Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

University of Algarve

A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers

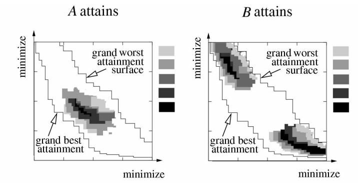
Carlos M. Fonseca
Joshua D. Knowles (speaker)
Lothar Thiele
Eckart Zitzler

Aims

Main goal: how to evaluate/compare the approximation sets from multiple runs of two or more stochastic multiobjective optimizers

We recommend two complementary approaches:

Empirical attainment function



- Applies statistical tests directly to the samples of approximation sets
- Gives detailed information about how and where performance differences occur

Dominance-compliant quality indicators

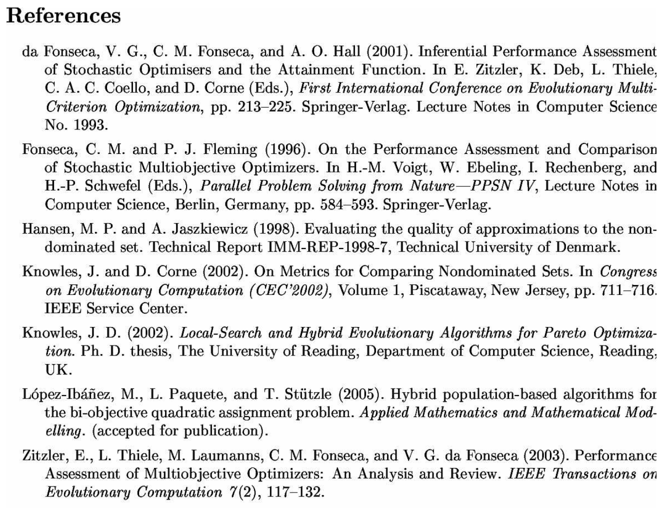
Indicator	A	B
Hypervolume indicator	6.3431	7.1924
ϵ-indicator	1.2090	0.12722
R_2 indicator	0.2434	0.1643
R_3 indicator	0.6454	0.3475

- First, reduces each approximation set to a single value of quality
- Applies statistical tests to the samples of quality values

Scope

- Do provide:
  - Description of the empirical attainment function
  - Recommendations for quality indicators to use
  - Software for indicators and empirical attainment function approaches
  - Statistical testing procedures/software for both EAFs and indicators
  - Case study giving a worked example, using the software provided
- Do not consider:
  - Number of alternative solutions found in decision space
  - Time or computational cost of optimizer
  - Test function choice
  - Scalability of optimizers to number of objectives / decision variables
  - …And many other issues!

Source Materials



References

da Fonseca, V. G., C. M. Fonseca, and A. O. Hall (2001). Inferential Performance Assessment of Stochastic Optimisers and the Attainment Function. In E. Zitzler, K. Deb, L. Thiele, C. A. C. Coello, and D. Corne (Eds.), *First International Conference on Evolutionary Multi-Criterion Optimization*, pp. 213–225. Springer-Verlag. Lecture Notes in Computer Science No. 1993.

Fonseca, C. M. and P. J. Fleming (1996). On the Performance Assessment and Comparison of Stochastic Multiobjective Optimizers. In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature—PPSN IV*, Lecture Notes in Computer Science, Berlin, Germany, pp. 584–593. Springer-Verlag.

Hansen, M. P. and A. Jaszkiewicz (1998). Evaluating the quality of approximations to the non-dominated set. Technical Report IMM-REP-1998-7, Technical University of Denmark.

Knowles, J. and D. Corne (2002). On Metrics for Comparing Nondominated Sets. In *Congress on Evolutionary Computation (CEC'2002)*, Volume 1, Piscataway, New Jersey, pp. 711–716. IEEE Service Center.

Knowles, J. D. (2002). *Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization.* Ph. D. thesis, The University of Reading, Department of Computer Science, Reading, UK.

López-Ibáñez, M., L. Paquete, and T. Stützle (2005). Hybrid population-based algorithms for the bi-objective quadratic assignment problem. *Applied Mathematics and Mathematical Modelling*. (accepted for publication).

Zitzler, E., L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca (2003). Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation* *7*(2), 117–132.

Overview

> “And what is good, Phaedrus, and what is not good? Need we as k any one to tell us these things?” – **Socrates**

- Part 1 - Introduction
  - Aims and scope
  - Definitions and basics of Pareto dominance
  - The limitations of dominance relations
- Part 2 – Methods
  - Special case: nondominated sorting of approximation sets
  - 1st approach – empirical attainment functions
  - 2nd approach – (dominance-compatible) quality indicators
- Part 3 – In Practice
  - Software guide – (PISA framework)
  - Case study

Definitions

We assume that the optimization problem under consideration involves n objective functions $f_1, \ldots, f_n$ that are all to be minimized.

Each objective function $f_i : X \Rightarrow \mathbb{R}$ assigns every potential solution x in the search space X a corresponding real value $z_i = f_i(x)$ that reflects its usefulness according to the ith criterion: the smaller the better.

In this sense, every $x \in X$ is mapped to a corresponding vector $\boldsymbol{z} = (z_1, \ldots, z_n) \in Z$ of objective values, with $\boldsymbol{z} = \boldsymbol{f}((f_1(x), \ldots, f_n(x)) \in \mathbb{R}^n$.

Definition 1 (Approximation set) *Let $A \subseteq Z$ be a set of objective vectors. A is called an approximation set if any element of A does not weakly dominate any other objective vector in A. The set of all approximation sets is denoted as Ω.*

Dominance Relations on Objective Vectors

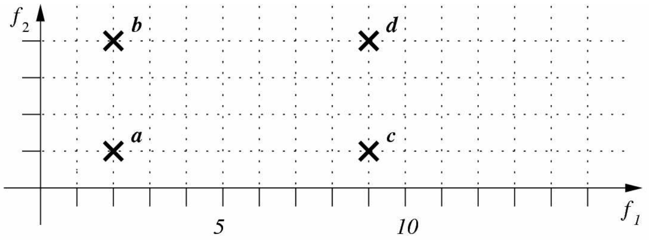


Figure : Examples of dominance relations on objective vectors. Assuming that two objectives are to be minimized, it holds that $\boldsymbol{a} \succ \boldsymbol{b}$, $\boldsymbol{a} \succ \boldsymbol{c}$, $\boldsymbol{a} \succ \boldsymbol{d}$, $\boldsymbol{b} \succ \boldsymbol{d}$, $\boldsymbol{c} \succ \boldsymbol{d}$, $\boldsymbol{a} \succ\succ \boldsymbol{d}$, $\boldsymbol{a} \succeq \boldsymbol{a}$, $\boldsymbol{a} \succeq \boldsymbol{b}$, $\boldsymbol{a} \succeq \boldsymbol{c}$, $\boldsymbol{a} \succeq \boldsymbol{d}$, $\boldsymbol{b} \succeq \boldsymbol{b}$, $\boldsymbol{b} \succeq \boldsymbol{d}$, $\boldsymbol{c} \succeq \boldsymbol{c}$, $\boldsymbol{c} \succeq \boldsymbol{d}$, $\boldsymbol{d} \succeq \boldsymbol{d}$, and $\boldsymbol{b} \parallel \boldsymbol{c}$.

Dominance Relations on Objective Vectors and Approximation Sets

Table : Relations on objective vectors and approximation sets. The relations $\prec$, $\prec\prec$, $\lhd$, and $\preceq$ are defined accordingly, e.g., $\boldsymbol{z}^1 \prec \boldsymbol{z}^2$ is equivalent to $\boldsymbol{z}^2 \succ \boldsymbol{z}^1$ and $A \lhd B$ is defined as $B \rhd A$.

relation	objective vectors		approximation sets	
strictly dominates	$\boldsymbol{z}^1 \succ\succ \boldsymbol{z}^2$	$\boldsymbol{z}^1$ is better than $\boldsymbol{z}^2$ in all objectives	$A \succ\succ B$	every $\boldsymbol{z}^2 \in B$ is strictly dominated by at least one $\boldsymbol{z}^1 \in A$
dominates	$\boldsymbol{z}^1 \succ \boldsymbol{z}^2$	$\boldsymbol{z}^1$ is not worse than $\boldsymbol{z}^2$ in all objectives and better in at least one objective	$A \succ B$	every $\boldsymbol{z}^2 \in B$ is dominated by at least one $\boldsymbol{z}^1 \in A$
better			$A \rhd B$	every $\boldsymbol{z}^2 \in B$ is weakly dominated by at least one $\boldsymbol{z}^1 \in A$ and $A \neq B$
weakly dominates	$\boldsymbol{z}^1 \succeq \boldsymbol{z}^2$	$\boldsymbol{z}^1$ is not worse than $\boldsymbol{z}^2$ in all objectives	$A \succeq B$	every $\boldsymbol{z}^2 \in B$ is weakly dominated by at least one $\boldsymbol{z}^1 \in A$
incomparable	$\boldsymbol{z}^1 \parallel \boldsymbol{z}^2$	neither $\boldsymbol{z}^1$ weakly dominates $\boldsymbol{z}^2$ nor $\boldsymbol{z}^2$ weakly dominates $\boldsymbol{z}^1$	$A \parallel B$	neither A weakly dominates B nor B weakly dominates A

Dominance Relations on Approximation Sets

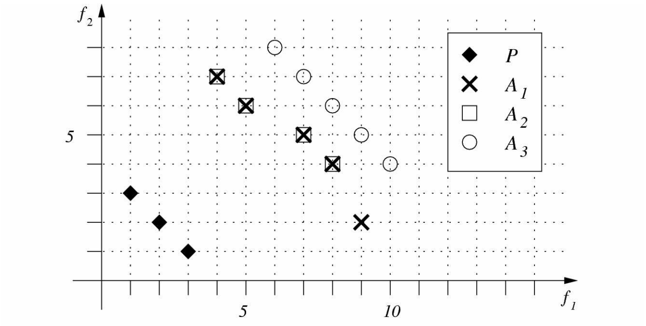


Figure : Outcomes of three hypothetical algorithms for a two-dimensional minimization problem. The corresponding approximation sets are denoted as A_1, A_2, and A_3; the Pareto-optimal set P comprises three objective vectors. Between A_1, A_2, and A_3, the following dominance relations hold: $A_1 \succ A_3$, $A_2 \succ A_3$, $A_1 \succ\succ A_3$, $A_1 \succeq A_1$, $A_1 \succeq A_2$, $A_1 \succeq A_3$, $A_2 \succeq A_2$, $A_2 \succeq A_3$, $A_3 \succeq A_3$, $A_1 \triangleright A_2$, $A_1 \triangleright A_3$, and $A_2 \triangleright A_3$.

Overlapping Subsets Induced by Dominance Relations (Zitzler et al, 2003)

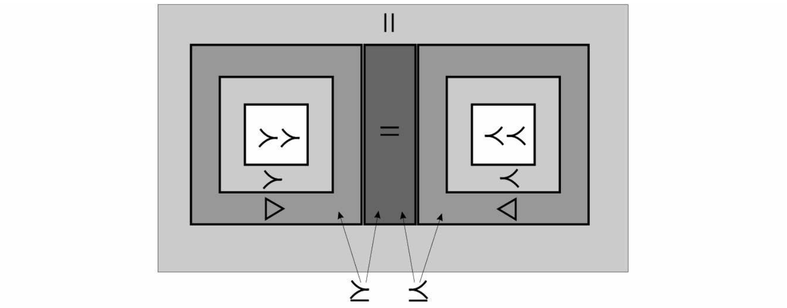
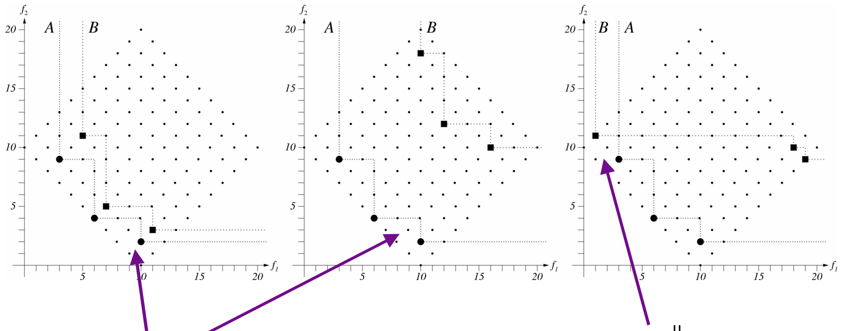


Figure : Partitioning of the set of ordered pairs $(A, B) \in \Omega^2$ of approximation sets into (overlapping) subsets induced by the different dominance relations; each subset labeled with a certain relation $\blacktriangleright$ contains those pairs (A, B) for which $A \blacktriangleright B$. Note that the set of all pairs (A, B) with $A \succeq B$ is the union of those with $A = B$ and $A \triangleright B$.

Limitations of Dominance Relations on Approximation Sets



$A \| B$

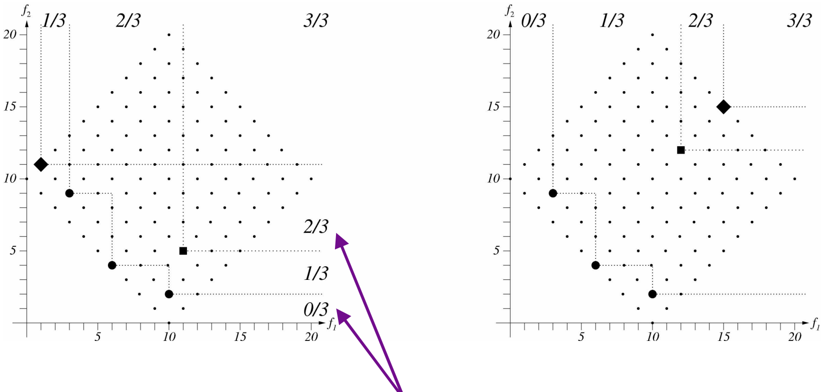
Both $A \succ\succ B$

But on right, A is ***much*** better

But A better to “most decision-makers in most situations”.

The Stochastic Element of Performance

Three runs of two stochastic optimizers



Frequency of attaining regions

Summary of Part 1

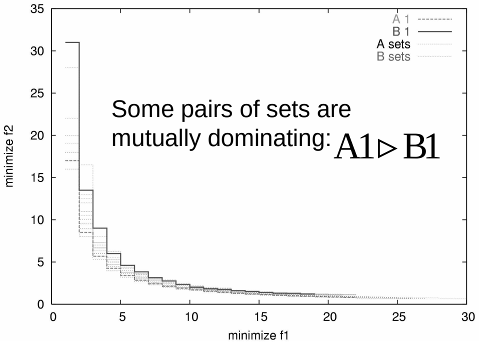
- Pareto dominance relations extend from objective vectors to approximation sets
- Dominance is scaling (linear or non-linear) independent, enabling comparison of vectors/sets even when objectives are non-commensurable
- Dominance does not use or account for preference information
  - It cannot detect degrees of “better” (it is just a binary relation)
  - It cannot detect differences between non-comparable sets
- Algorithms are stochastic – different sets are attained with different frequencies. How can this be measured?

Part 2 - Methods

- Using dominance relations only to compare stochastic optimizers – a special case
- Empirical attainment functions – describing the frequency distribution of attained regions
- Quality indicators – reducing the dimension of approximation sets but still respecting dominance
  - Principles of quality indicators, and interpretation functions
  - Non-Pareto-compatible indicators
  - Recommended indicators
  - Combinations of indicators
  - Reference points and reference sets
  - Statistical methods for inferences from multiple runs
  - Multiple testing issues

A Special Case: Nondominated Sorting of Approximation Sets

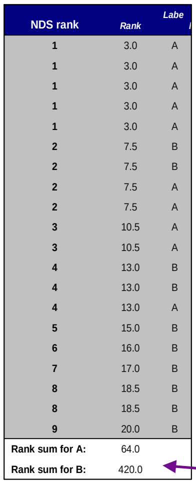
Two optimizers on ZDT5



Some pairs of sets are mutually dominating: $A1 \triangleright B1$

- When comparing optimizers, often performance differences are small
- Then, many pairs of sets will be incomparable
- However, in some special cases (due to the objective function or the optimizers), pairs of sets tend do dominate each other
- If there are many pairs dominating each other, then nondominated sorting *of the sets* can be applied

Nondominated Sorting of Approximation Sets – Rank Test



NDS rank	*Rank*	*Labe*
1	3.0	A
1	3.0	A
1	3.0	A
1	3.0	A
1	3.0	A
2	7.5	B
2	7.5	B
2	7.5	A
2	7.5	A
3	10.5	A
3	10.5	A
4	13.0	B
4	13.0	B
4	13.0	A
5	15.0	B
6	16.0	B
7	17.0	B
8	18.5	B
8	18.5	B
9	20.0	B
Rank sum for A:	64.0	
Rank sum for B:	420.0	

Step 1: Apply nondominated sorting to approximation sets i.e. “peel off” all sets that are nondominated and rank them 1, repeat with the remainder and give them rank 2, and so on.

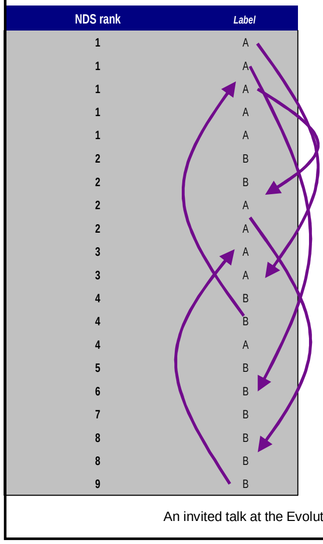
Step 2: Compute ranks, giving ties the same rank

Step 3:Compute the test statistic – the sum of ranks for each algorithm

Step 4: Test the significance of the sum of ranks, e.g. using the Mann-Whitney U tables.

Differences in the sum of ranks significant at $\alpha = 0.05$

Nondominated Sorting of Approximation Sets – Fisher's Test



NDS rank	Label
1	A
1	A
1	A
1	A
1	A
2	B
2	B
2	A
2	A
3	A
3	A
4	B
4	B
4	A
5	B
6	B
7	B
8	B
8	B
9	B

Step 1: Apply nondominated sorting to approximation sets, i.e. "peel off" all sets that are nondominated and rank them 1, repeat with the remainder and give them rank 2, and so on.

Step 2: Compute the test statistic t_0 = the sum of nondominated ranks for A

Step 3: Permute the labels randomly and recompute the test statistic t. Repeat 10,000 times.

Step 4: Estimate the probability that A and B are from the same distribution as

$$P(T > t_0) = \frac{\left|\{t \mid t > t_0\}\right|}{10,000}$$

Method 1 – Attainment Functions

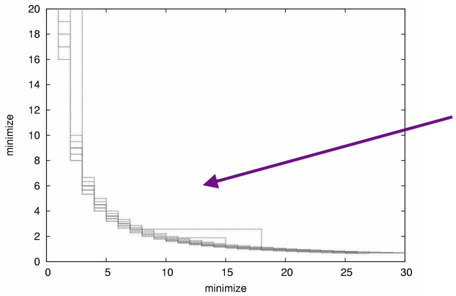
(Fonseca and Fleming, 1996; and Grunert da Fonseca et al, 2001)

Stochastic Optimizers

The solution quality outcomes of stochastic optimizers in the multiobjective case are represented by random point sets

$$\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_M\}$$

where the $Y_j \in \mathbb{R}^n$ are random, nondominated objective vectors, and M is random.



Random point sets

Attainment Functions

The attainment function of $\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_M\}$ is defined by the following

$$\boxed{\begin{aligned} \alpha_{\mathcal{Y}}(z) &= P(Y_1 \le z \vee Y_2 \le z \vee \ldots \vee Y_M \le z) \\ &= P(\mathcal{Y} \trianglerighteq z) \\ &= P(\text{optimizer attains the goal } z \text{ in a single run}). \end{aligned}}$$

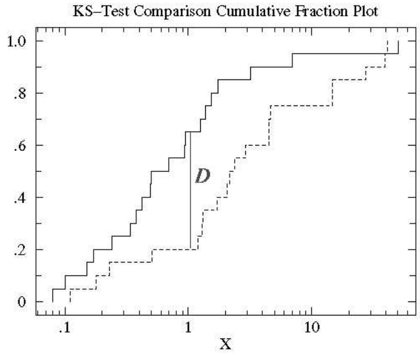
This can be estimated via the empirical attainment function

$$\alpha_r(z) = \frac{1}{r} \sum_{i=1}^{r} \boldsymbol{I}\{\mathcal{Y}_i \trianglerighteq z\},$$

where r is the number of runs and $\boldsymbol{I}$ is the indicator function.

Global Statistical Inference from KS-like Tests

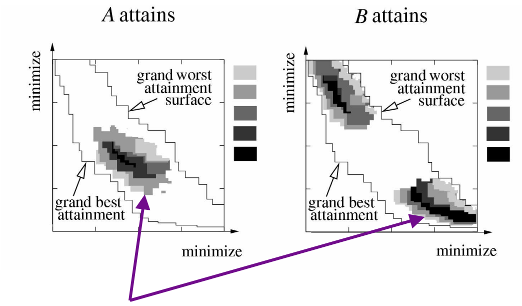
- A Kolmogorov-Smirnov test examines the maximum difference between two cumulative distribution functions
- A KS-like test can be used to probe differences between the empirical attainment functions of a pair of optimizers, *A* and *B*
- The null hypothesis is that the attainment functions of *A* and *B* are identical
- The alternative hypothesis is that the distributions differ somewhere



Investigating Differences in the Distributions of Two Sets

- If the null hypothesis of the KS-test is rejected, it is possible to investigate exactly what differences there are in the two attainment functions
- The regions where the largest differences in the frequency with which they are obtained can be found
- In 2*d* (i.e. two objectives), these differences can be shown clearly with pairs of plots (See next slide)
- For three or more objectives, differences can best be seen using interactive, graphical software

Visualizing Differences in the Empirical Attainment Functions (Lopez-Ibanez 2005)

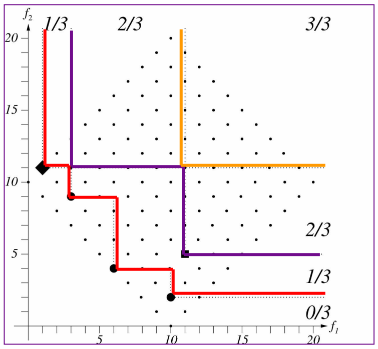


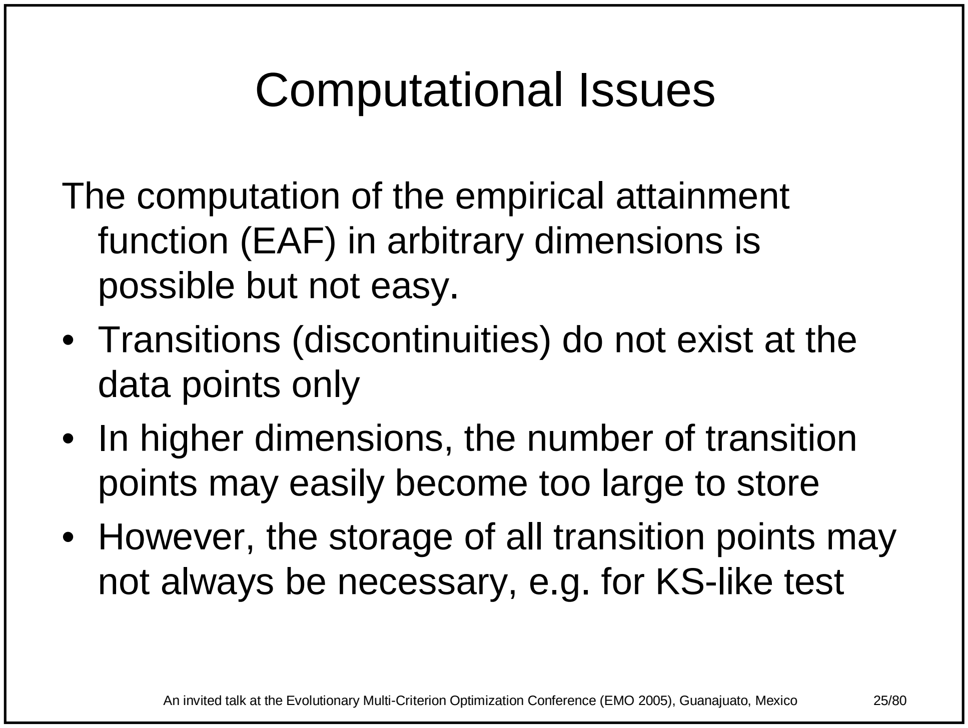
These indicate either the difference in frequencies of attaining the goal or the *p*-value of the difference, coded as a shaded level

- Concatenate all runs from *A* and *B* and compute the grand best and grand worst attainment surfaces
- Compute all goals where there is a statistically significant difference in probability of attaining that region between algorithm *A* and *B*
- Either: plot the *difference in* empirical frequency of attaining those goal where *A* is better in the left plot (and where *B* is better in the right plot)
- Or: plot the *p*-values where *A* is better in left plot (respectively *B* is better in right plot)

Attainment Surface Plots

- Three runs of an optimizer are shown
- Can show the surface that bounds the region obtained in 1/3 of the runs
- Can do the same for the 2/3 and 3/3 "attainment surfaces"
- In general, we can plot the surface obtained in 50% of runs

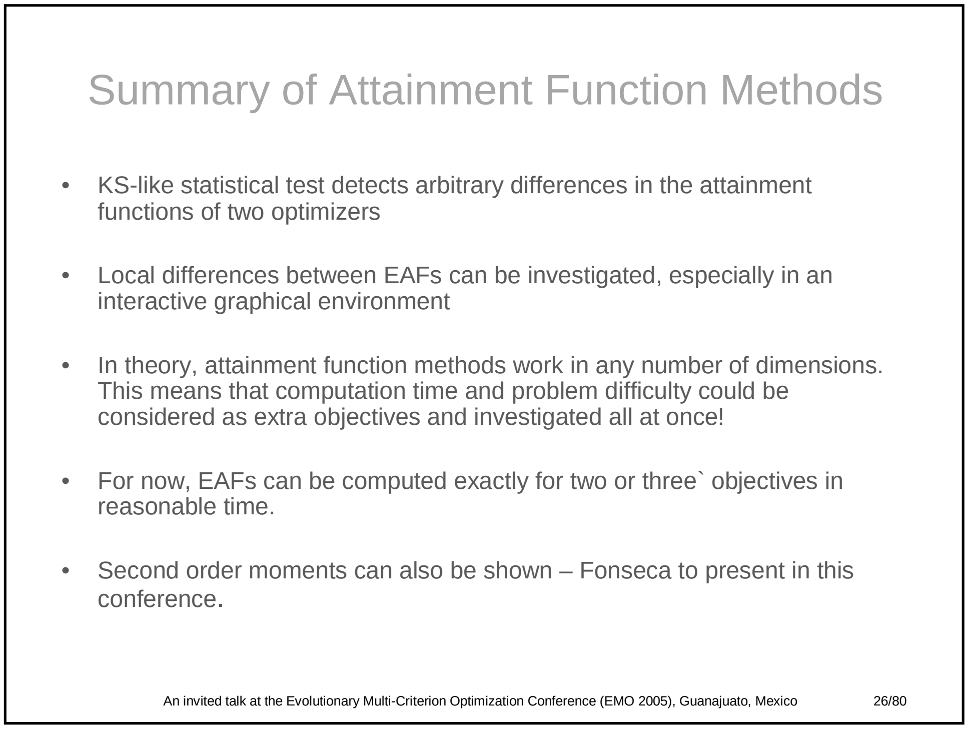




Computational Issues

The computation of the empirical attainment function (EAF) in arbitrary dimensions is possible but not easy.

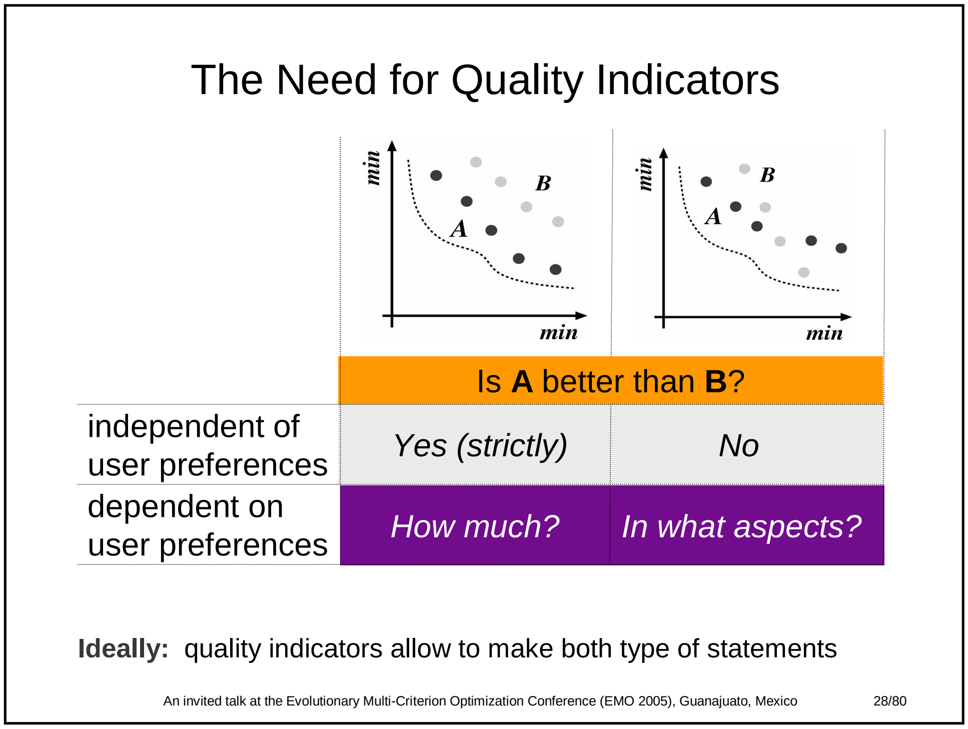
- Transitions (discontinuities) do not exist at the data points only
- In higher dimensions, the number of transition points may easily become too large to store
- However, the storage of all transition points may not always be necessary, e.g. for KS-like test



Summary of Attainment Function Methods

- KS-like statistical test detects arbitrary differences in the attainment functions of two optimizers
- Local differences between EAFs can be investigated, especially in an interactive graphical environment
- In theory, attainment function methods work in any number of dimensions. This means that computation time and problem difficulty could be considered as extra objectives and investigated all at once!
- For now, EAFs can be computed exactly for two or three` objectives in reasonable time.
- Second order moments can also be shown – Fonseca to present in this conference.

Method 2 – Quality Indicators

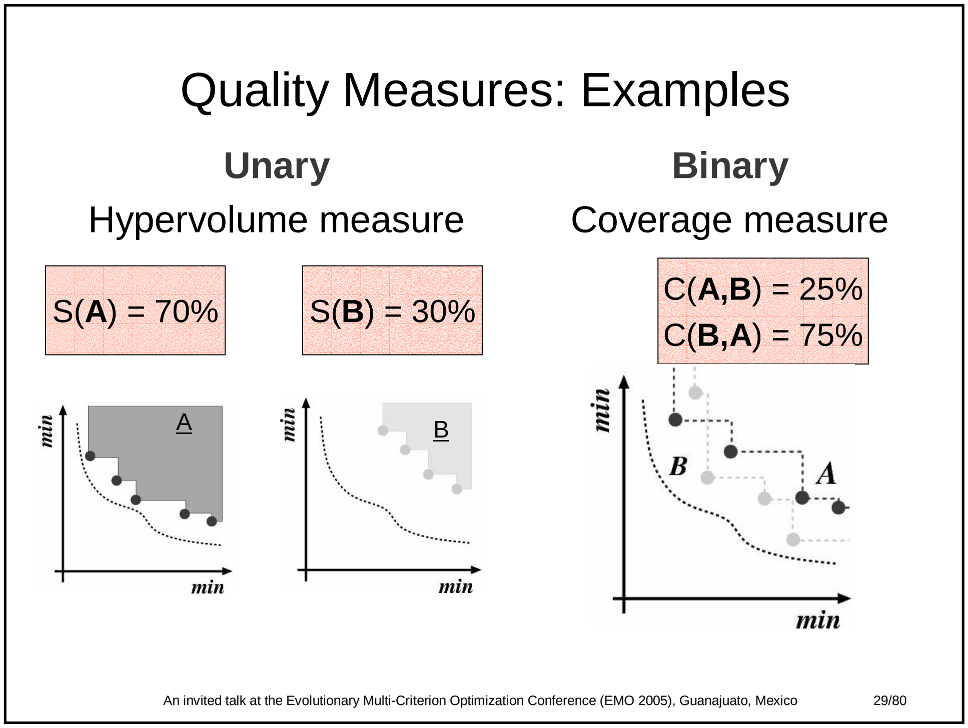


The Need for Quality Indicators

min
B
A
min
min
B
A
min

	Is **A** better than **B**?	
independent of user preferences	*Yes (strictly)*	*No*
dependent on user preferences	*How much?*	*In what aspects?*

Ideally: quality indicators allow to make both type of statements



Quality Measures: Examples

Unary

Hypervolume measure

S(**A**) = 70%

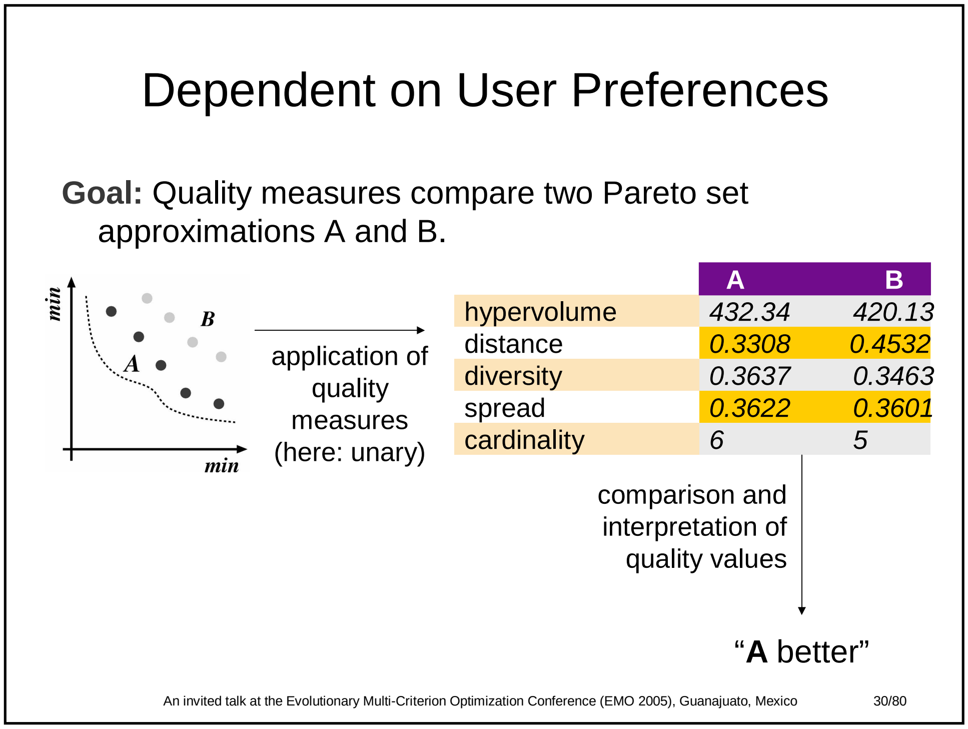
S(**B**) = 30%

Binary

Coverage measure

C(**A,B**) = 25%

C(**B,A**) = 75%



Dependent on User Preferences

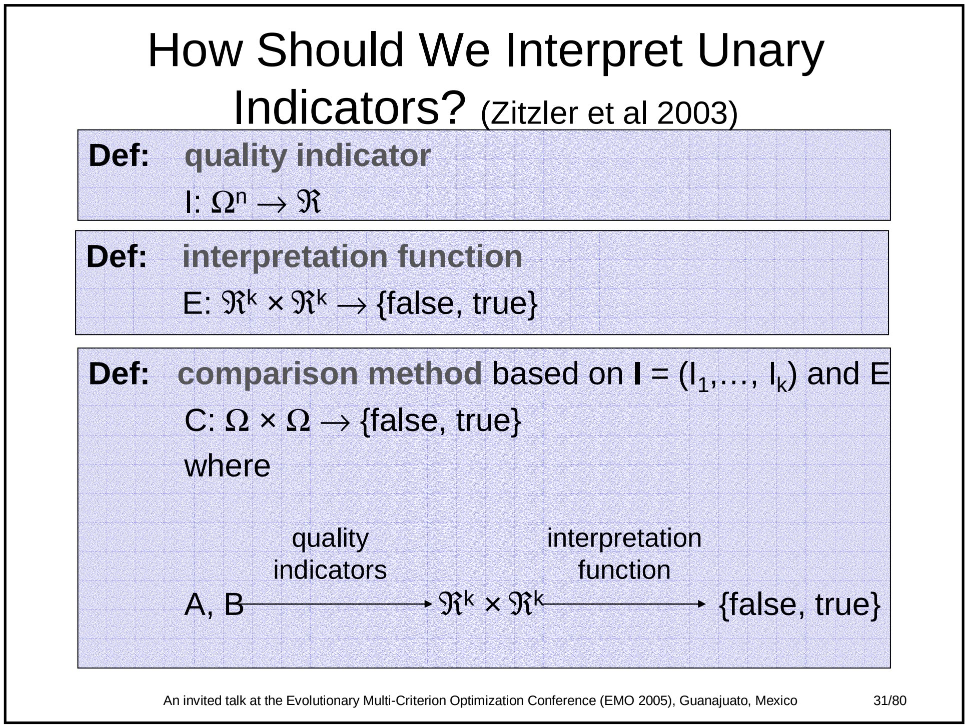
Goal: Quality measures compare two Pareto set approximations A and B.

application of quality measures (here: unary)

	A	B
hypervolume	*432.34*	*420.13*
distance	*0.3308*	*0.4532*
diversity	*0.3637*	*0.3463*
spread	*0.3622*	*0.3601*
cardinality	*6*	*5*

comparison and interpretation of quality values

“**A** better”



How Should We Interpret Unary Indicators? (Zitzler et al 2003)

Def: **quality indicator**

$I: \Omega^n \rightarrow \Re$

Def: **interpretation function**

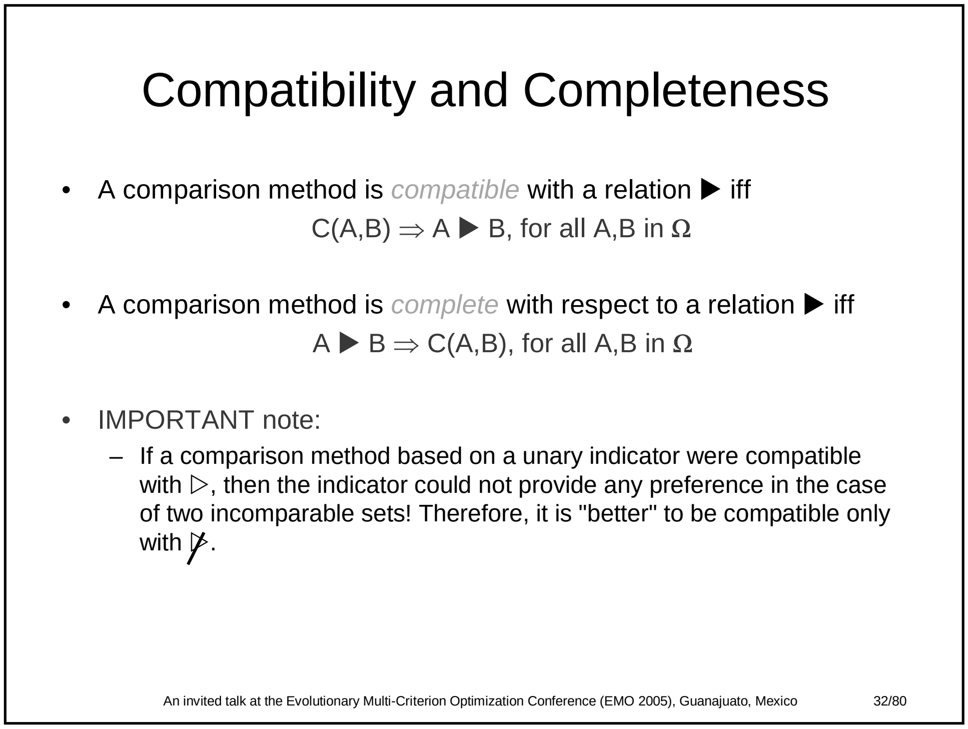
$E: \Re^k \times \Re^k \rightarrow \{false, true\}$

Def: **comparison method** based on $\mathbf{I} = (I_1, \ldots, I_k)$ and E

$C: \Omega \times \Omega \rightarrow \{false, true\}$

where

$$\begin{tikzcd} A, B \arrow[r, "\text{quality indicators}"] & \Re^k \times \Re^k \arrow[r, "\text{interpretation function}"] & \{false, true\} \end{tikzcd}$$



Compatibility and Completeness

- A comparison method is *compatible* with a relation ▶ iff

  $C(A,B) \Rightarrow A \blacktriangleright B$, for all A,B in Ω

- A comparison method is *complete* with respect to a relation ▶ iff

  $A \blacktriangleright B \Rightarrow C(A,B)$, for all A,B in Ω

- IMPORTANT note:
  - If a comparison method based on a unary indicator were compatible with $\triangleright$, then the indicator could not provide any preference in the case of two incomparable sets! Therefore, it is "better" to be compatible only with $\ntriangleright$.

Example of a Dominance-Compliant Comparison Method

The hypervolume indicator can be used as the basis of a dominance-compliant comparison method as follows:

$$C_{I_H}(A,B) \equiv E(I_H(A) > I_H(B))$$

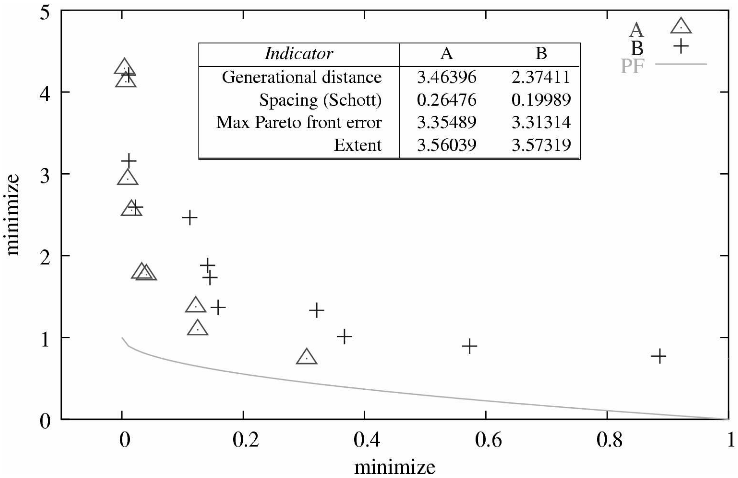
$C_{I_H}(A,B)$ is compatible with $\ntriangleright$ so: $C_{I_H}(A,B) \Rightarrow B \ntriangleright A$

That is, if the comparison yields true, then we know THAT *B* is not better than *A*

$C_{I_H}(A,B)$ Is also complete with respect to $\triangleright$ so $A \triangleright B \Rightarrow C_{I_H}(A,B)$

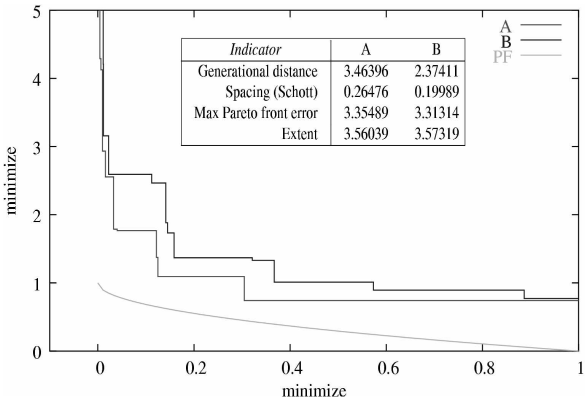
Therefore, all cases where *A* is better than *B* are detected by the indicator

Troubles with “Functionally Independent” Indicators



Indicator	A	B
Generational distance	3.46396	2.37411
Spacing (Schott)	0.26476	0.19989
Max Pareto front error	3.35489	3.31314
Extent	3.56039	3.57319

An Experiment – Are Functionally Independent Indicators Misleading in Non-Pathological Cases?



Indicator	A	B
Generational distance	3.46396	2.37411
Spacing (Schott)	0.26476	0.19989
Max Pareto front error	3.35489	3.31314
Extent	3.56039	3.57319

Repeat 10000 times:

- Generate set A by generating 1000 points in the box. For each point, accept it into A with probability p proportional to distance from PF
- Generate B in exactly same way but discard points that dominate A
- Apply four indicators

Indicator	GD	Spacing	MPFE	Extent
Error rate	0.598	0.270	0.621	0.231
Number of errors	1.720			

- Mean number of "errors" from indicators = 1.72

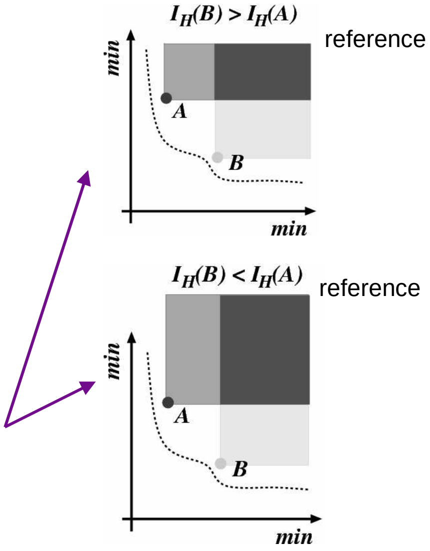
Three Recommended Unary Indicators

- We recommend and provide software for three unary quality indicators:
  - The hypervolume indicator (Zitzler, 1998)
  - The unary epsilon indicator (multiplicative and additive, Zitzler et al, 2003)
  - The R_2 and R_3 indicators (based on those proposed by Hansen and Jaszkiewicz, 1998)
- **Each indicator is based on different preference information – therefore using them all will provide more information than using just one**
- We also provide tools for pre-processing steps: setting reference points and reference sets
- And tools for non-parametric statistical testing of differences in indicator values

Recommended quality indicator 1:

The Hypervolume Indicator

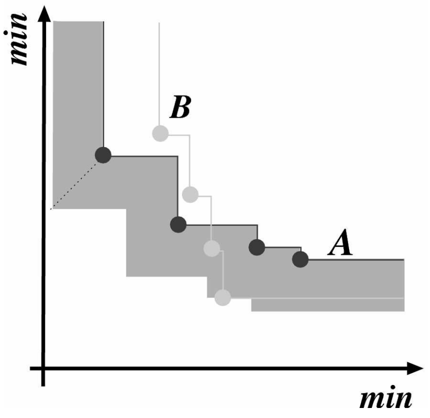
- Advantages
  - Compatibility with $\ntriangleright$ and completeness with $\triangleright$
  - Conceptually intuitive meaning
  - Scaling independent
- Weaknesses
  - Computational cost for large number of objectives (see While, 2005 – this conference)
  - Need reference point
  - Reference point affects ordering of pairs of incomparable sets



Principles of recommended unary indicator 2

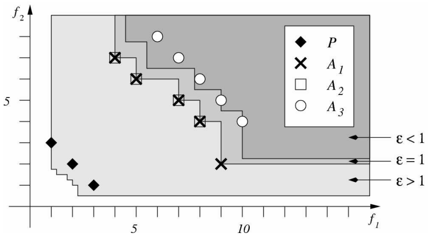
Binary ε-indicator

Principle: what is the smallest amount ε that I have to translate the set *A* so that every point in B is covered ?



Recommended unary indicator 2:

The Unary ε-indicator



Definition (Binary ϵ-indicator) *We define the binary ϵ-indicator I_ϵ as*

$$I_\epsilon(A,B) = \inf_{\epsilon \in \mathbb{R}} \{\forall \boldsymbol{z}^2 \in B \, \exists \boldsymbol{z}^1 \in A : \boldsymbol{z}^1 \succeq_\epsilon \boldsymbol{z}^2\}$$

for any two approximation sets $A, B \in \Omega$.

B	A			
	A_1	A_2	A_3	P
A_1	1	2	2	1/2
A_2	1	1	3/2	3/7
A_3	9/10	1	1	1/3
P	4	4	6	1

Definition (Unary ϵ-indicator) *The indicator is based on the binary ϵ-indicator and a reference set of points:*

$$I_{\epsilon 1}(A) = I_\epsilon(A,P)$$

where P is the Pareto front. If P is not known, then any reference set R can be used instead.

Recommended unary indicator 2:

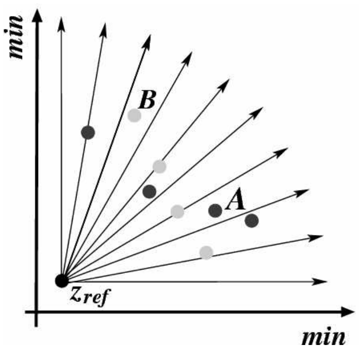
The Unary ε-indicator

- Advantages:
  - If used with Pareto front as reference set, comparison method is compatible with $\triangleright$ for pairs A,B with A=P and B≠P.
  - If used with general reference set, compatible with $\ntriangleright$ and complete with $\succ\succ$
  - Indicator is fast to compute
  - Intuitive meaning: how much do I need to translate/scale the set A so that it covers the reference set?
- Weaknesses:
  - Need to choose a reference set, and this affects the outcome of the indicator
  - Objective vectors need to be in positive hyper-quadrant

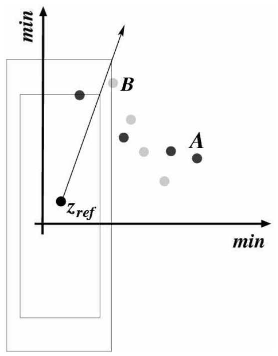
Principles of recommended unary indicator 3

Uniformly Distributed Utility Functions

On each member of a set, *Λ*, of uniformly distributed scalarizing vectors, *λ*:



…measure the distance of the point (in each set) that is closest to the reference point,...



... and add up the differences in these distances.

Asymptotically compatible with $\triangleright$/and complete with $\succ\succ$

Recommended unary indicator 3

The R_R Indicators (Hansen and Jaszkiewicz, 1998)

Define the utility *u**(*A, λ*) of the approximation set *A* on scalarizing vector *λ* as the minimum distance of a point in the set *A* from the reference point. Then:

$$I_{R2}(A,B) = \frac{\sum_{\lambda\in\Lambda} u^*(\lambda,B) - u^*(\lambda,A)}{|\Lambda|},$$

$$I_{R3}(A,B) = \frac{\sum_{\lambda\in\Lambda}[u^*(\lambda,B) - u^*(\lambda,A)]/u^*(\lambda,B)}{|\Lambda|}$$

To obtain a unary indicator from these, replace *B* with a reference set, *R*:

$$I_{R2_R}(A) \overset{\text{def}}{=} I_{R2}(A,R)$$

$$I_{R3_R}(A) \overset{\text{def}}{=} I_{R3}(A,R)$$

These effectively measure the difference in the mean distance of the attainment surfaces *A* and *R*, from the reference point.

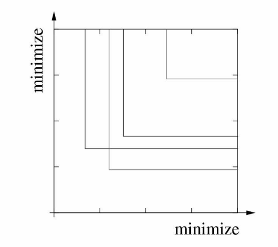
Choosing Reference Points

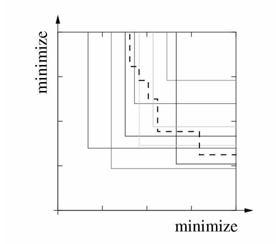
- If bounds of the objective space are known, use these
- If not:
  - concatenate all approximation sets and compute the ideal and nadir points
  - shift the ideal/nadir point so that it strictly dominates/is strictly dominated by all points in the approximation sets
- It would be good practice to report the bound points used, to allow future comparisons to be made

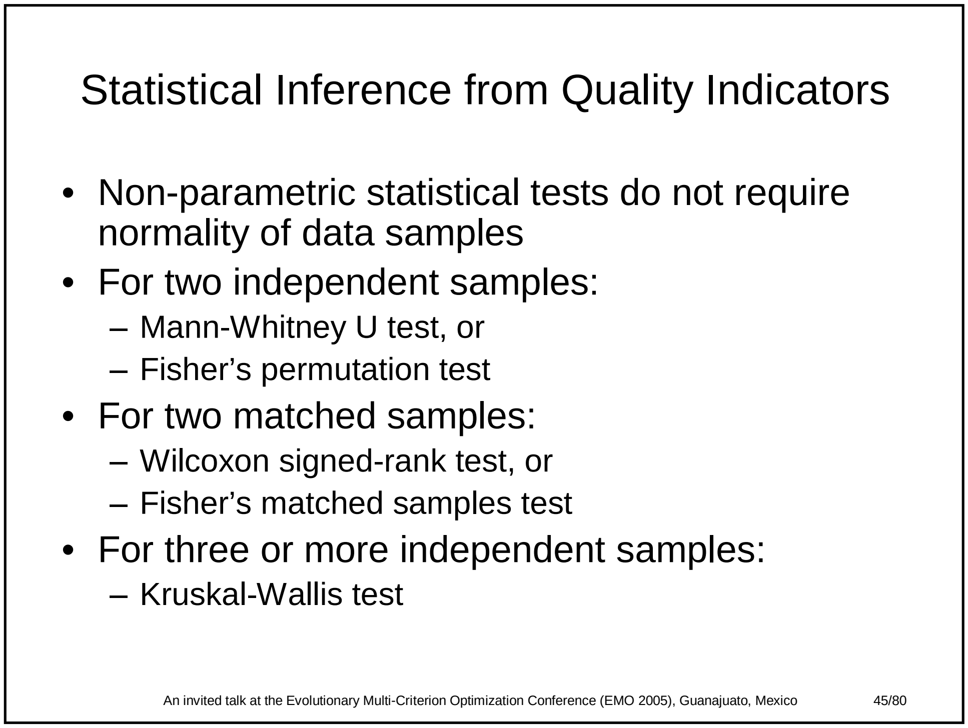
Choosing Reference Sets

Potentially the choice can affect the outcome of a comparison study. However, there are several choices which are *a priori* unbiased for any particular algorithm:

- The true Pareto front, if it is known
- An approximation set from the literature, thought to be a good approximation
- The 50% attainment surface of random search
  - half of all points in the decision space will weakly dominate this surface
  - we provide a software tool for generating this

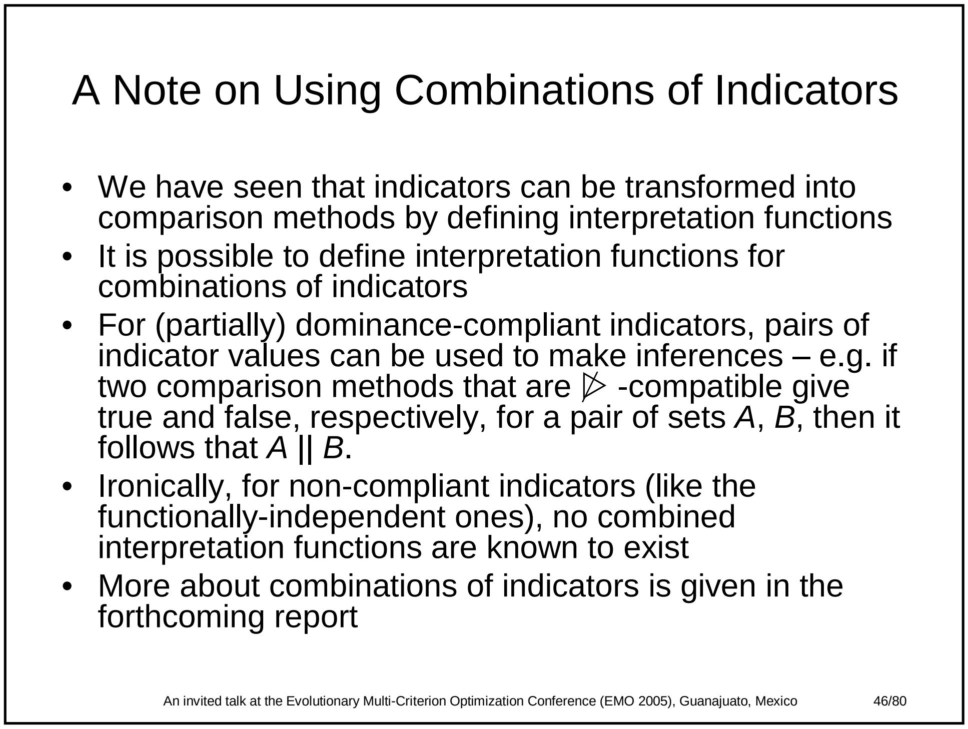






Statistical Inference from Quality Indicators

- Non-parametric statistical tests do not require normality of data samples
- For two independent samples:
  - Mann-Whitney U test, or
  - Fisher’s permutation test
- For two matched samples:
  - Wilcoxon signed-rank test, or
  - Fisher’s matched samples test
- For three or more independent samples:
  - Kruskal-Wallis test



A Note on Using Combinations of Indicators

- We have seen that indicators can be transformed into comparison methods by defining interpretation functions
- It is possible to define interpretation functions for combinations of indicators
- For (partially) dominance-compliant indicators, pairs of indicator values can be used to make inferences – e.g. if two comparison methods that are $\ntriangleright$ -compatible give true and false, respectively, for a pair of sets *A*, *B*, then it follows that *A* || *B*.
- Ironically, for non-compliant indicators (like the functionally-independent ones), no combined interpretation functions are known to exist
- More about combinations of indicators is given in the forthcoming report

Multiple Testing Issues

When multiple statistical tests are carried out on the same samples, p-values of the individual tests do not accurately reflect the probabilities.

- What if I have more than two algorithms?
  - Option 1: use the Kruskal-Wallis test. This provides an overall judgment if any algorithms differ and, if so, gives all pair-wise one tailed *p*-values (software provided)
  - Option 2: use standard test but adjust the *p*-values, e.g. using Bonferroni correction (facility not provided)
- What if I have more than one indicator?
  - Option 1: collect separate samples for each indicator
  - Option 2: use correction factors
  - Option 3: do tests normally but report that *p*-values do not reflect true probabilities

More information given in the forthcoming report

Summary of Quality Indicators

- Quality indicators are always based on preference information
- Comparison methods based on an indicator and an interpretation function can be compatible and complete with dominance, or not
- We recommend the use of quality indicators that are the basis of comparison methods that are compatible with the $\ntriangleright$ relation
- Non-parametric tests can be used to test significance of differences in the distribution of indicator values
- Reference points and reference sets are needed for some indicators – we have given some methods for defining these

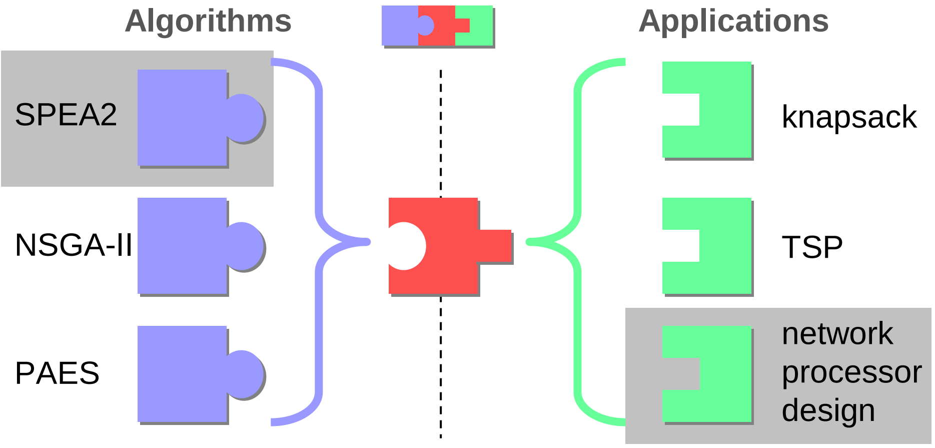
PART 3

CASE STUDY

Overview

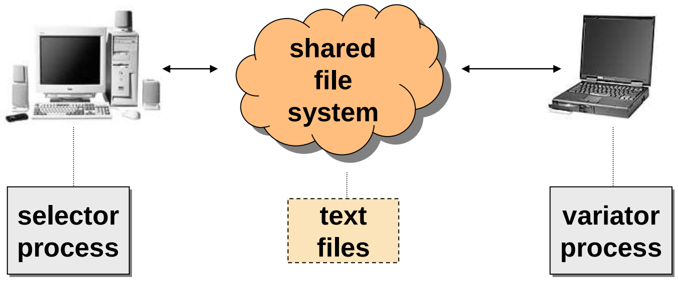
- **PISA**
  Making Performance Assessment Easy
- **Toolflow**
  How to Use PISA for Assessment
- **Sample Scenario**
  From Runs to Statistical Evidence

The Concept of PISA



Platform and programming language independent **I**nterface for **S**earch **A**lgorithms *[Bleuler et al.: 2002]*

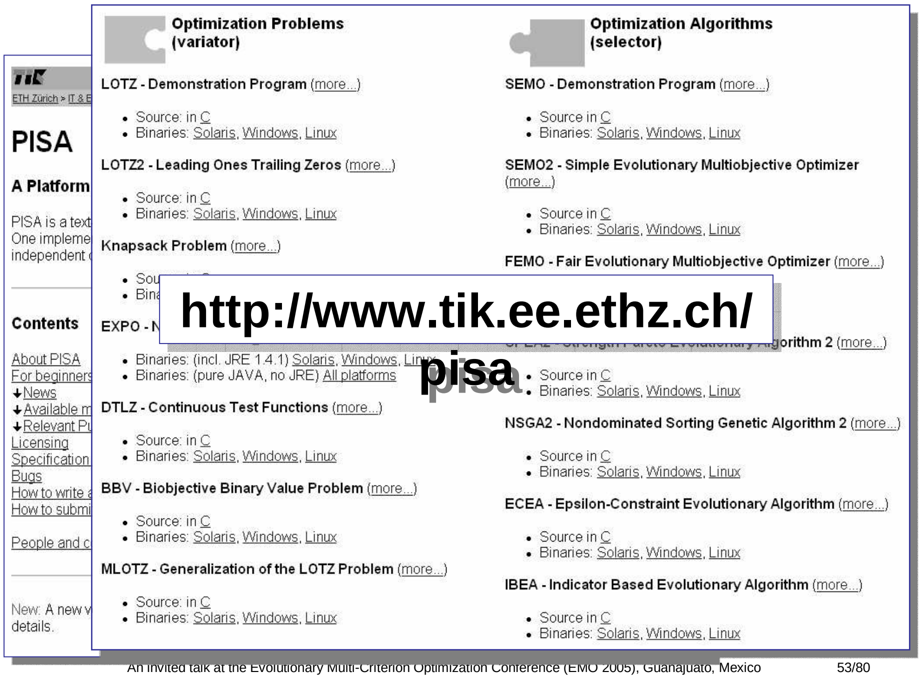
PISA: Implementation



application independent
- selection
- individuals described by IDs and objective vectors

handshake protocol
- state / action
- individual IDs
- objective vectors

application dependent
- variation operators
- stores and manages individuals



ETH Zürich > IT & E

PISA

A Platform

PISA is a text
One impleme
independent

Contents

About PISA
For beginners
↓News
↓Available m
↓Relevant Pu
Licensing
Specification
Bugs
How to write a
How to submi

People and c

New: A new v
details.

Optimization Problems (variator)

LOTZ - Demonstration Program (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

LOTZ2 - Leading Ones Trailing Zeros (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

Knapsack Problem (more...)

- Sou
- Bina

EXPO - N

- Binaries: (incl. JRE 1.4.1) Solaris, Windows, Linux
- Binaries: (pure JAVA, no JRE) All platforms

DTLZ - Continuous Test Functions (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

BBV - Biobjective Binary Value Problem (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

MLOTZ - Generalization of the LOTZ Problem (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

Optimization Algorithms (selector)

SEMO - Demonstration Program (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

SEMO2 - Simple Evolutionary Multiobjective Optimizer (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

FEMO - Fair Evolutionary Multiobjective Optimizer (more...)

SPEA2 - Strength Pareto Evolutionary Algorithm 2 (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

NSGA2 - Nondominated Sorting Genetic Algorithm 2 (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

ECEA - Epsilon-Constraint Evolutionary Algorithm (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

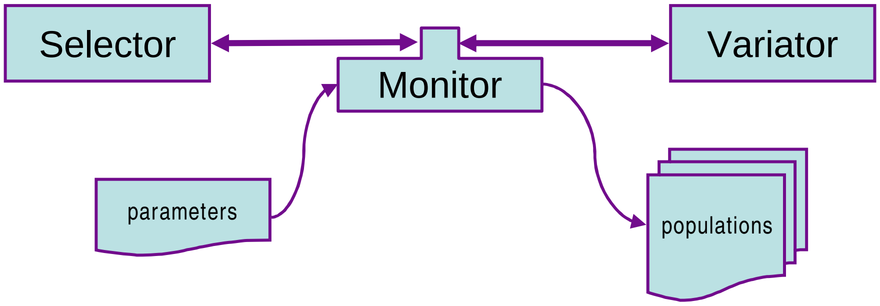
IBEA - Indicator Based Evolutionary Algorithm (more...)

- Source: in C
- Binaries: Solaris, Windows, Linux

http://www.tik.ee.ethz.ch/pisa

PISA Monitor

- Observe communication between Variator and Selector and store intermediate populations.
- Automatically invoke optimization runs.



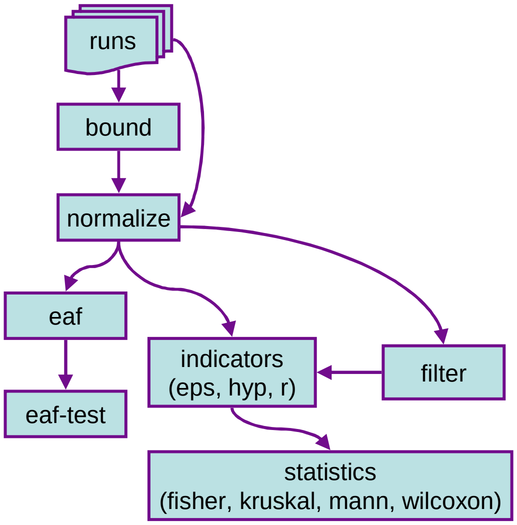
Overview

- **PISA**
  Making Performance Assessment Easy
- **Toolflow**
  How to Use PISA for Assessment
- **Sample Scenario**
  From Runs to Statistical Evidence

Tools for Generated Runs

- **Postprocessing** of objective vector sets
  - bound: calculates lower and upper bounds of obj. vectors
  - normalize: transforms all objective values to the interval [1,2]
  - filter: computes the nondominated front of obj. vectors
- Computation of quality **indicators**
  - eps_ind: unary epsilon indicators
  - hyp_ind: unary hypervolume indicators
  - r_ind: R indicators
- **Statistics**
  - 5 different tests: 2x Fisher, Kruskal, Mann and Wilcoxon
- **Attainment Function**
  - eaf: computes attainment function
  - eaf-test: tests two functions for difference

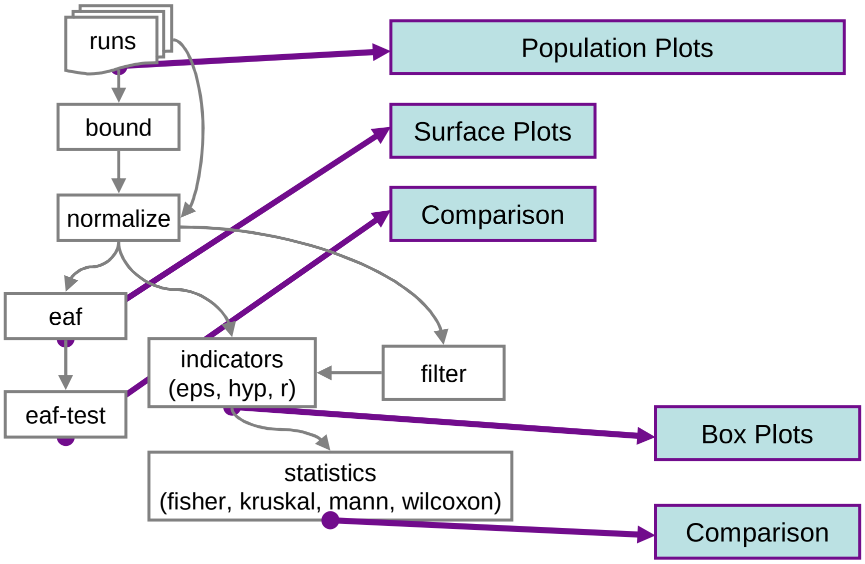
Recommended Tool Flow



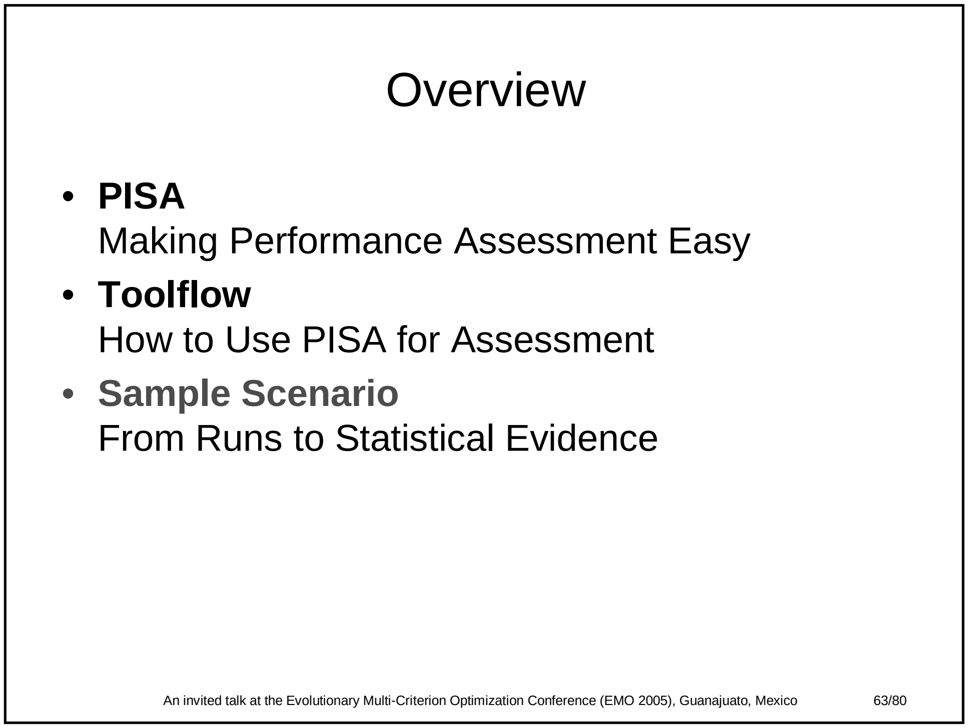
runs
bound
normalize
eaf
eaf-test
indicators (eps, hyp, r)
filter
statistics (fisher, kruskal, mann, wilcoxon)

- All intermediate data files are collected in a common directory.
- The whole process can be controlled by batch files (samples for all platforms are provided).

Output of Tool Flow

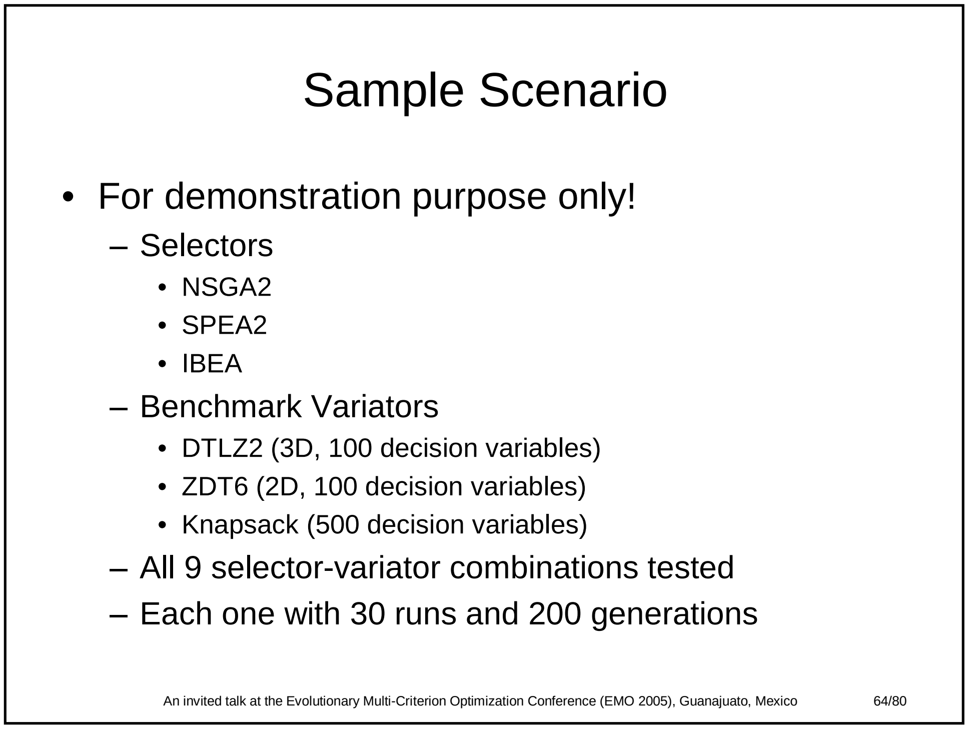


runs
Population Plots
bound
Surface Plots
normalize
Comparison
eaf
eaf-test
indicators (eps, hyp, r)
filter
Box Plots
statistics (fisher, kruskal, mann, wilcoxon)
Comparison



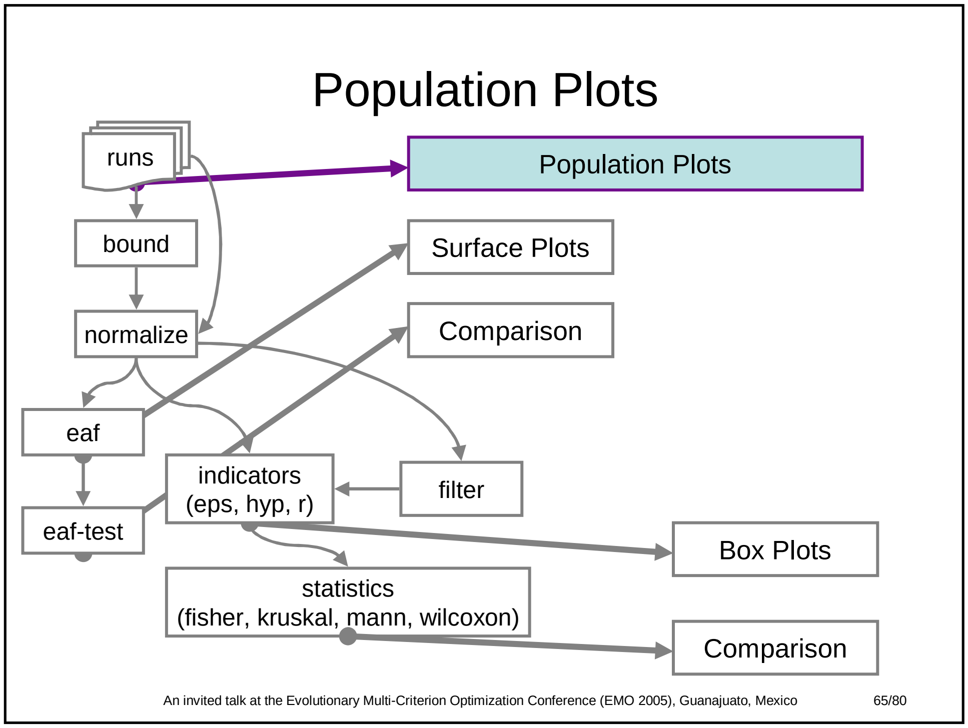
Overview

- **PISA**
  Making Performance Assessment Easy
- **Toolflow**
  How to Use PISA for Assessment
- **Sample Scenario**
  From Runs to Statistical Evidence



Sample Scenario

- For demonstration purpose only!
  - Selectors
    - NSGA2
    - SPEA2
    - IBEA
  - Benchmark Variators
    - DTLZ2 (3D, 100 decision variables)
    - ZDT6 (2D, 100 decision variables)
    - Knapsack (500 decision variables)
  - All 9 selector-variator combinations tested
  - Each one with 30 runs and 200 generations



Population Plots

runs

bound

normalize

eaf

eaf-test

indicators
(eps, hyp, r)

filter

statistics
(fisher, kruskal, mann, wilcoxon)

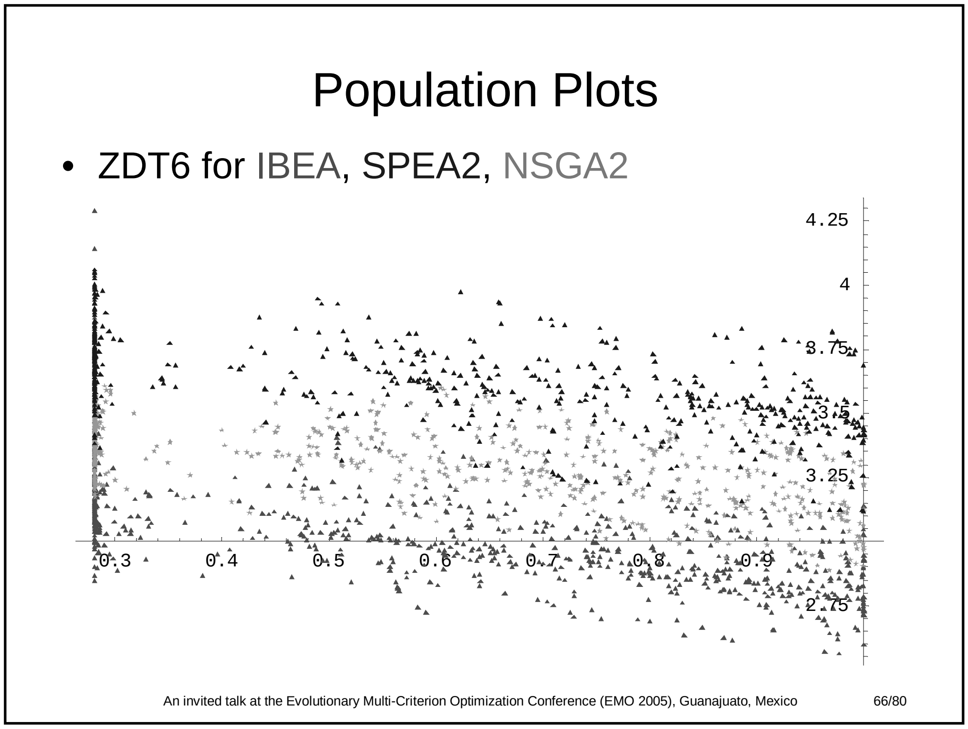
Population Plots

Surface Plots

Comparison

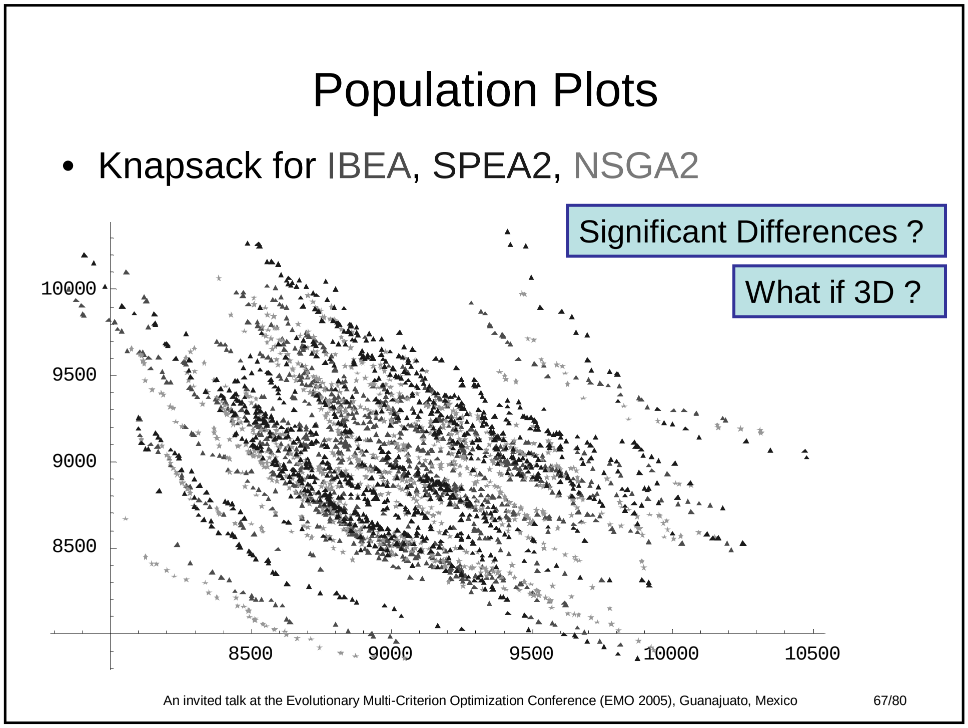
Box Plots

Comparison



Population Plots

- ZDT6 for IBEA, SPEA2, NSGA2



Population Plots

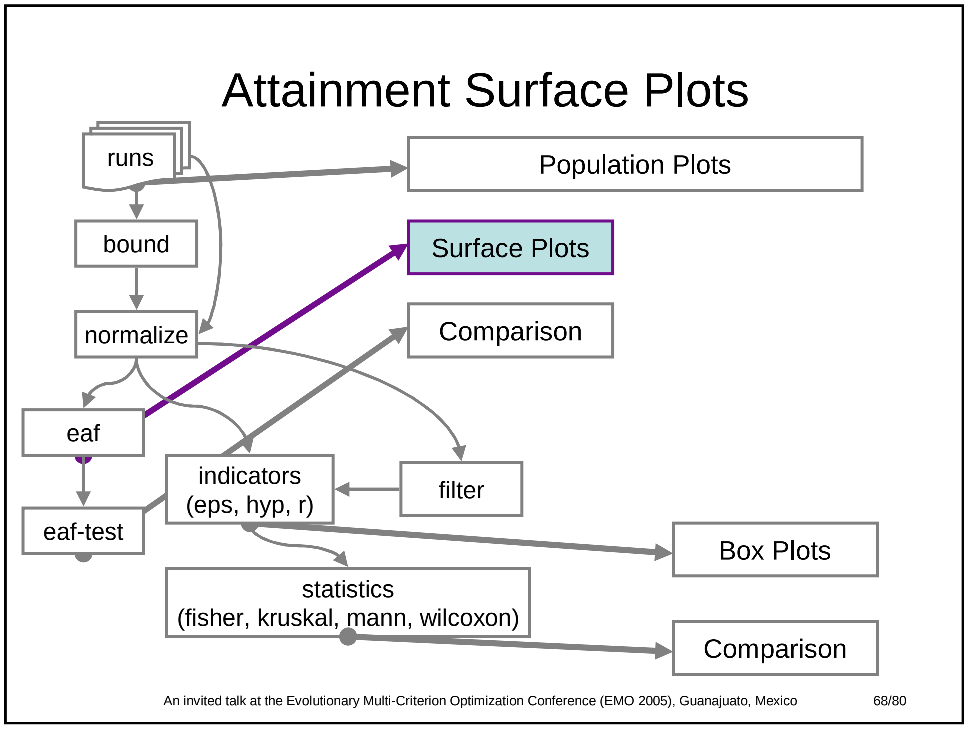
- Knapsack for IBEA, SPEA2, NSGA2

Significant Differences ?

What if 3D ?

An invited talk at the Evolutionary Multi-Criterion Optimization Conference (EMO 2005), Guanajuato, Mexico

67/80



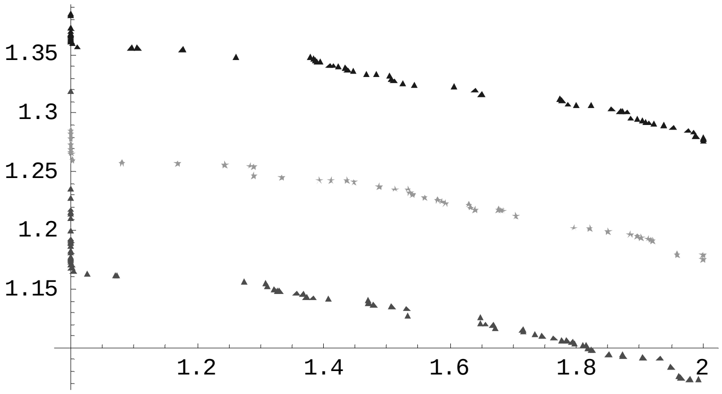
Attainment Surface Plots

An invited talk at the Evolutionary Multi-Criterion Optimization Conference (EMO 2005), Guanajuato, Mexico

68/80

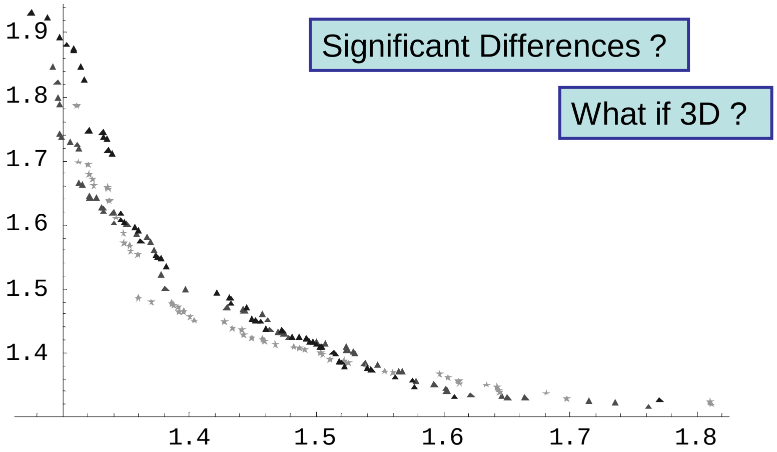
Attainment Plots

- 50% surface
  ZDT6 for IBEA, SPEA2, NSGA2



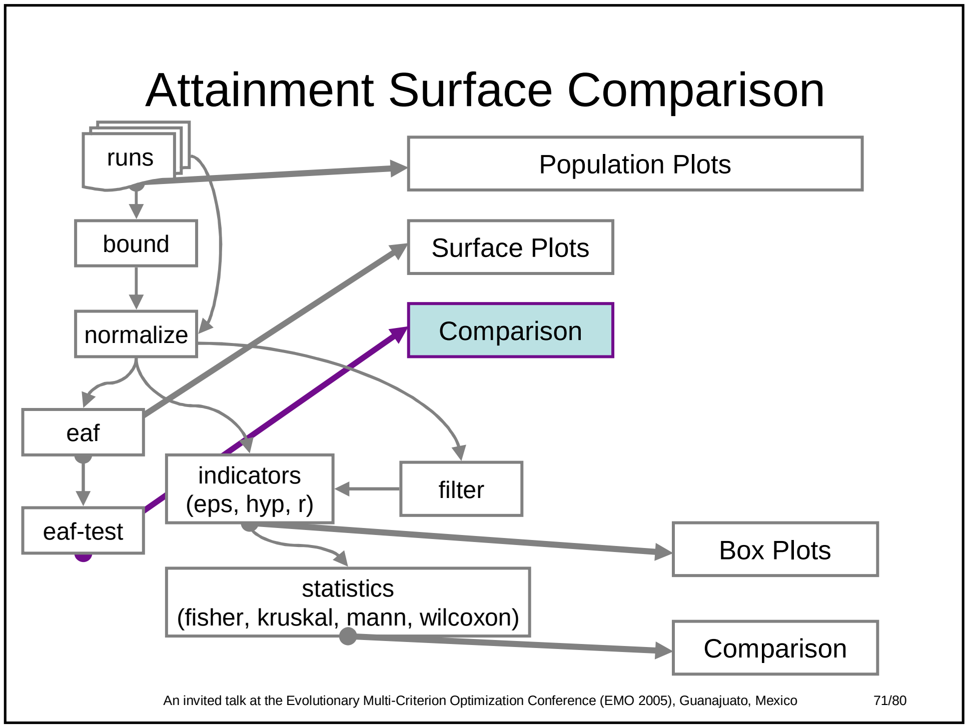
Attainment Plots

- 50% surface
  knapsack for IBEA, SPEA2, NSGA2

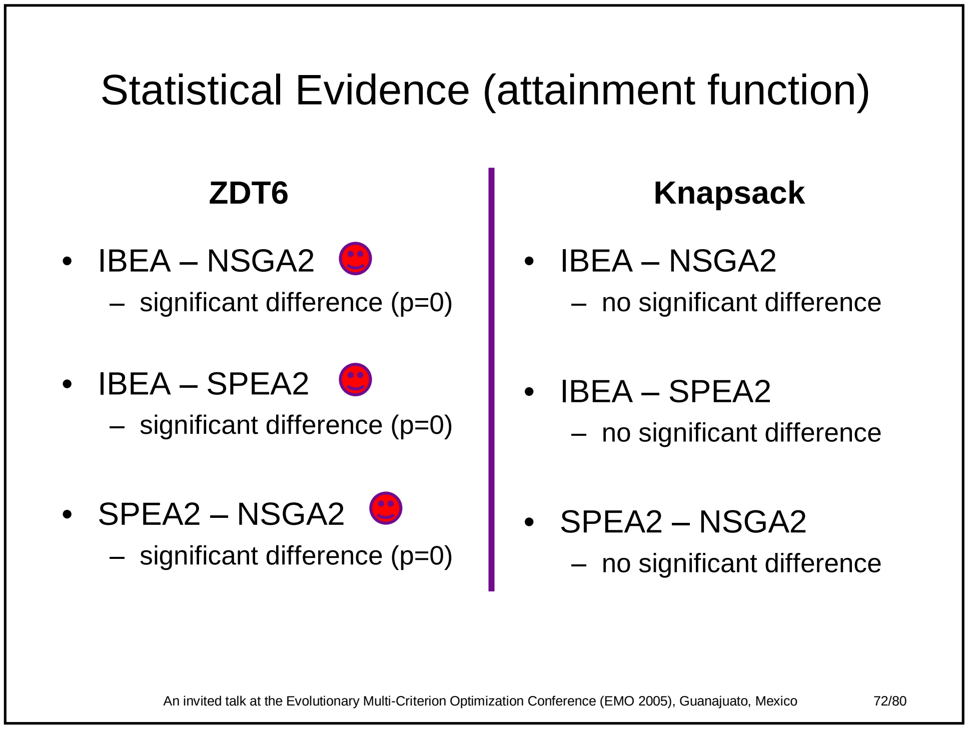


Significant Differences ?

What if 3D ?



Attainment Surface Comparison



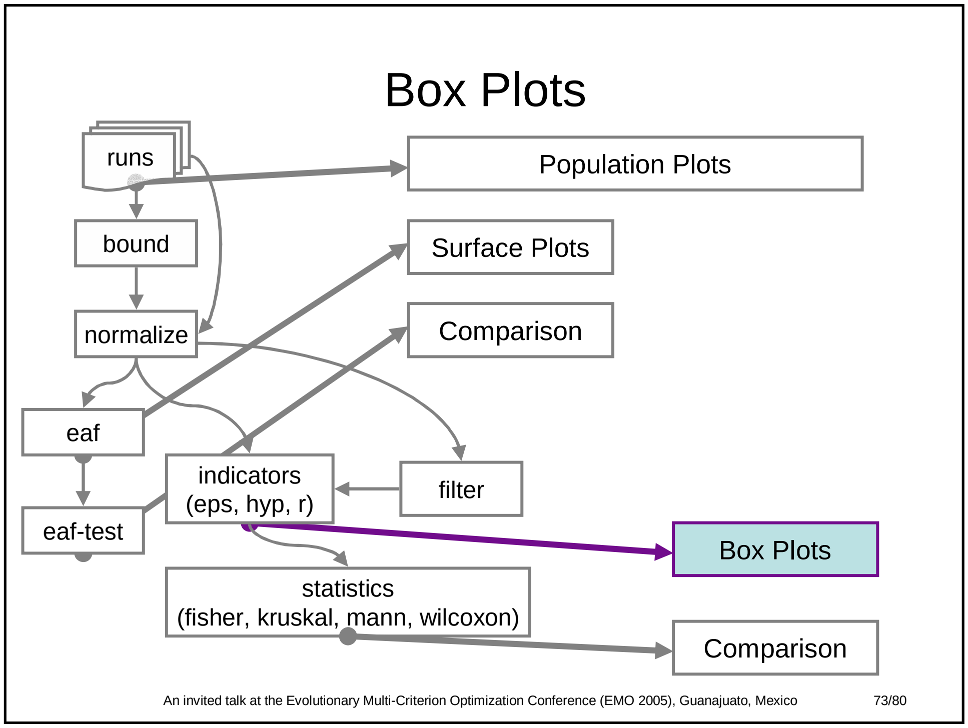
Statistical Evidence (attainment function)

ZDT6

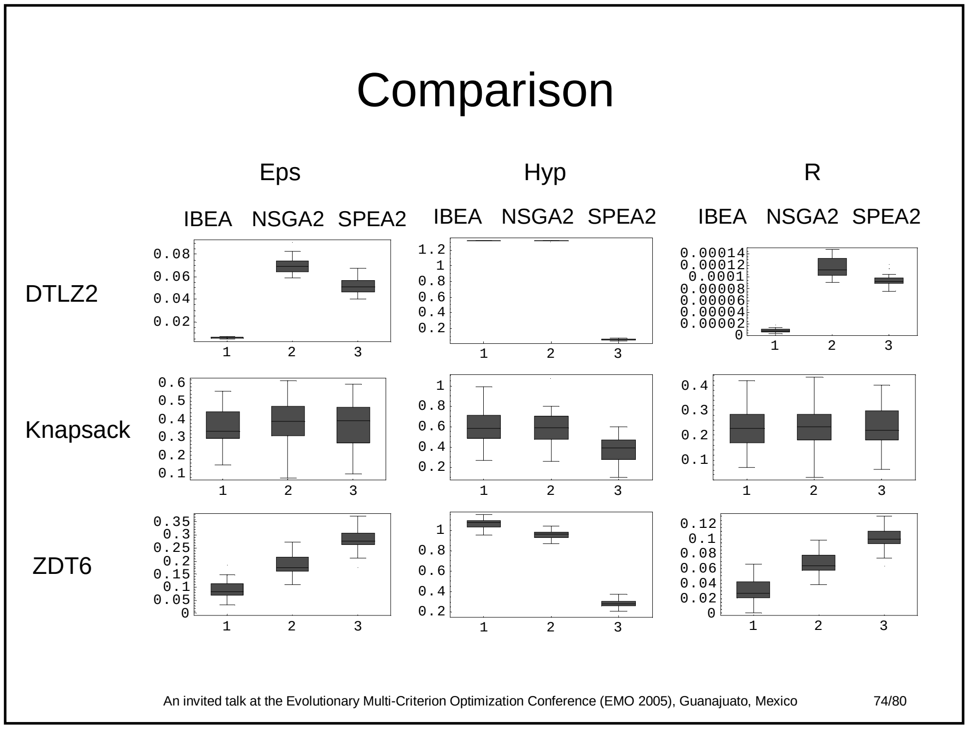
- IBEA – NSGA2 ☺
  - significant difference (p=0)
- IBEA – SPEA2 ☺
  - significant difference (p=0)
- SPEA2 – NSGA2 ☺
  - significant difference (p=0)

Knapsack

- IBEA – NSGA2
  - no significant difference
- IBEA – SPEA2
  - no significant difference
- SPEA2 – NSGA2
  - no significant difference

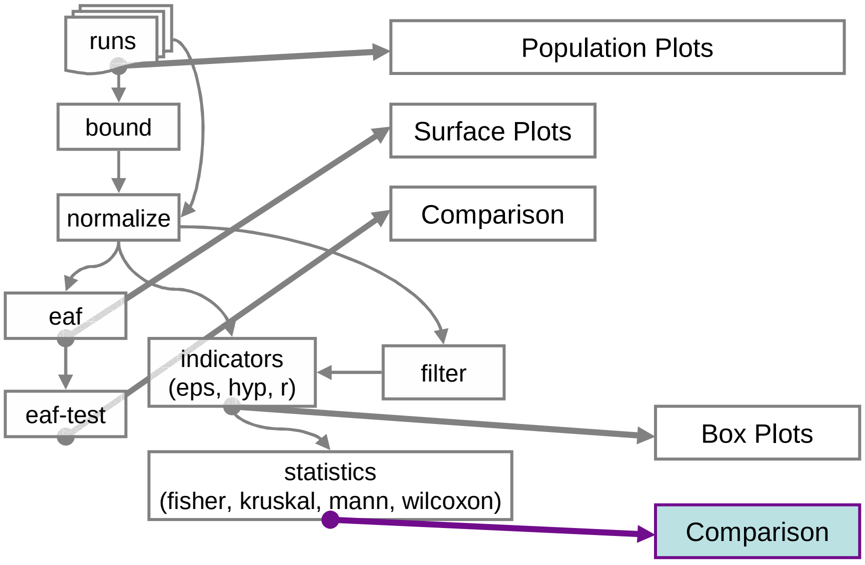


Box Plots



Comparison

Comparison



runs

bound

normalize

eaf

eaf-test

indicators
(eps, hyp, r)

filter

statistics
(fisher, kruskal, mann, wilcoxon)

Population Plots

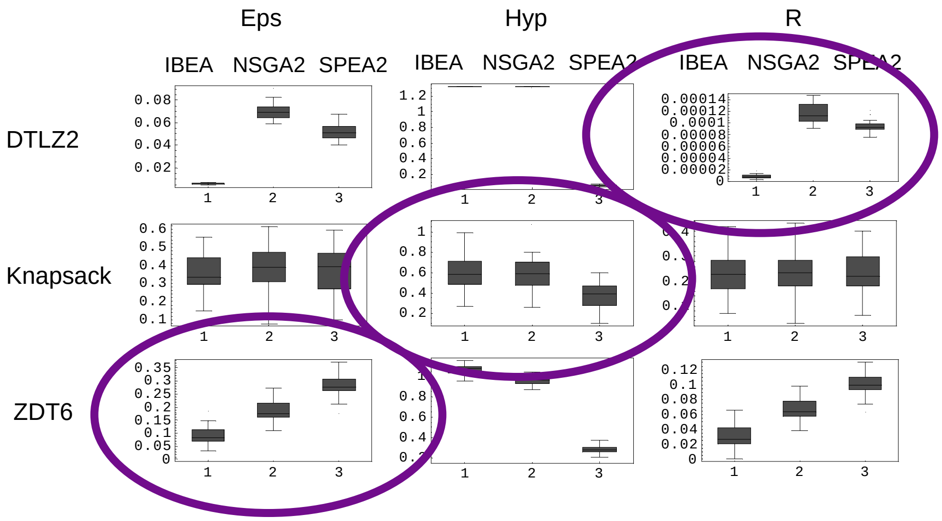
Surface Plots

Comparison

Box Plots

Comparison

Comparison



Statistical Evidence (Kruskal Test)

ZDT6
Eps

is better than	IBEA	NSGA2	SPEA2
IBEA		~0 ☺	~0 ☺
NSGA2	1		~0 ☺
SPEA2	1	1	

Overall p-value = 6.22079e-17.
Null hypothesis rejected (alpha 0.05)

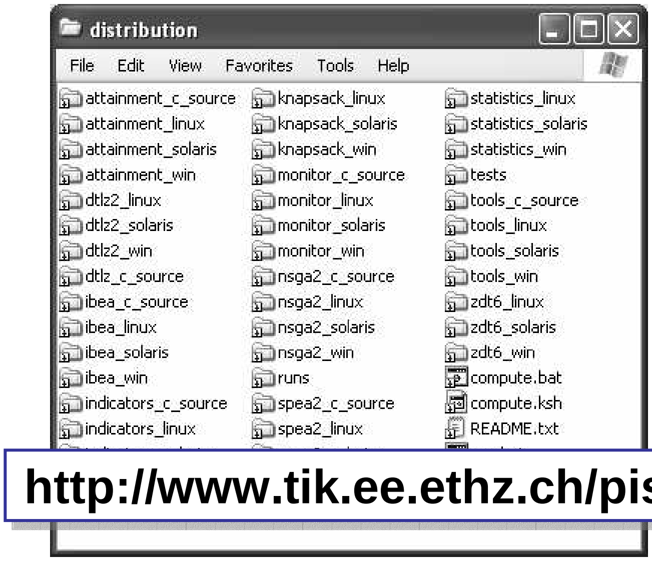
DTLZ2
R

is better than	IBEA	NSGA2	SPEA2
IBEA		~0 ☺	~0 ☺
NSGA2	1		1
SPEA2	1	~0 ☺	

Overall p-value = 7.86834e-17.
Null hypothesis rejected (alpha 0.05)

Knapsack/Hyp: H0 = No significance of any differences

Code Source, Binaries and Documentation



attainment_c_source
attainment_linux
attainment_solaris
attainment_win
dtlz2_linux
dtlz2_solaris
dtlz2_win
dtlz_c_source
ibea_c_source
ibea_linux
ibea_solaris
ibea_win
indicators_c_source
indicators_linux
knapsack_linux
knapsack_solaris
knapsack_win
monitor_c_source
monitor_linux
monitor_solaris
monitor_win
nsga2_c_source
nsga2_linux
nsga2_solaris
nsga2_win
runs
spea2_c_source
spea2_linux
statistics_linux
statistics_solaris
statistics_win
tests
tools_c_source
tools_linux
tools_solaris
tools_win
zdt6_linux
zdt6_solaris
zdt6_win
compute.bat
compute.ksh
README.txt

http://www.tik.ee.ethz.ch/pisa

Thanks for listening !