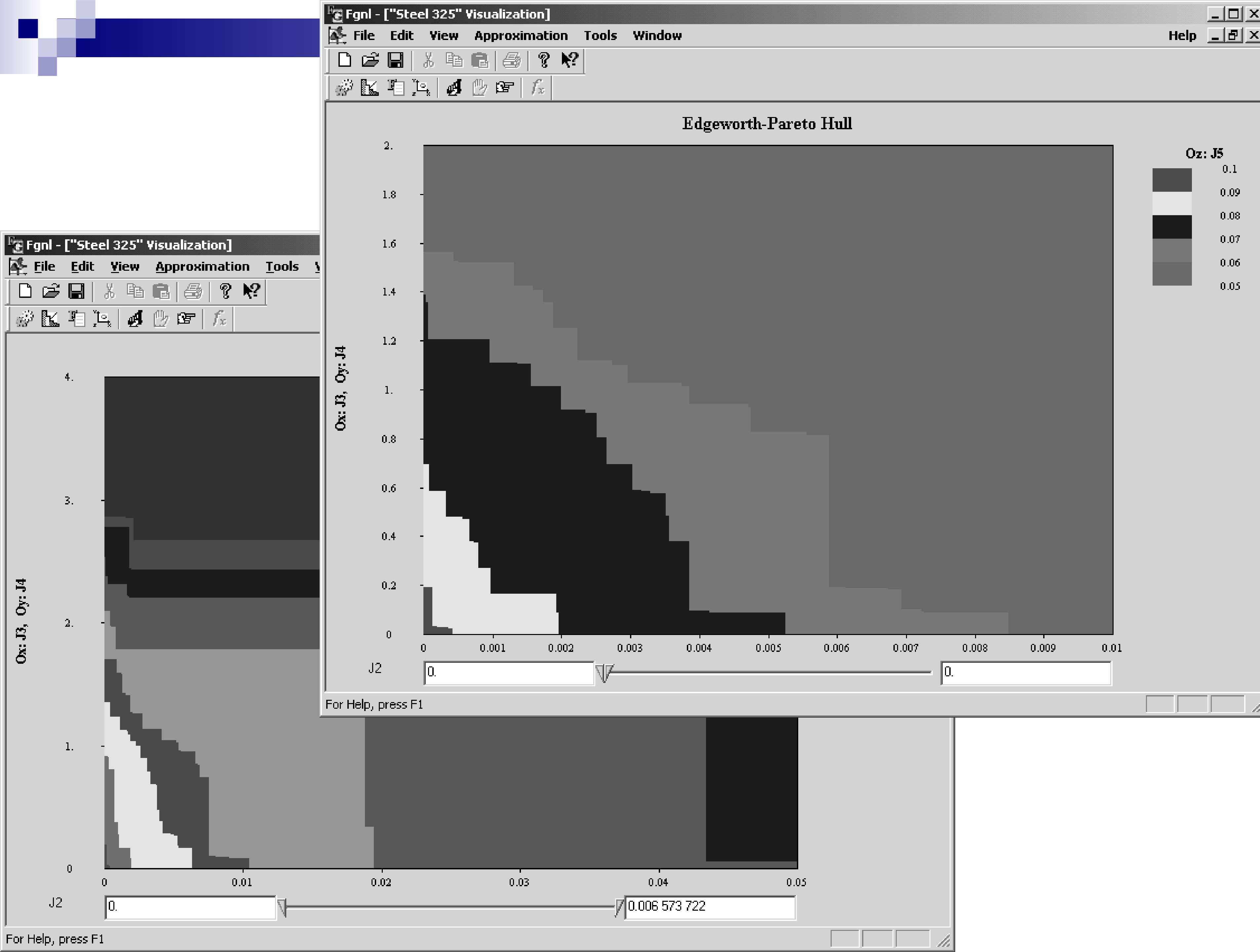
Netscape: WWW-NIMBUS ## Test_problem
File Edit View Go Communicator Help
Bookmarks Location: http://nimbus.mit.jyu.fi/cgi-bin/N2/show_graphics.p
What's Related
3D BAR DIAGRAM
Different alternatives have different colours
Alternative 1
Alternative 2
Alternative 3
Alternative 4
Alternative 5
f1 f2 f3
VALUE PATHS IN THE RELATIVE RANGE OF VALUES
Different alternatives have different colours
f1 = linear
f2 = rosenb
f3 = maxabs
NADIR
f1 2.667216
f2 62.5
f3 2.001214
ICV
-3.0
1.8E-7
0.0
Alternative f1 f2 f3 Colour
1 : -1.780514 8.501136 1.140479
2 : -1.508025 7.482954 1.042538
3 : -1.235536 5.935334 0.94459
4 : -0.96304 4.263464 0.84665
5 : -0.69055 3.093369 0.74871
------- : Current classification
PETAL DIAGRAM OF THE ALTERNATIVES
Alternative 1
Alternative 2
Alternative 3
Alternative 4
Alternative 5
= linear
= rosenb
= maxabs

Summary: NIMBUS &

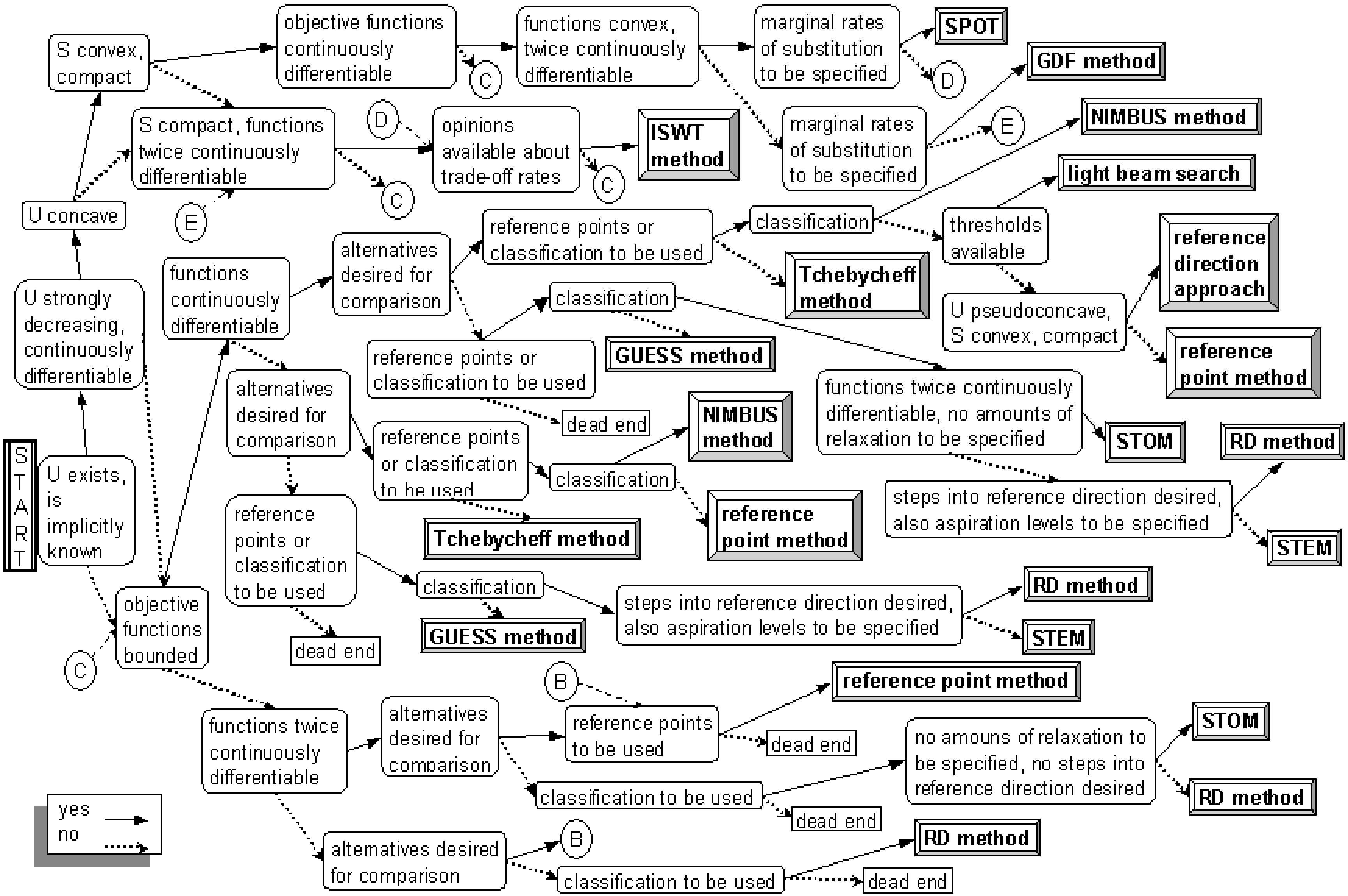
- Interactive, classification-based method for continuous even nondifferentiable problems
  - DM indicates desirable changes; no consistency required
  - No demanding questions posed to the DM
  - DM is assumed to have knowledge about the problem, no deep understanding of the optimization process required
  - 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
  - Flexible and versatile: classification, comparison, extracting undesirable solutions are used in the extent the DM desires

Some Other Methods

- Reference Direction approaches (Korhonen, Laakso, Narula et al)
  - Steps are taken in the direction between reference point and current solution
- Parameter Space Investigation (PSI) method (Statnikov, Matusov)
  - For complicated nonlinear problems
  - Upper and lower bounds required for functions
  - PO set is approximated: generate randomly uniformly distributed points and drop a) those not satisfying bounds specified by the DM b) non-PO ones.
- Feasible Goals Method (FGM) (Lotov et al)
  - Pictures display rough approximations of Z and the PO set. Pictures are projections or slices.
  - Z is approximated e.g. by a system of boxes. It contains only a small part of possible boxes, but approximates Z with a desired degree of accuracy
  - DM identifies a preferred objective vector


Fgnl - ["Steel 325" Visualization]
File Edit View Approximation Tools Window Help
Edgeworth-Pareto Hull
Oz: J5
0.1
0.09
0.08
0.07
0.06
0.05
Ox: J3, Oy: J4
2.
1.8
1.6
1.4
1.2
1.
0.8
0.6
0.4
0.2
0
0
0.001
0.002
0.003
0.004
0.005
0.006
0.007
0.008
0.009
0.01
J2
0.
0.
For Help, press F1
Fgnl - ["Steel 325" Visualization]
File Edit View Approximation Tools
Ox: J3, Oy: J4
4.
3.
2.
1.
0
0
0.01
0.02
0.03
0.04
0.05
J2
0.
0.006 573 722
For Help, press F1

Tree Diagram of Methods



Graphical Illustration

- The DM is often asked to compare several alternatives
- Both discrete and continuous problems
  - Some of interactive methods (GDF, ISWT, Tchebycheff, reference point method, light beam search, NIMBUS)
- Illustration is difficult but important
  - Should be easy to comprehend
  - Important information should not be lost
  - No unintentional information should be included
  - Makes it easier to see essential similarities and differences

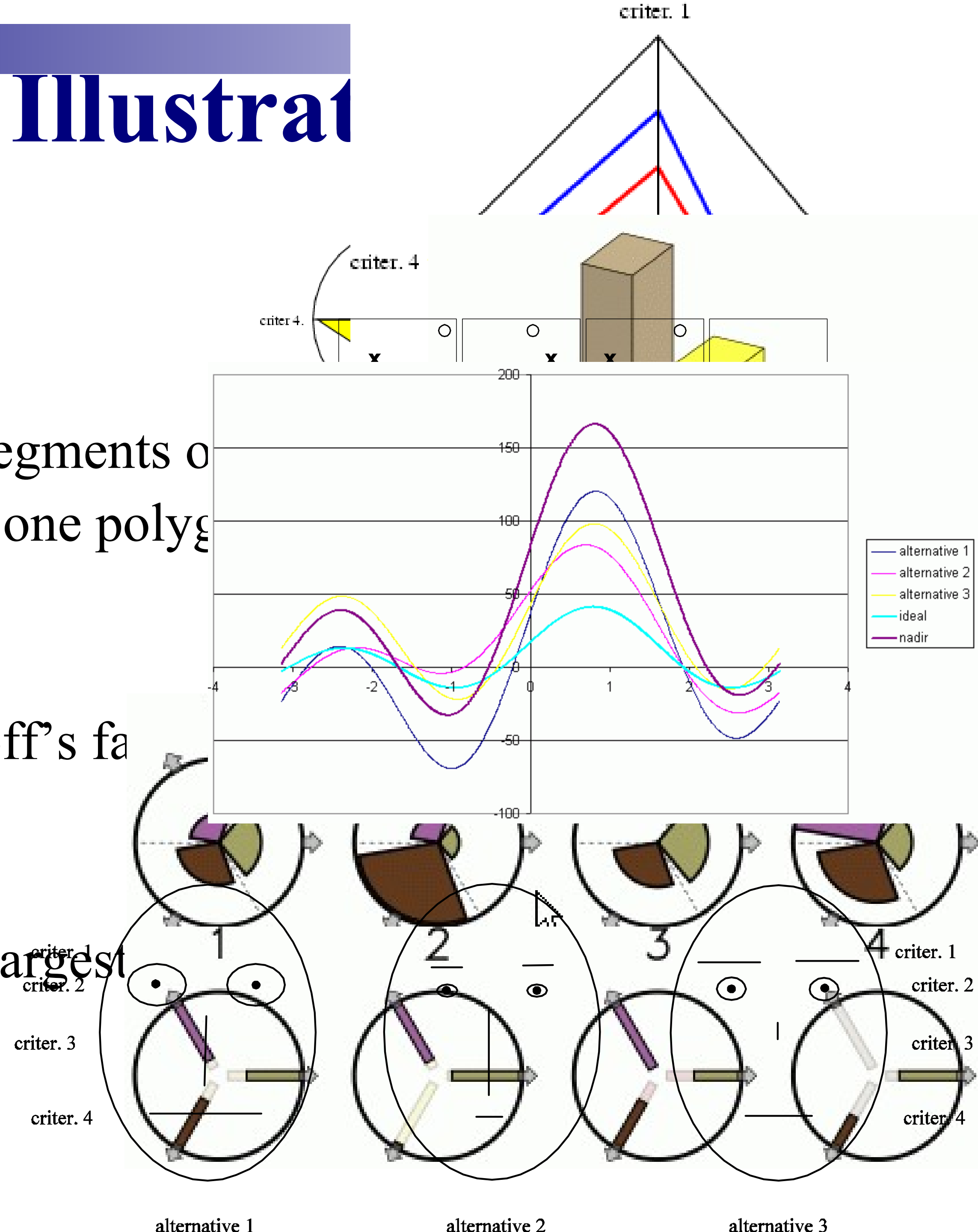
Graphical Illustration cont.

- General-purpose illustration tools are not necessarily applicable
- Surveys of different illustration possibilities are hard to find
- Goal: deeper insight and understanding into the data
- Human limitations (receive, process or remember large amounts of data)
- Magical number
- The more information, the less used $\Rightarrow$ too much information should be avoided

Normalization: (value-ideal)/range

Different Illustrat

- Value path
- Bar chart
- Star presentation (or line segments o
- Spider-web chart (or all in one polyg
- Petal diagram
- Whisker plot
- Iconic approaches (Chernoff's fa
- Fourier series
- Scatterplot matrix
- Projection ideas (e.g. two largest
  projection plane)
- Ordinary tables!!!



Discussion

- Graphs and tables complement each other
- Tables – information acquisition
- Graphs – relationships, viewed at a glance
- Cognitive fit
- Colours – good for association
- New illustrations need time for training
- Let the DM select the most preferred illustrations, select alternatives to be displayed, manipulate order of criteria etc.
- Interaction
  - Hide some pieces of information
  - Highlight
- DMs have different cognitive styles
- Let the DM tailor the graphical display, if possible

Industrial Applications

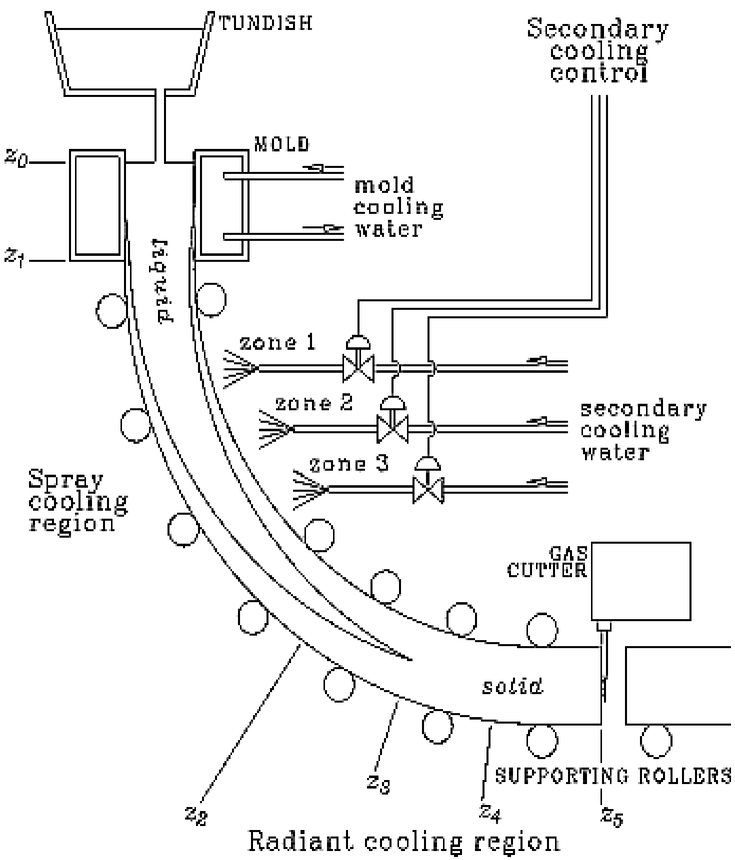
- Continuous casting of steel
- Headbox design for paper machines
- Subprojects of the project

NIMBUS – multiobjective optimization in product development

financed by the National Technology Agency and industrial partners

  - Paper machine design optimizing paper quality (Metso Paper Inc.)
  - Process optimization with chemical process simulation (VTT Processes)
  - Ultrasonic transducer design (Numerola Oy)

Continuous Casting of Steel



- Originally, empty feasible region
- Constraints into objectives
  - 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
- Minimize constraint violations

Paper Machine

- 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

General View of the Wet End

M27004-02

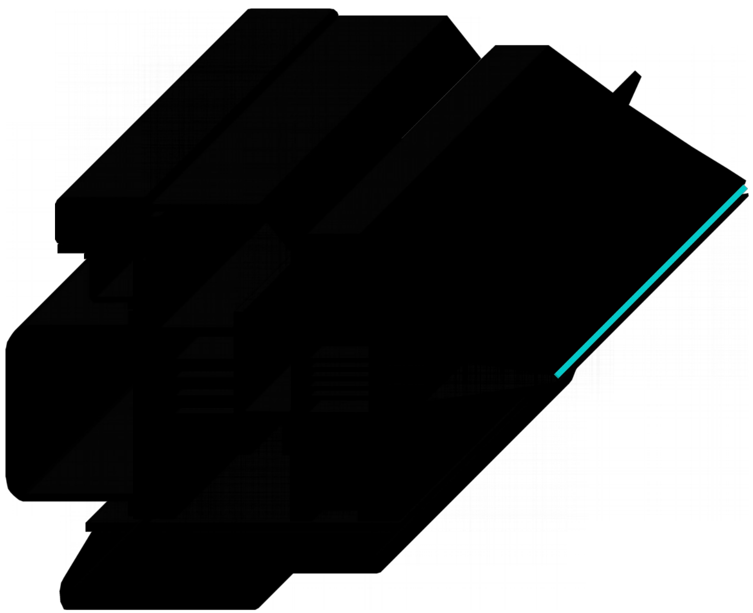
General View of the Dry End

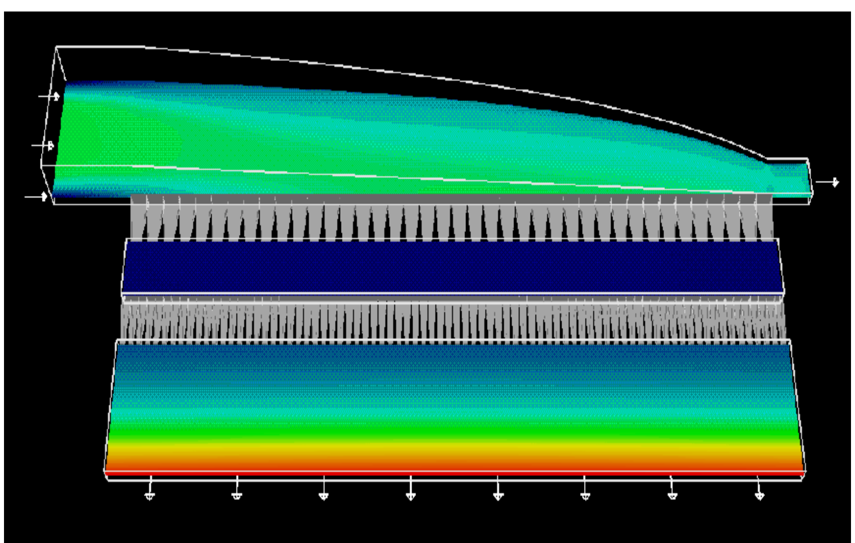
3359-02

Headbox Design

- **Headbox** is located at the wet end
- Distributes furnish (wood fibres, filler clays, chemicals, water) on a moving wire (former) so that outlet jet has controlled
  - concentration, thickness
  - velocity in machine and cross direction
  - turbulence
- Flow properties affect the quality of paper. 3 objective functions
  - basis weight
  - fibre orientation
  - machine direction velocity component
- Headbox outlet height control
- PDE-based models: depth-averaged Navier-Stokes equations for flows with a model for fibre consistency

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

Optimizing Paper Quality

- Consider paper making process and paper machine as a whole
- Paper making process is complex and includes several different phases taken care of by different components of the paper machine
- We have (PDE-based or statistical) submodels for
  - different components
  - different qualitative properties
- We connect submodels to get chains of them to form model-based optimization problems where a simulation model constitutes a *virtual paper machine*
- Dynamic simulation model generation
- Optimal paper machine design is important because, e.g., 1% increase in production means about 1 million euros value of saleable production

Example with 4 Objectives

- Problem related to paper making in four main parts of paper machine: headbox, former, press and drying
- 4 objective functions
  - fiber orientation angle
  - basis weight
  - tensile strength ratio
  - normalized β-formation
  - all of the form: deviations between simulated and goal profiles in the cross-machine direction
- 22 decision variables
  - for example, slice opening, under pressures of rolls and press nip loads
- Simulation model contains 15 submodels
- Interactive solution process with WWW-NIMBUS
  - underlying single objective optimizer: genetic algorithms

Problem Formulation and Solution Process with NIMBUS

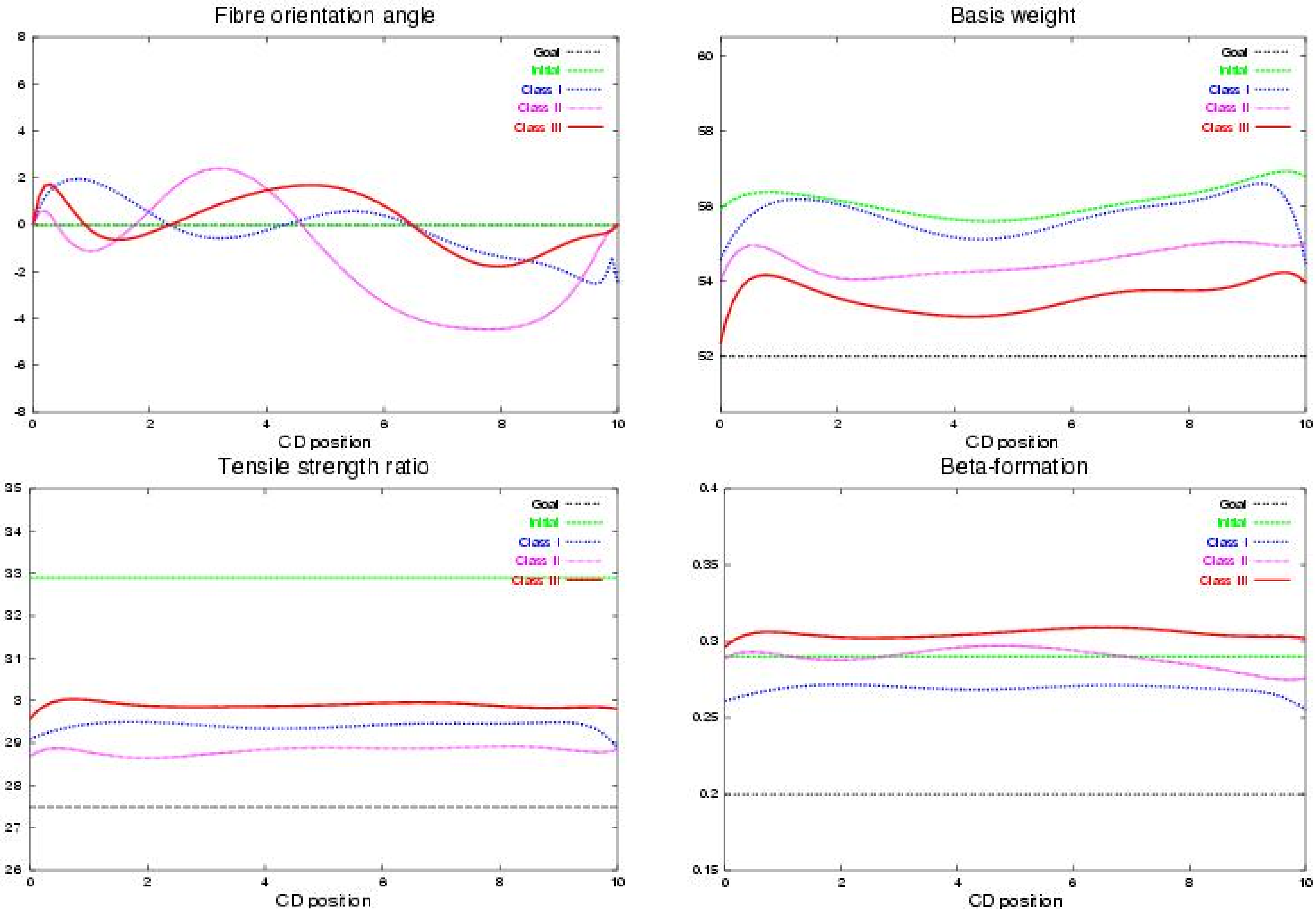
$$
\begin{array}{ll}
\text{minimize} & \{f_1(\mathbf{x}; \mathbf{q}_1, \ldots, \mathbf{q}_{15}), \ldots, \\
& f_4(\mathbf{x}; \mathbf{q}_1, \ldots, \mathbf{q}_{15})\} \\
\text{subject to} & B_1(\mathbf{x}; \mathbf{q}_1) = 0 \\
& B_2(\mathbf{x}; \mathbf{q}_1, \mathbf{q}_2) = 0 \\
& \ldots \\
& B_{15}(\mathbf{x}; \mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_{15}) = 0, \\
& \mathbf{x} \in S,
\end{array}
$$

where

- $\boldsymbol{x}$ is the vector of decision variables
- B_i is the i^{th} submodel in the simulation model, i.e., in the state system
- $\boldsymbol{q}_{\mathrm{i}}$ is the output of B_i, *i.e.,* i^{th} state vector

- Expert DM made 3 classifications and produced intermediate solutions once (between solutions of different scalarizations)

Solution Process cont.



- Black: goal profile, green: initial profile, red: final profile

Example with 5 Objectives

- Problem includes also the finishing part
- 5 objective functions describing qualitative properties of the finished paper
  - min PPS 10-properties (roughness) on top and bottom sides of paper
  - max gloss of paper on top and bottom sides
  - max final moisture
- 22 decision variables
  - typical controls of paper machine including controls in the finishing part of machine
- Simulation model contains 21 submodels
- Interactive solution process with WWW-NIMBUS
  - DM wanted to improve PPS 10-properties and have equal quality on the top and bottom sides of paper
  - underlying single objective optimizer: proximal bundle method

Solution Process with NIMBUS

- 4 classifications and intermediate solutions generated once

Objective function	min/ max	Initial solution	2. class. solution	Interm. solution	3. class. solution	Final solution
PPS 10 top	min	1.20	0.82	0.94	1.24	1.01
PPS 10 bottom	min	1.29	1.03	1.15	1.27	1.04
Gloss top	max	1.09	1.09	1.09	1.05	1.07
Gloss bottom	max	0.99	1.14	1.06	0.95	1.09
Final moisture	max	1.88	0.1	0.89	1.93	1.19

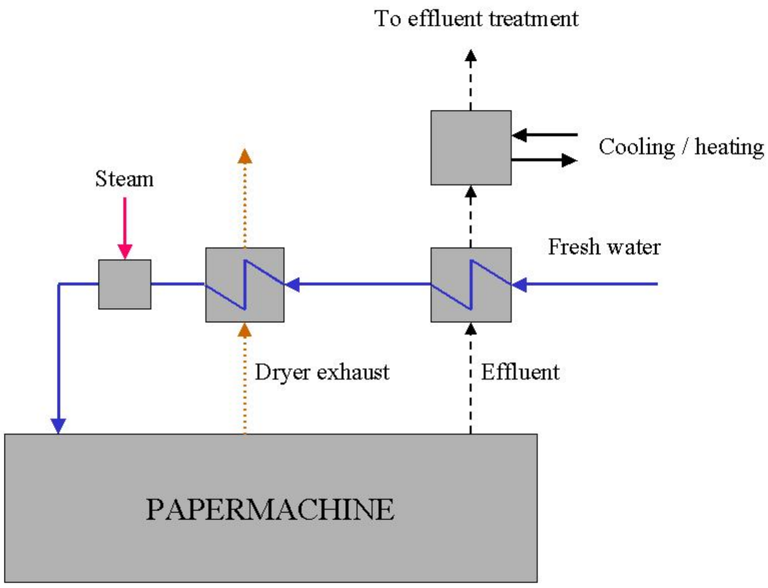
- DM learned about the conflicting qualitative properties
- DM obtained new insight into complex and conflicting phenomena
- DM could consider several objectives simultaneously
- DM found the method easy to use
- DM found a satisfactory solution and was convinced of its goodness

Process Simulation

- Process simulation is widely used in chemical process design
- Optimization problems arising from process simulation (related to chemical processes that can be mathematically modelled)
- Solutions generated must satisfy a mathematical model of a process
- So far, no interactive process design tool has existed that could have handled multiple objectives
- BALAS process simulator (by VTT Processes) is used to provide function values via simulation and combined with WWW-NIMBUS) interactive process optimization

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



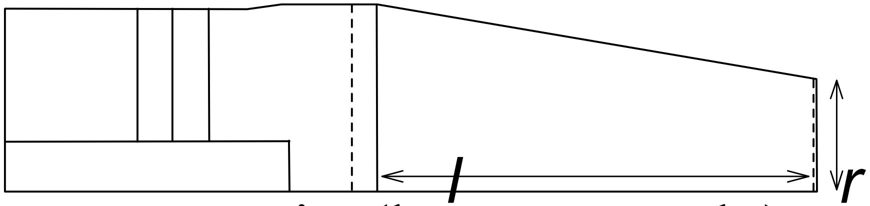
$$\begin{array}{ll}\text{minimize} & \{f_1(\mathbf{y}(\mathbf{x})),\ldots,f_4(\mathbf{y}(\mathbf{x}))\} \\ \text{subject to} & \mathbf{F}(\mathbf{y}(\mathbf{x})) = \mathbf{y}(\mathbf{x}) - \tilde{\mathbf{y}}(\mathbf{y}(\mathbf{x})) = 0 \\ & \mathbf{x} \in S,\end{array}$$

Ultrasonic Transducer

- Optimal shape design problem to find good dimensions (shape) for a cylinder-shaped ultrasonic transducer
- Sound is generated with Langevin-type piezo-ceramic piled elements
- Besides piezo elements, transducer package contains head mass of steel (front), tail mass of aluminium (back) and screw located in the middle axis in the back of the transducer
- Vibrations of the structure are modelled with PDEs
- Simulation model: so-called axisymmetric piezo-equation, i.e., a PDE describing displacements of materials, electric field in the piezo-material and interrelationships
- Axisymmetric structure) geometry as a two-dimensional cross-section (a half of it). Separate density, Poisson ratio, modulus of elasticity and relative permittivity for each type of material

Transducer cont.

- 3 objectives
  - maximal sound output (i.e. vibration of tip)
  - minimal vibration (of fixing part) casing
  - minimal electric impedance
- 2 variables: length of the head mass *l* and radius of tip *r*



- Combine Numerrin (by Numerola), a FEM-simulation software package with WWW-NIMBUS to be able to handle objective functions defined by PDE-based simulation models (with automatic differentiation)

Conclusions

- Multiobjective optimization problems can be solved!
- Multiobjective optimization gives new insight into problems with conflicting criteria
- No extra simplification is needed (e.g., in modelling)
- A large variety of methods; none of them is superior
- Selecting a method = a problem with multiple criteria. Pay attention to features of the problem, opinions of the DM, practical applicability
- Interactive approach good if DM can participate
- Important: user-friendliness
- Methods should support learning
- (Sometimes special methods for special problems)

International Society on Multiple Criteria Decision Making

- More than 1400 members from about 90 countries
- No membership fees at the moment
- Newsletter once a year
- International Conferences organized every two years
- http://www.terry.uga.edu/mcdm/
- Contact me if you wish to join

Further Links

- Suomen Operaatiotutkimusseura ry http://www.optimointi.fi
- Collection of links related to optimization, operations research, software, journals, conferences etc. http://www.mit.jyu.fi/miettine/lista.html