Visualization in Multiobjective Optimization

Bogdan Filipič **Tea Tušar**

CEC Tutorial, Donostia - San Sebastián, June 5, 2017

Computational Intelligence Group
Department of Intelligent Systems
Jožef Stefan Institute
Ljubljana, Slovenia

Final version

Tutorial slides are available at
`http://dis.ijs.si/tea/research.htm`

Contents

Introduction

Introduction

Multiobjective optimization problem

Minimize

$$\mathbf{f}\colon X \to F$$
$$\mathbf{f}\colon (x_1, \ldots, x_n) \mapsto (f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n))$$

- X is an n-dimensional decision space
- $F \subseteq \mathbb{R}^m$ is an m-dimensional objective space $(m \geq 2)$

Conflicting objectives $\to$ a set of optimal solutions

- Pareto set in the decision space
- Pareto front in the objective space

Introduction

Visualization in multiobjective optimization

Useful for different purposes [14]

- Analysis of solutions and solution sets
- Decision support in interactive optimization
- Analysis of algorithm performance

Visualizing solution sets in the decision space

- Problem-specific
- If $X \subseteq \mathbb{R}^m$, any method for visualizing multidimensional solutions can be used
- Not the focus of this tutorial

Introduction

Visualizing solution sets in the objective space

- Interested in sets of mutually nondominated solutions called approximation sets
- Different from ordinary multidimensional solution sets
- The focus of this tutorial

Challenges

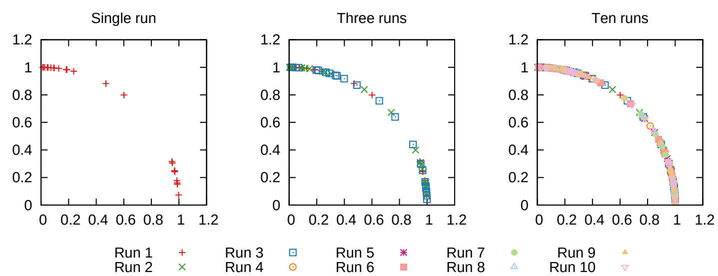
- High dimension and large number of solutions
- Limitations of computing and displaying technologies
- Cognitive limitations

Introduction

Visualization can be hard even in 2-D

Stochastic optimization algorithms

- Single run $\to$ single approximation set
- Multiple runs $\to$ multiple approximation sets



The Empirical Attainment Function (EAF) [15] or the Average Runtime Attainment Function (aRTA) [3] can be used in such cases

Introduction

This tutorial is not about

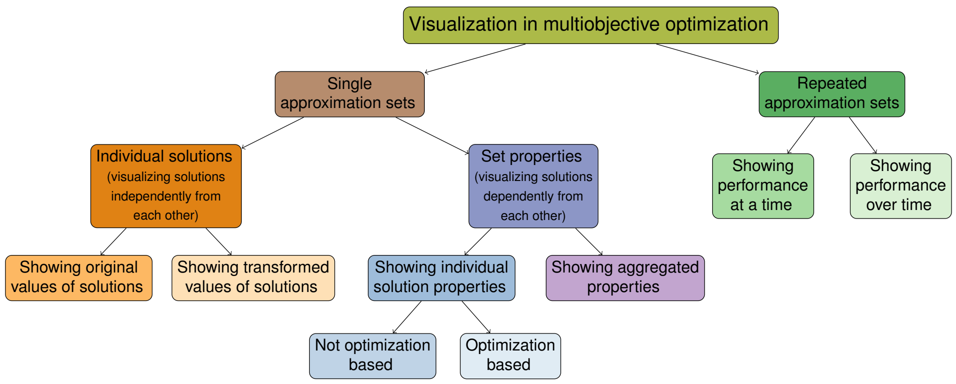
- Visualization of a few solutions for decision making purposes (see [29])
- Visualization in the decision space
- General multidimensional visualization methods not previously used on approximation sets

This tutorial covers

- Visualization of entire sets in the objective space
  - Single approximation sets [1]
  - Repeated approximation sets [2, 3]

A taxonomy of visualization methods

A taxonomy of visualization methods



Visualizing single approximation sets

Methodology

Evaluating and comparing visualization methods

- No existing methodology for evaluating or comparing visualization methods
- Propose benchmark approximation sets (analog to benchmark problems in multiobjective optimization)
- Visualize the sets using different methods
- Observe which set properties are distinguishable after visualization

Benchmark approximation sets

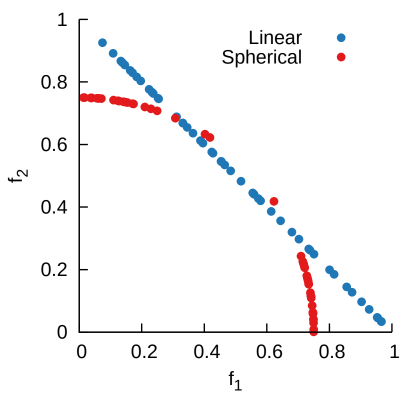
Two different sets that can be instantiated in any dimension [1]

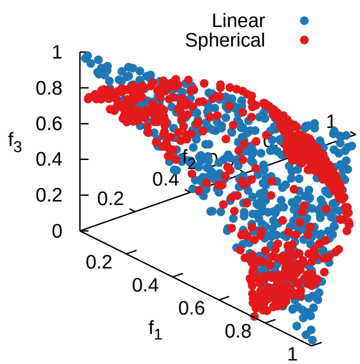
- Linear with a uniform distribution of solutions
- Spherical with a nonuniform distribution of solutions (more at the corners and less at the center)
- Sets are intertwined

Size of each set

- 2-D: 50 solutions
- 3-D: 500 solutions
- 4-D: only 300 solutions since most methods cannot handle more

Benchmark approximation sets



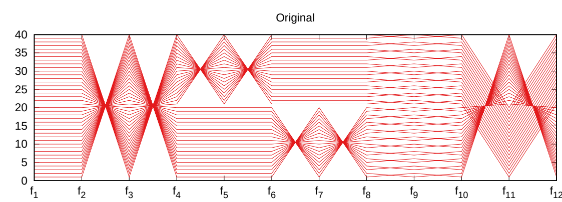


Benchmark approximation sets

These two sets are not sufficient for all purposes!

Missing:

- A set with knees
- A set with different relations between objectives, temporarily using [13]:



- A sequence of sets mimicking convergence in time
- ... (possibly others)

Desired properties of visualization methods

Demonstration on the 4-D linear and spherical sets

- Preservation of the
  - Dominance relation between solutions
  - Front shape
  - Objective range
  - Distribution of solutions
- Robustness
- Handling of large sets
- Simultaneous visualization of multiple sets
- Scalability in number of objectives
- Simplicity

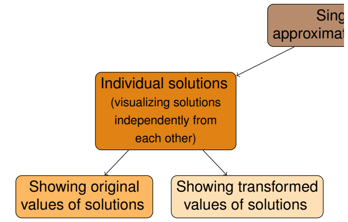
Demonstration on the 12-D approximation set

- Showing relations between objectives

Visualizing single approximation sets

Individual solutions (Visualizing solutions independently from each other) $\rightarrow$ Showing original values of solutions

- Scatter plot matrix
- Bubble chart
- Parallel coordinates [19]
- Radar chart
- Chord diagram [22], TBA
- Heat maps [32]
- Interactive decision maps [26]



Scatter plot matrix

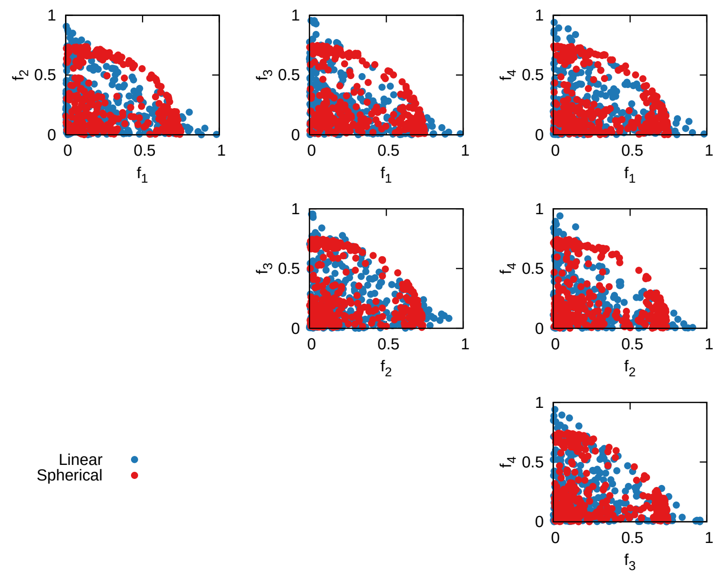
Most often

- Scatter plot in a 2-D space
- Matrix of all possible combinations
- m objectives $\rightarrow \frac{m(m-1)}{2}$ different combinations

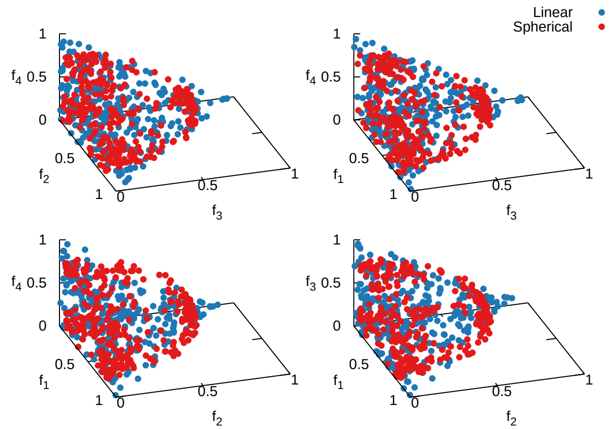
Alternatively

- Scatter plot in a 3-D space
- m objectives $\rightarrow \frac{m(m-1)(m-2)}{6}$ different combinations

Scatter plot matrix



Scatter plot matrix



Preservation of the								
dominance relation	front shape	objective range	distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
×	≈	✓	≈	✓	≈	✓	×	✓

Bubble chart

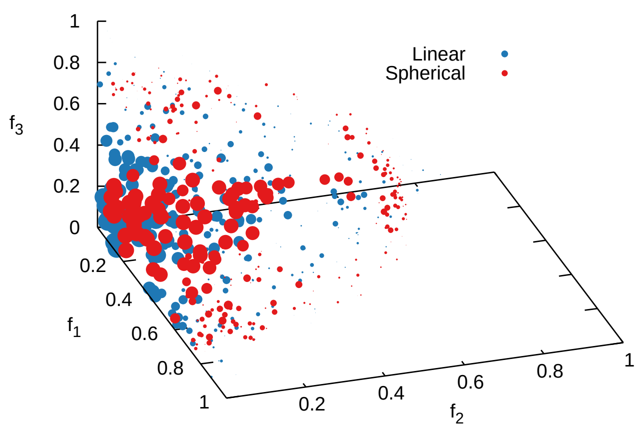
4-D objective space

- Similar to a 3-D scatter plot
- Fourth objective visualized with point size

5-D objective space

- Fifth objective visualized with colors

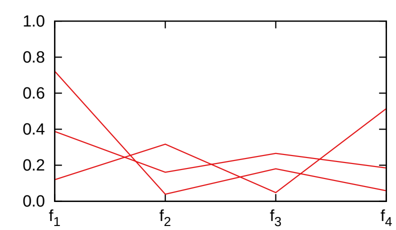
Bubble chart



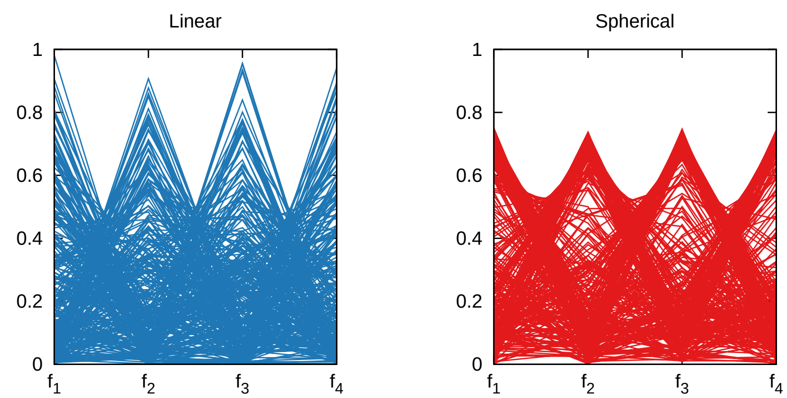
Preservation of the								
dominance relation	front shape	objective range	distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
×	≈	✓	≈	✓	≈	✓	×	✓

Parallel coordinates

- m objectives $\rightarrow$ m parallel axes
- Solution represented as a polyline with vertices on the axes
- Position of each vertex corresponds to that objective value
- No loss of information

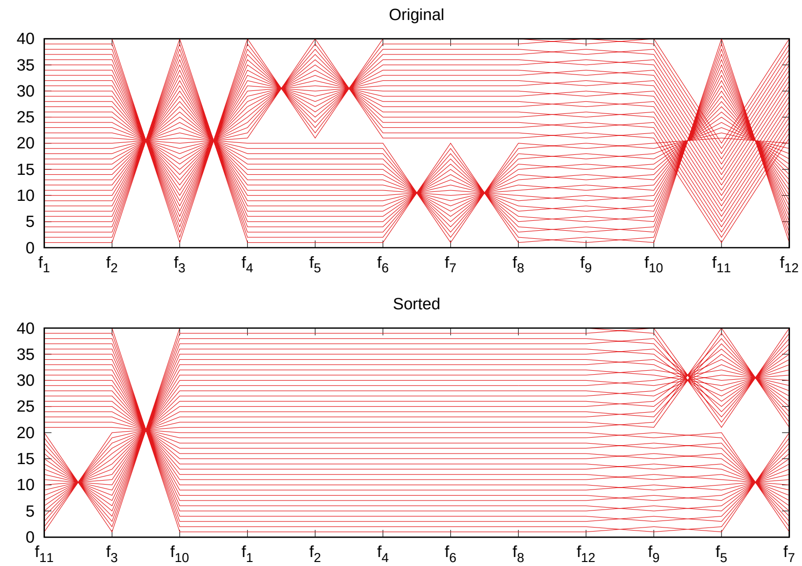


Parallel coordinates



Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
≈	×	✓	≈	✓	×	×	✓	✓

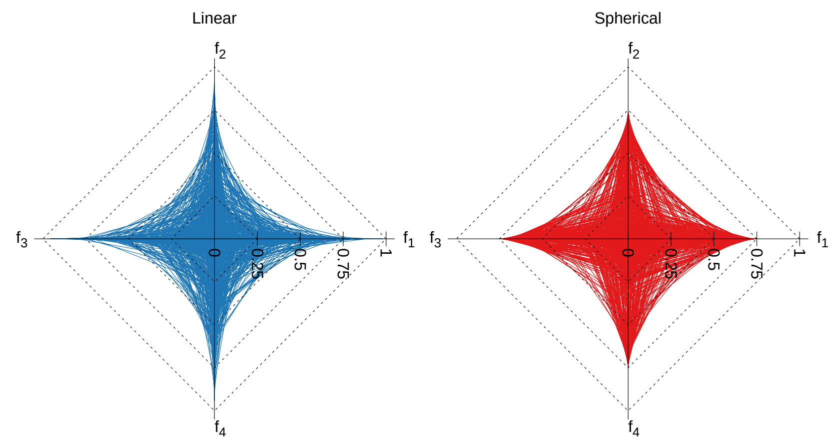
Parallel coordinates



Radar chart

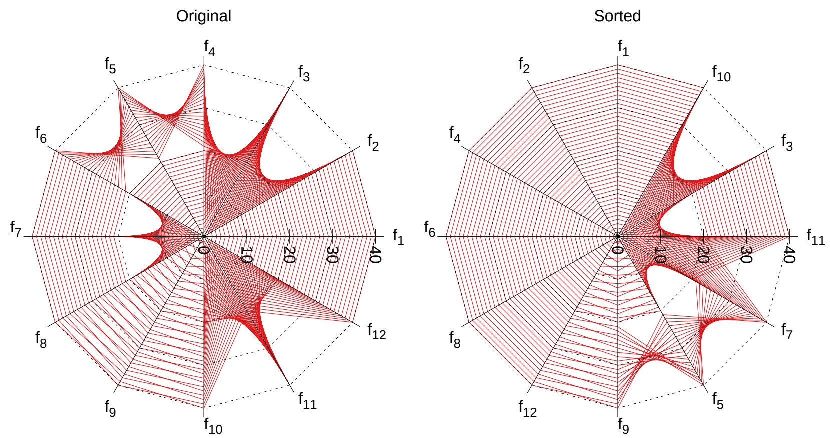
- Similar to parallel coordinates
- Additionally connects the two extreme coordinates
- m objectives $\rightarrow$ m radial axes
- Also called a *spider chart*, *polar chart*, *star plot*, ...

Radar chart



Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
≈	×	✓	≈	✓	×	×	✓	✓

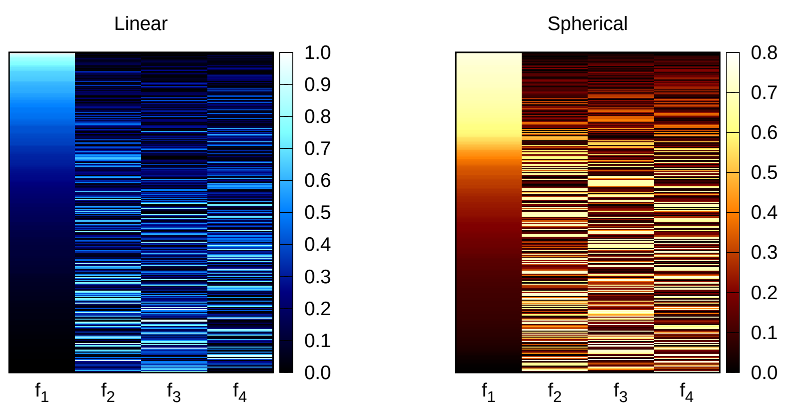
Radar chart



Heat maps

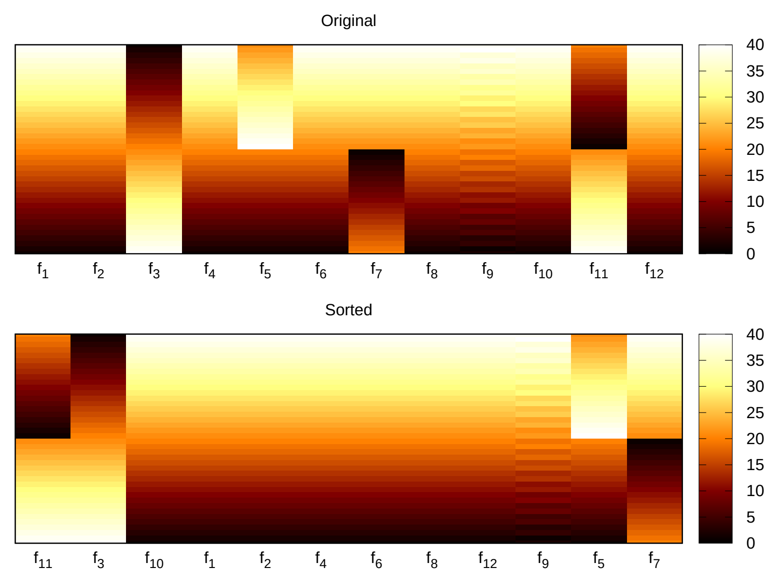
- m objectives $\rightarrow$ m columns
- One solution per row
- Each cell colored according to objective value
- No loss of information

Heat maps



Preservation of the dominance relation	Preservation of the front shape	Preservation of the objective range	Preservation of the distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
×	×	✓	×	✓	×	×	✓	✓

Heat maps



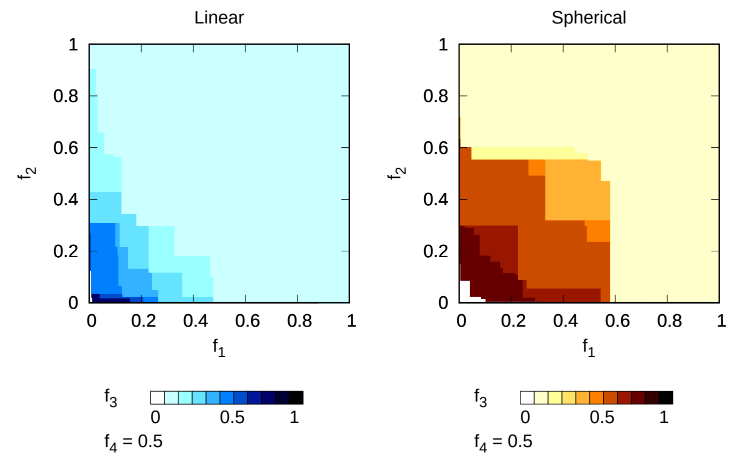
Interactive decision maps

The Edgeworth-Pareto hull (EPH) of an approximation set A contains all points in the objective space that are weakly dominated by any solution in A.

Interactive decision maps

- Visualize the surface of the EPH, not the actual approximation set
- Plot a number of axis-aligned sampling surfaces of the EPH
- Color used to denote third objective
- Fixed value of the forth objective

Interactive decision maps

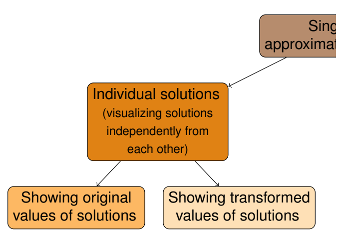


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	≈	✓	≈	✓	✓	×	×	≈

Visualizing single approximation sets

Individual solutions (Visualizing solutions independently from each other) $\rightarrow$ Showing transformed values of solutions

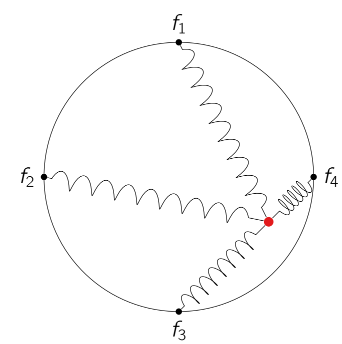
- Radial coordinate visualization [17, 39]
- 3-D Radial coordinate visualization [18], TBA
- Tetrahedron coordinates model [6]
- Polar plots [16], TBA
- Hyper-radial visualization [9]
- Level diagrams [7, 8]
- Prosections [1]



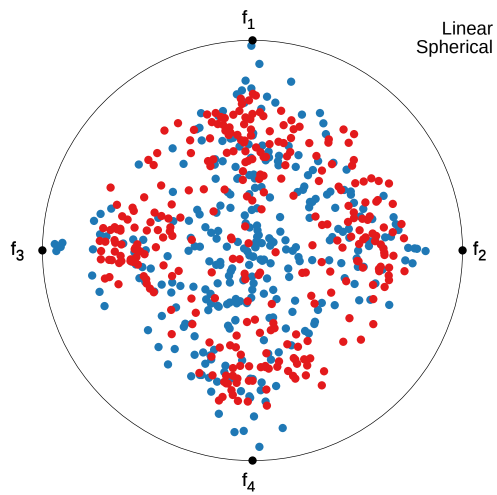
Radial coordinate visualization

Also called RadViz

- Inspired from physics
- Objectives treated as anchors, equally spaced around the circumference of a unit circle
- Solutions attached to anchors with ‘springs’
- Spring stiffness proportional to the objective value
- Solution placed where the spring forces are in equilibrium



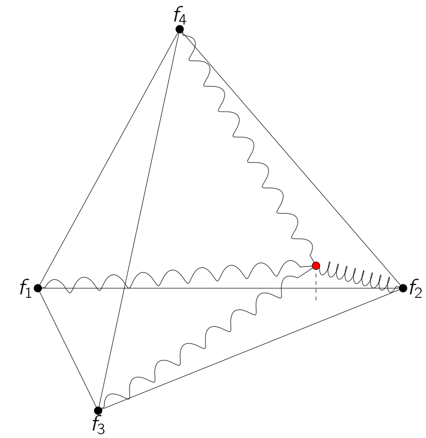
Radial coordinate visualization



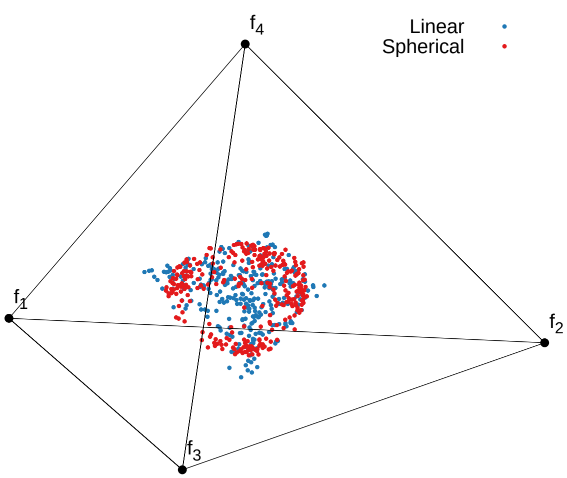
Preservation of the: dominance relation	front shape	objective range	distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
×	×	×	≈	✓	≈	✓	✓	✓

Tetrahedron coordinates model

- Only for four objectives
- Similar to RadViz
- Objectives treated as anchors, placed at the vertices of a regular tetrahedron
- Solutions attached to anchors with ‘springs’
- Spring flexibility proportional to the objective value
- Solution placed where the forces are in equilibrium



Tetrahedron coordinates model

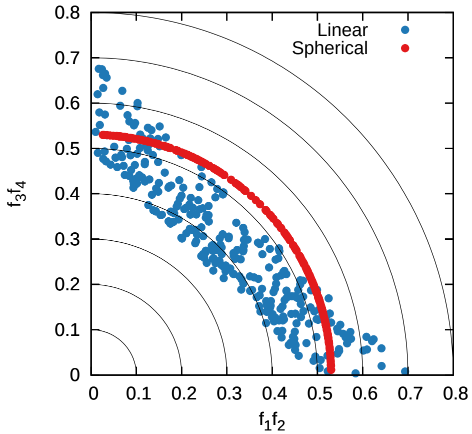


Preservation of the: dominance relation	front shape	objective range	distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
×	×	×	≈	✓	≈	✓	×	✓

Hyper-radial visualization

- Solutions preserve distance (hyper-radius) to the ideal point
- Distances are computed separately for two subsets of objectives
- Indifference curves denote points with the same preference

Hyper-radial visualization

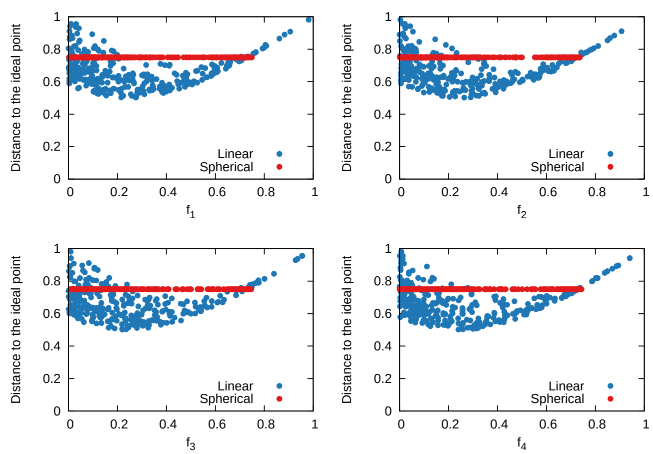


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	≈	✓	×	✓	≈	✓	✓	✓

Level diagrams

- m objectives $\rightarrow$ m diagrams
- Plot solutions with objective f_i on the x axis and distance to the ideal point on the y axis

Level diagrams

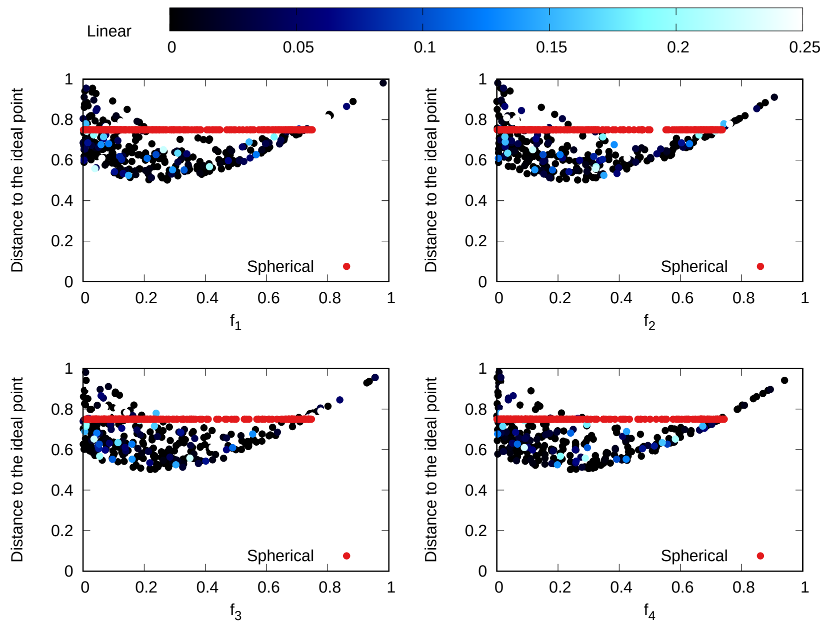


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	≈	✓	×	✓	≈	✓	✓	✓

Level diagrams with asymmetric norm

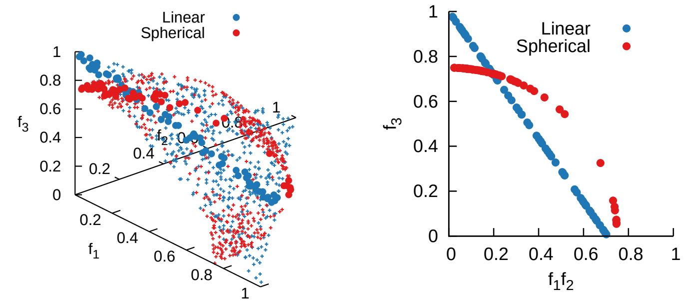
- Compute an asymmetric norm a (very similar to the $I_{\varepsilon+}$ indicator) between any solution and the reference point
  - $a = 0 \Rightarrow$ the solution dominates the reference point
  - $a > 0 \Rightarrow$ the solution needs to be moved by a to dominate the reference point
- Use on our benchmark approximation sets
  - The spherical set is used as the reference set
  - $a = 0 \Rightarrow$ the solution from the linear set dominates a solution from the spherical set
  - $a > 0 \Rightarrow$ the solution from the linear set needs to be moved by at least a to dominate one solution from the spherical set

Level diagrams with asymmetric norm



Prosections

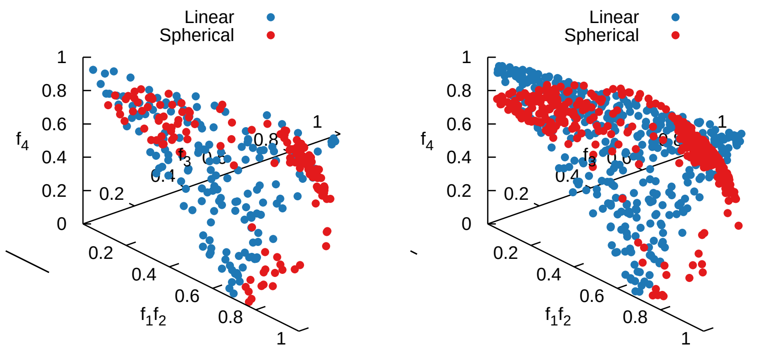
- Visualize only part of the objective space
- Dimensionality reduction by projection of solutions in a section
- Need to choose prosection plane, angle and section width



Before prosection | After prosection

Prosections

300 solutions | 3000 solutions

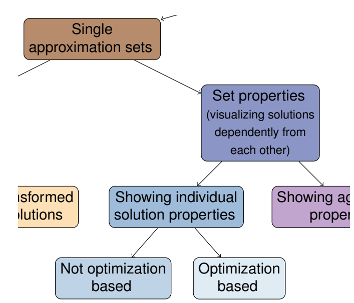


Preservation of the dominance relation	Preservation of the front shape	Preservation of the objective range	Preservation of the distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
✓	✓	≈	✓	✓	✓	✓	×	≈

Visualizing single approximation sets

Set properties (Visualizing solutions dependently from each other) → Showing individual solution-based properties → Not optimization based

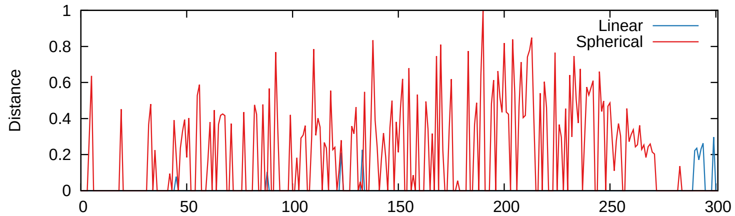
- Distance and distribution charts [5]
- Pareto shells [38]
- Hyper-space diagonal counting [4]
- Treemaps [40], TBA
- Trade-off region maps [31], TBA

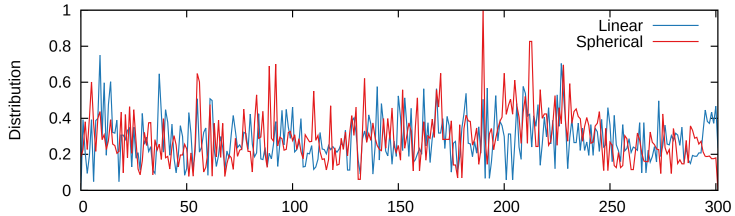


Distance and distribution charts

- Plot solutions against their distance to the Pareto front and distance to other solutions
- Distance chart
  - Plot distance to the nearest non-dominated solution
- Distribution chart
  - Sort solutions w.r.t. first objective
  - Plot distances between consecutive solutions
  - For the first/last solution, compute distance to first/last non-dominated solution
  - k solutions $\rightarrow k+1$ distances
- All distances normalized to $[0, 1]$

Distance and distribution charts



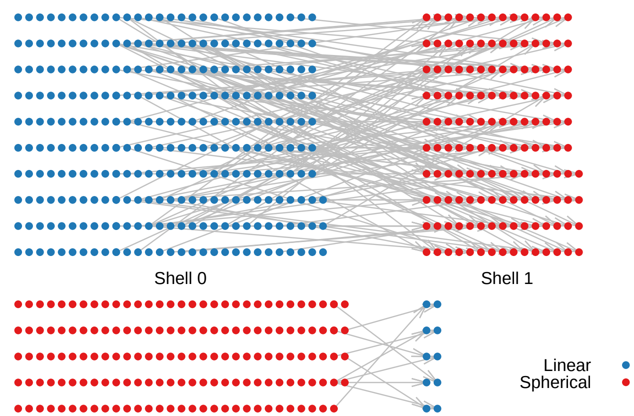


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
≈	×	×	×	✓	×	✓	✓	≈

Pareto shells

- Use nondominated sorting to split solutions to Pareto shells
- Represent solutions in a graph
- Connect dominated solutions to those that dominate them (we show only one arrow per dominated solution)

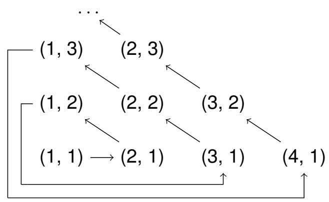
Pareto shells



Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
✓	×	×	×	×	×	✓	✓	✓

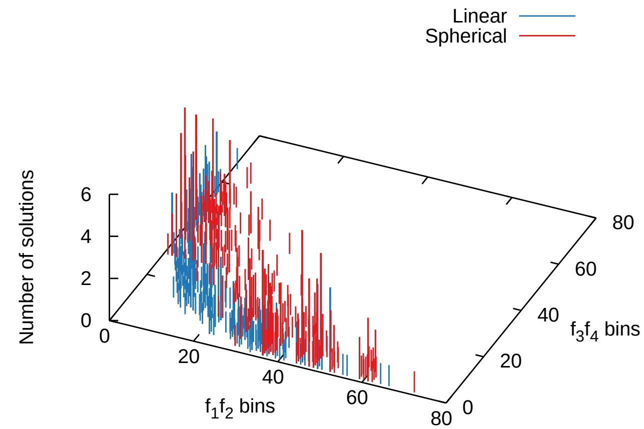
Hyper-space diagonal counting

- Inspired by Cantor's proof that shows $|\mathbb{N}| = |\mathbb{N}^2| = |\mathbb{N}^3| \ldots$



- Discretize each objective (choose a number of bins)
- In the 4-D case
  - Enumerate the bins for objectives f_1 and f_2
  - Enumerate the bins for objectives f_3 and f_4
  - Plot the number of solutions in each pair of bins

Hyper-space diagonal counting

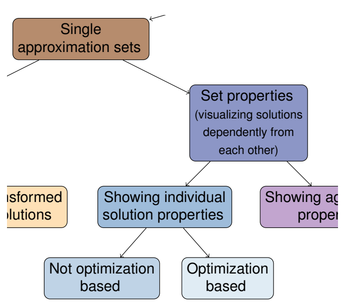


Preservation of the								
dominance relation	front shape	objective range	distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
×	×	×	≈	✓	✓	✓	✓	≈

Visualizing single approximation sets

Set properties (Visualizing solutions dependently from each other) $\rightarrow$ Showing individual solution-based properties $\rightarrow$ Optimization based

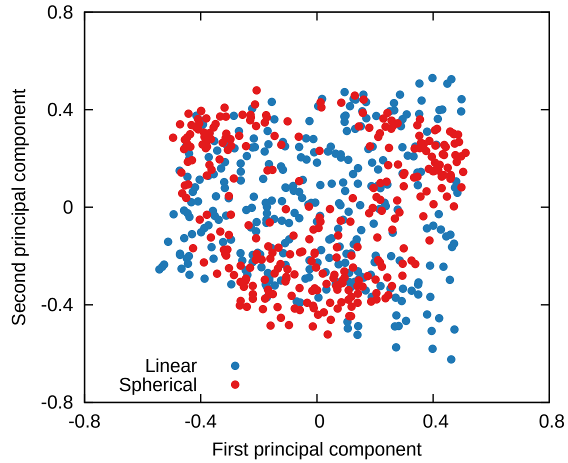
- Principal component analysis [42]
- Sammon mapping [33, 36]
- Neuroscale [27, 11]
- Multidimensional scaling [39]
- Isomap [34, 24]
- Seriated heatmaps [39]
- Two-stage mapping [23]
- Distance-based and dominance-based mappings [12]



Principal component analysis

- Principal components are linear combinations of objectives that maximize variance (and are uncorrelated with already chosen components)
- They are the eigenvectors with the highest eigenvalues of the covariance matrix

Principal component analysis



Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	×	×	×	≈	≈	✓	✓	×

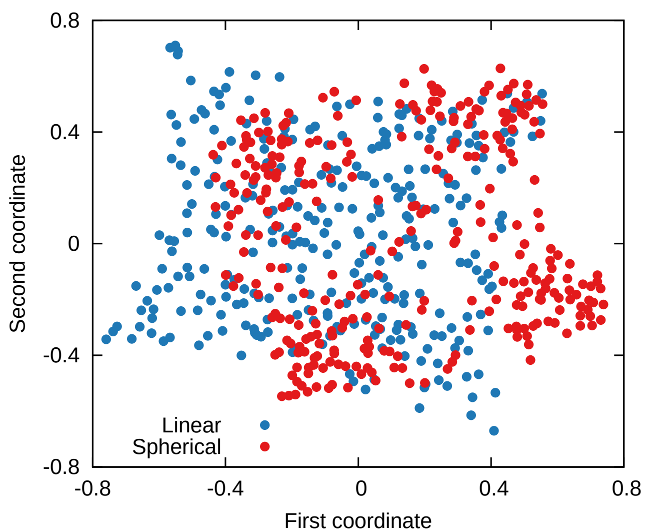
Sammon mapping

- A non-linear mapping
- Aims to preserve distances between solutions
  - d^*_{ij} distance between solutions $\mathbf{x}_i$ and $\mathbf{x}_j$ in the objective space
  - d_{ij} distance between solutions $\mathbf{x}_i$ and $\mathbf{x}_j$ in the visualized space
- Stress function to be minimized

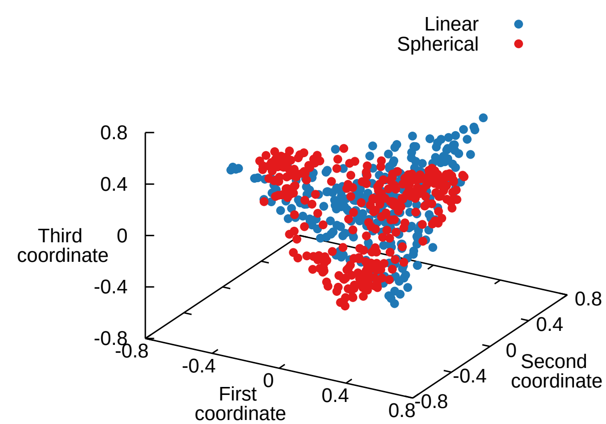
$$S = \sum_{i<j} \frac{(d^*_{ij} - d_{ij})^2}{d_{ij}}$$

- Minimization by gradient descent or other (iterative) methods

Sammon mapping



Sammon mapping

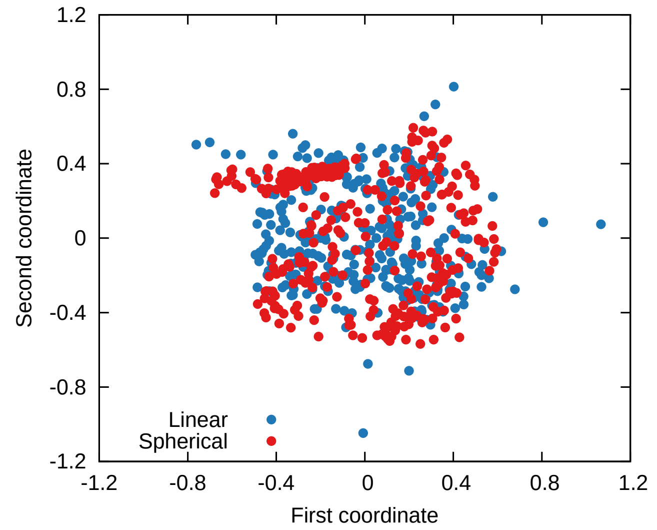


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	×	×	✓	≈	≈	✓	✓	×

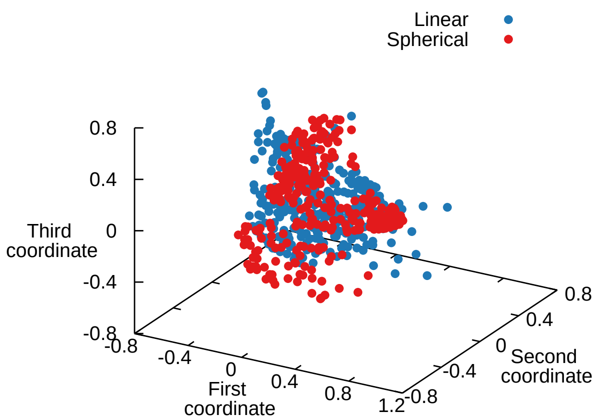
Neuroscale

- A non-linear mapping
- Aims to minimize the same stress function as Sammon mapping
- Uses a radial basis function neural network to model the projection

Neuroscale



Neuroscale



Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	×	×	×	≈	≈	✓	✓	×

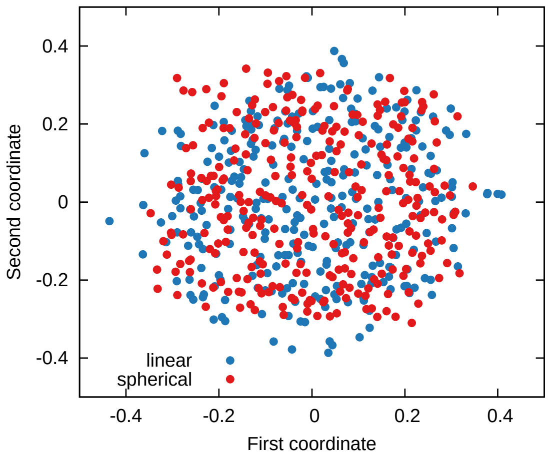
Multidimensional scaling

- Classical multidimensional scaling aims at preserving similarities between solutions
- Here, dominance distance is used to measure similarity
- Two solutions are similar if they share dominance relationships with a third solution

$$
\begin{aligned}
S(\mathbf{a}, \mathbf{b}; \mathbf{z}) = \frac{1}{m} \sum_{i=1}^{m} \big[& I((a_i < z_i) \wedge (b_i < z_i)) \\
& + I((a_i = z_i) \wedge (b_i = z_i)) \\
& + I((a_i > z_i) \wedge (b_i > z_i)) \big]
\end{aligned}
$$

$$
D(\mathbf{a}, \mathbf{b}) = \frac{1}{k-2} \sum_{\mathbf{z} \notin \{\mathbf{a}, \mathbf{b}\}} (1 - S(\mathbf{a}, \mathbf{b}; \mathbf{z}))
$$

Multidimensional scaling

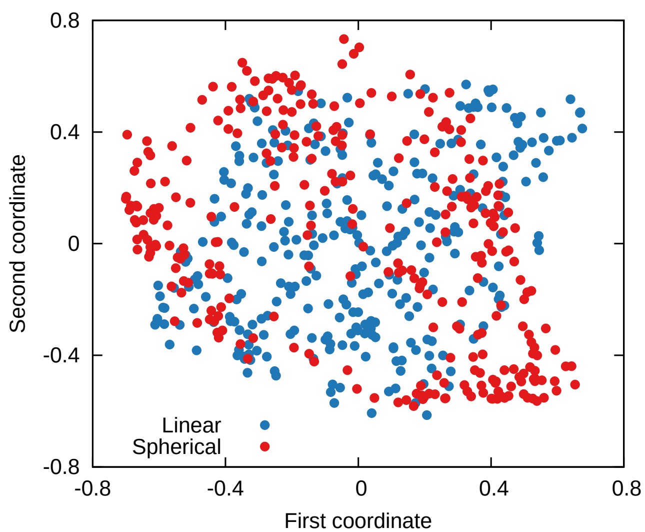


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	×	×	×	×	≈	✓	✓	×

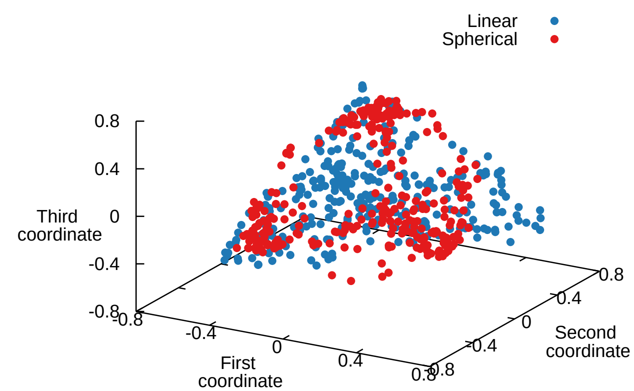
Isomap

- Assumes solutions lie on some low-dimensional manifold and the distances along this manifold should be preserved
- Creates a graph of solutions, where only the neighboring solutions are linked
- The geodesic distance between any two solutions is calculated as the sum of Euclidean distances on the shortest path between the two solutions
- Uses multidimensional scaling to perform the mapping based on these distances

Isomap



Isomap

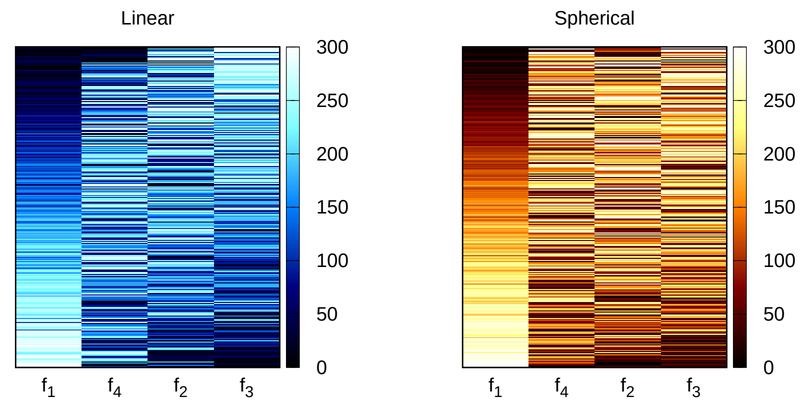


Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	×	×	≈	≈	≈	✓	✓	×

Seriated heatmaps

- Heatmaps with rearranged objectives and solutions
- Similar objectives and similar solutions are placed together
- Ranks are used instead of actual objective values for a more uniform color usage
- Similarity can be computed using
  - Euclidean distance
  - Spearman's footrule
  - Kendall's τ metric

Seriated heatmaps



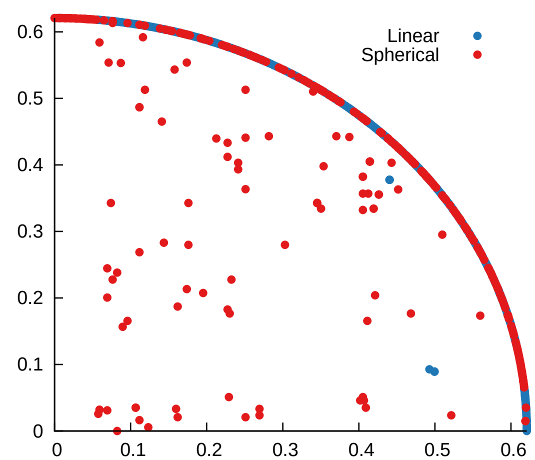
Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
×	×	×	×	≈	×	×	✓	×

Two-stage mapping

Steps

- Split solutions to nondominated and dominated solutions
- Compute r as the average norm of nondominated solutions
- Find a permutation of nondominated solutions that minimizes implicit dominance errors and sum of distances between consecutive solutions
- First stage: distribute nondominated solutions on the circumference of a quarter-circle with radius r in the order of the permutation and with distances proportional to their distances in the objective space
- Second stage: map each dominated solution to the minimal point of all nondominated solutions that dominate it

Two-stage mapping



Preservation of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
dominance relation	front shape	objective range	distribution of solutions					
≈	×	×	×	×	×	✓	≈	×

Distance- and dominance-based mappings

Both mappings

- Use nondominated sorting to split solutions to Pareto shells
- Project solutions onto the circumference of circles (with circle radius proportional to front number)

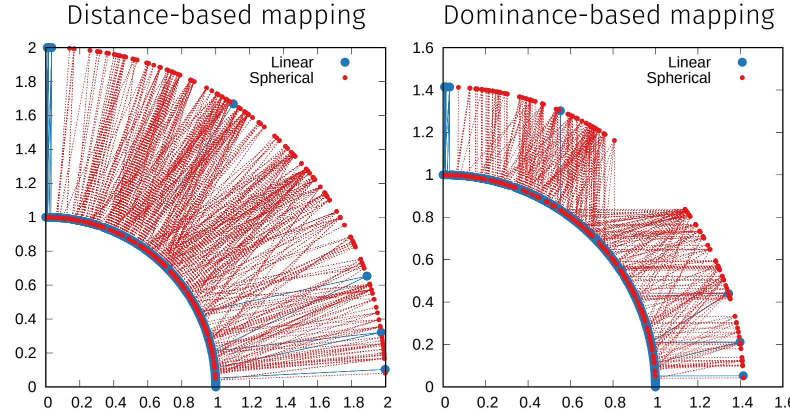
Distance-based mapping

- Tries to preserve closeness of solutions
- Similarity between solutions defined as dominance similarity
- Solution ordering using spectral seriation

Dominance-based mapping

- Aims at preserving dominance relations among solutions
- All $\mathbf{x} \prec \mathbf{y}$ can be shown correctly
- Tries to minimize cases where $\mathbf{x} \not\prec \mathbf{y}$ is not shown correctly

Distance- and dominance-based mappings

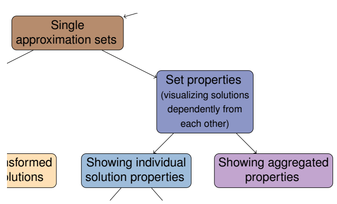


Preservation of the: dominance relation	front shape	objective range	distribution of solutions	Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
× / ✓	×	×	× / ≈	≈	×	✓	✓	×

Visualizing single approximation sets

Set properties (Visualizing solutions dependently from each other)
→ Showing aggregated properties

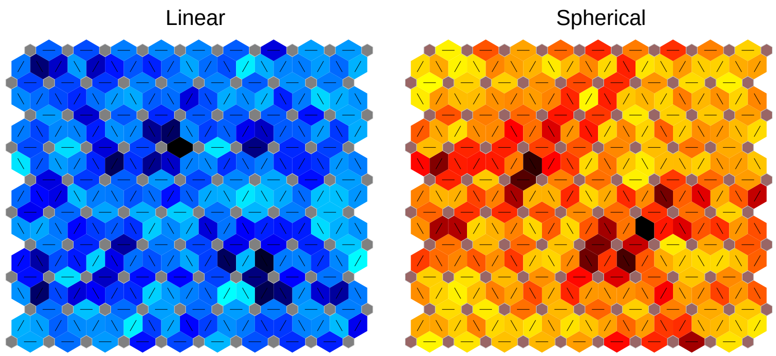
- Self-organizing maps [21, 30]
- Aggregation trees [13]
- MoGrams [35], TBA



Self-organizing maps

- Self-organizing maps (SOMs) are neural networks
- Nearby solutions are mapped to nearby neurons in the SOM
- A SOM can be visualized using the unified distance matrix
- Distance between adjacent neurons is denoted with color
  - Similar neurons → light color
  - Different neurons (cluster boundaries) → dark color

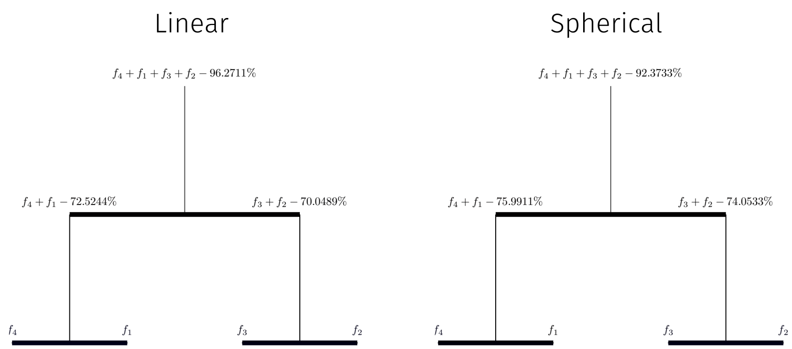
Self-organizing maps



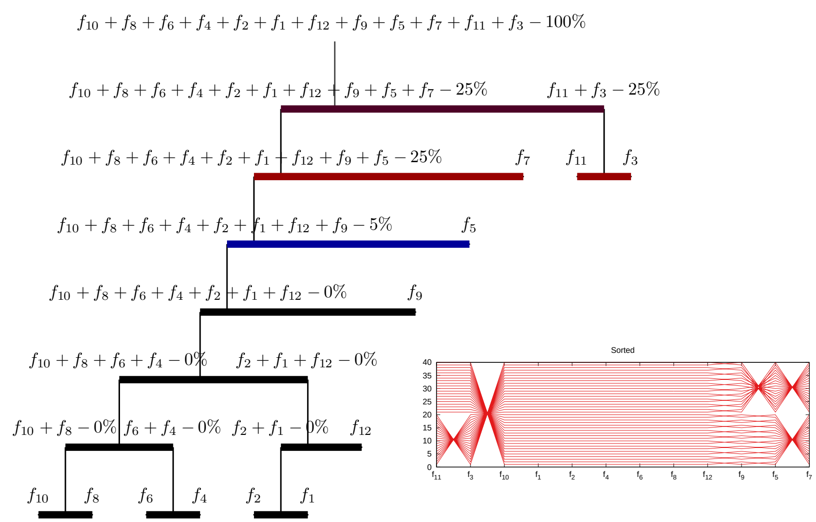
Aggregation trees

- Binary trees that show relationships between objectives
- Iterative clustering of objectives based on their harmony
- Computation of different types of conflict
- Percentages quantify the conflict between objectives
- Colors used to show type of conflict
  - global conflict (black)
  - local conflict on 'good' values (red)
  - local conflict on 'bad' values (blue)
- Can be used to sort objectives in other representations (parallel coordinates, radial charts, heat maps)

Aggregation trees



Aggregation trees



Visualizing repeated approximation sets

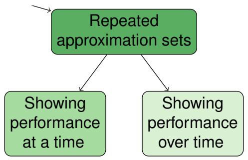
Visualizing repeated approximation sets

Showing performance at a time

- Empirical Attainment Function (EAF) [15]

Showing performance over time

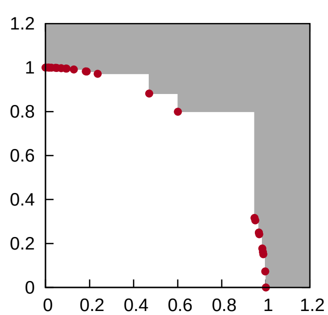
- Average Runtime Attainment Function (aRTA) [3]



Empirical attainment function

Goal-attainment

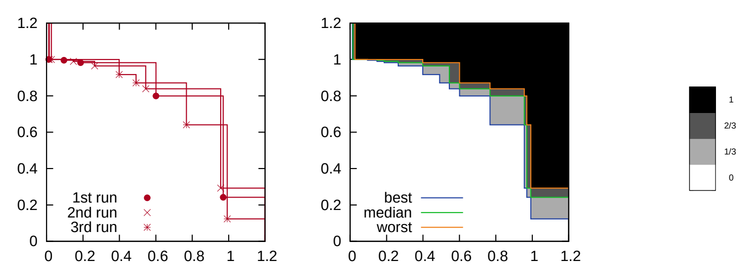
- Approximation set A
- A point in the objective space $\mathbf{z}$ is attained by A when $\mathbf{z}$ is weakly dominated by at least one solution from A



Empirical attainment function

EAF values [15]

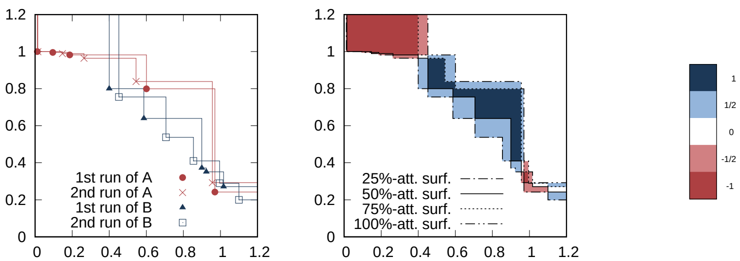
- Algorithm $\mathcal{A}$, approximation sets $A_1, A_2, \ldots, A_r$
- EAF of $\mathbf{z}$ is the frequency of attaining $\mathbf{z}$ by $A_1, A_2, \ldots, A_r$
- Summary (or $k\%$-) attainment surfaces



Empirical attainment function

Differences in EAF values [25]

- Algorithm $\mathcal{A}$, approximation sets $A_1, A_2, \ldots, A_r$
- Algorithm $\mathcal{B}$, approximation sets $B_1, B_2, \ldots, B_r$
- Visualize differences between EAF values



Visualization of 3-D EAF

Need to compute and visualize a large number (over 10 000) of points/cuboids

Exact case

- Attainment surfaces: Visualization of facets
- EAF values: Slicing [2]
- EAF differences: Slicing, Maximum intensity projection [41, 2]

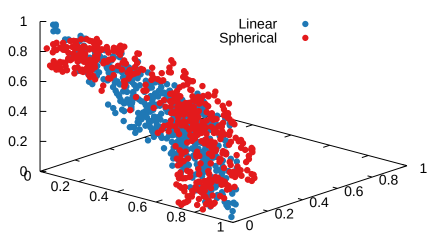
Approximated case

- Attainment surfaces: Grid-based sampling [20]
- EAF values: Slicing, Direct volume rendering [10, 2]
- EAF differences: Slicing, Maximum intensity projection, Direct volume rendering

Benchmark approximation sets

Sets of approximation sets

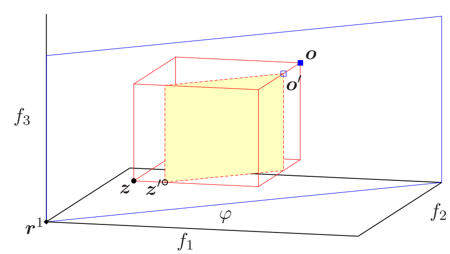
- 5 linear approximation sets with a uniform distribution of solutions (100 solutions in each)
- 5 spherical approximation sets with a nonuniform distribution of solutions (100 solutions in each)



Exact 3-D EAF values and differences

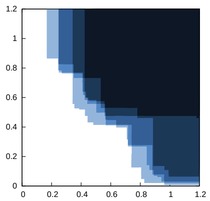
Slicing

- Visualize cuboids intersecting the slicing plane
- Need to choose coordinate and angle

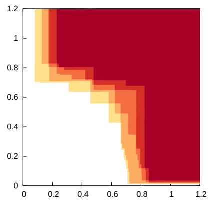


Exact 3-D EAF values and differences

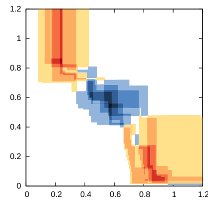
Slicing



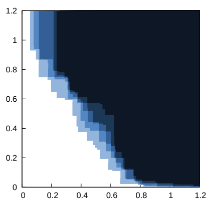
Slice of Lin at $\varphi = 5^\circ$



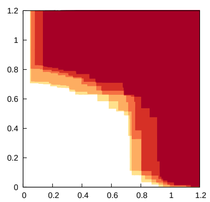
Slice of Sph at $\varphi = 5^\circ$



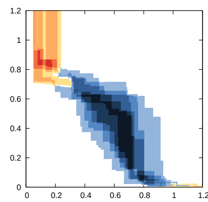
Slice of Lin-Sph and Sph-Lin at $\varphi = 5^\circ$



Slice of Lin at $\varphi = 45^\circ$



Slice of Sph at $\varphi = 45^\circ$

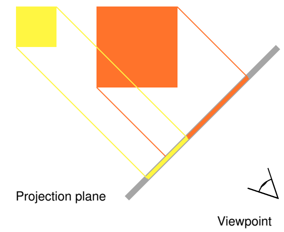


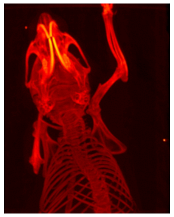
Slice of Lin-Sph and Sph-Lin at $\varphi = 45^\circ$

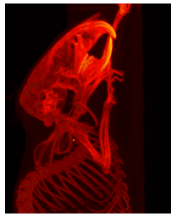
Exact 3-D EAF differences

Maximum intensity projection

- Volume rendering method for spatial data represented by voxels
- Simple and efficient
- No sense of depth, cannot distinguish between front and back



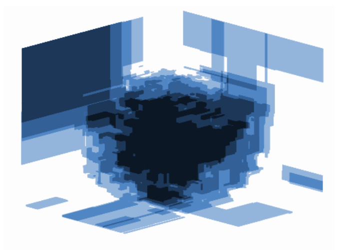


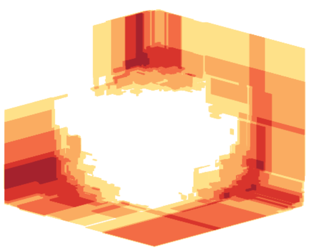


Exact 3-D EAF differences

Maximum intensity projection

- Suitable for visualizing EAF differences (focus on large differences)
- Sorting w.r.t. EAF differences (smaller to larger)
- Plot on top of previous ones





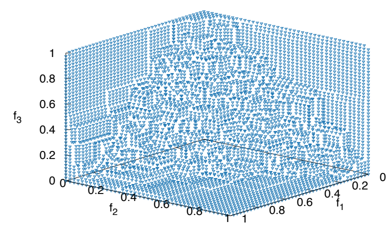
Approximated attainment surfaces

Grid-based sampling

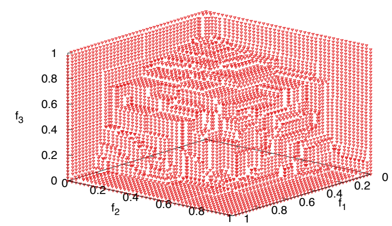
Repeat for all $f_i f_j$, $i < j$ (i.e. $f_1 f_2$, $f_1 f_3$ and $f_2 f_3$) :

- Construct a $k \times k$ grid on the plane $f_i f_j$
- Compute intersections between the attainment surface and the axis-aligned lines on the grid

Median attainment surfaces



Linear



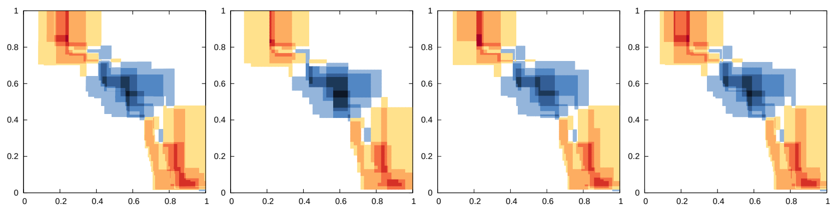
Spherical

Approximated EAF values and differences

Discretization into voxels

- Discretization of cuboids
- Discretization from the space of EAF values/differences

Slicing

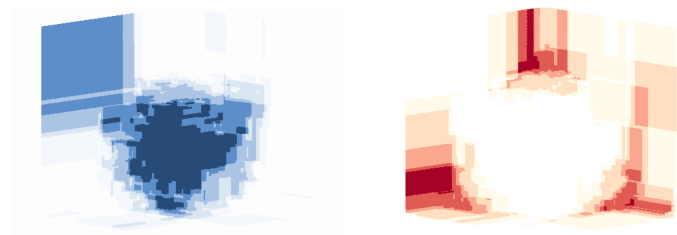


Exact | 64^3 voxels | 128^3 voxels | 256^3 voxels

Approximated 3-D EAF differences

Maximum intensity projection

- Plots produced using Voreen [28, 37]
- Some loss of information



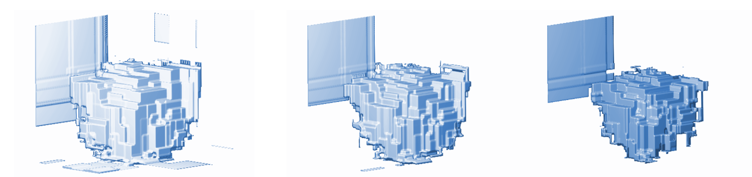
Approximated 3-D EAF values and differences

Direct volume rendering

- Volume rendering method for spatial data represented by voxels
- A transfer function assigns color and opacity to voxel values
- Enables to see "inside the volume"
- Requires the definition of the transfer function

Approximated 3-D EAF differences

Direct volume rendering of Lin-Sph

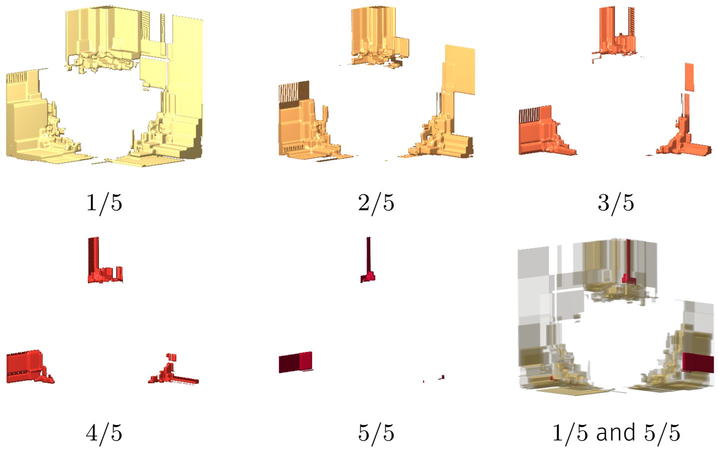


$1/5$ | $2/5$ | $3/5$

$4/5$ | $5/5$ | $1/5$ and $5/5$

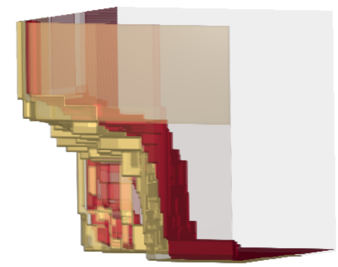
Approximated 3-D EAF differences

Direct volume rendering of Sph-Lin



Approximated 3-D EAF values

Direct volume rendering of Sph



1/5 and 5/5

Average Runtime Attainment Function

aRTA value

- Algorithm $\mathcal{A}$ run r times
- All solutions that are nondominated at creation are recorded
- aRTA($\mathbf{z}$) is the average number of evaluations needed to attain $\mathbf{z}$

aRTA ratio

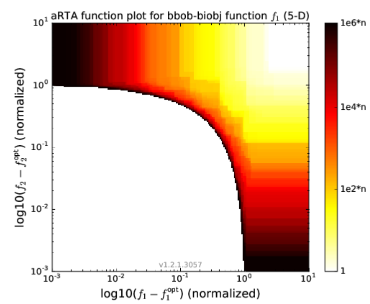
- Algorithms $\mathcal{A}$ and $\mathcal{B}$
- Compute ratio between aRTA($\mathbf{z}$) values for $\mathcal{A}$ and $\mathcal{B}$

Visualization using **grid-based sampling** [3]

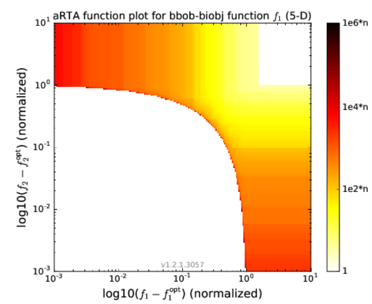
Approximated aRTA values

Two algorithms on the sphere-sphere problem [3]

Algorithm $\mathcal{A}$

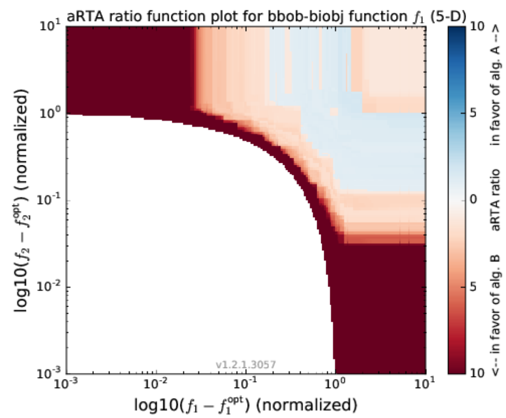


Algorithm $\mathcal{B}$



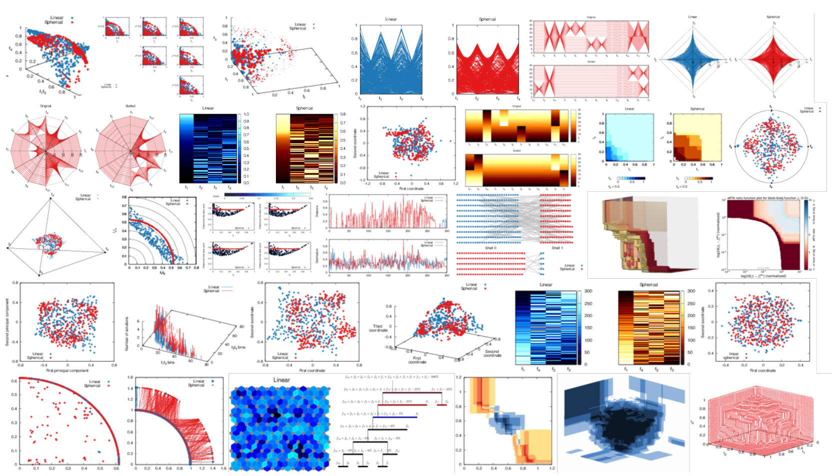
Approximated aRTA ratios

aRTA ratio between Algorithms $\mathcal{A}$ and $\mathcal{B}$ [3]

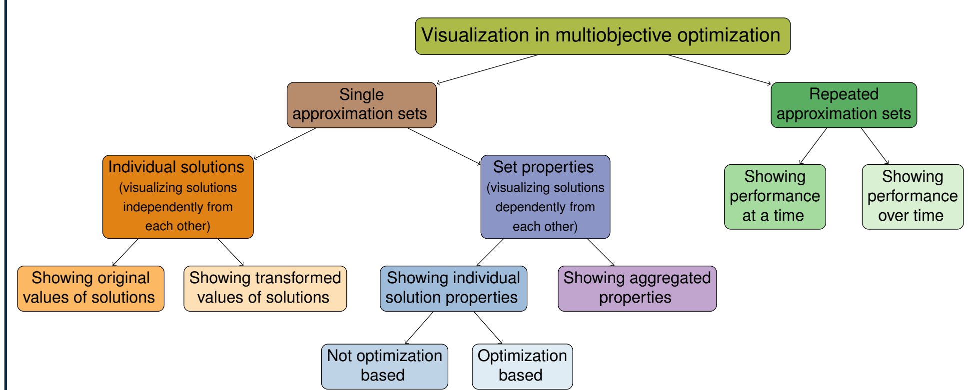


Summary

Summary

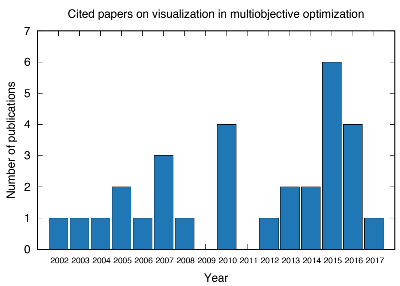


Summary



Summary

- Visualization in multiobjective optimization useful for various purposes
- Customized methods are needed to address the peculiarities of approximation set visualization
- Many new approaches in the last years



Index

Methods for visualizing single approximation sets (page)

Index

Methods for visualizing repeated approximation sets (page)

Acknowledgement

The authors acknowledge the financial support from the Slovenian Research Agency (research core funding No. P2-0209 and project No. Z2-8177 *Incorporating real-world problems into the benchmarking of multiobjective optimizers*).

This work is part of a project that has received funding from the *European Union's Horizon 2020 research and innovation program* under grant agreement No. 692286.

SYNERGY
Synergy for Smart Multi-Objective Optimization
`www.synergy-twinning.eu`

References

References i

[1] T. Tušar and B. Filipič.
Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosection method.
IEEE Transactions on Evolutionary Computation, 19(2):225-245, 2015.

[2] T. Tušar and B. Filipič.
Visualizing exact and approximated 3D empirical attainment functions.
Mathematical Problems in Engineering, Article ID 569346, 18 pages, 2014.

[3] D. Brockhoff, A. Auger, N. Hansen and T. Tušar.
Quantitative performance assessment of multiobjective optimizers: The average runtime attainment function.
EMO 2017, pages 103–119, 2017.

References ii

[4] G. Agrawal, C. L. Bloebaum, and K. Lewis.
Intuitive design selection using visualized n-dimensional Pareto frontier.
American Institute of Aeronautics and Astronautics, 2005.

[5] K. H. Ang, G. Chong, and Y. Li.
Visualization technique for analyzing nondominated set comparison.
SEAL '02, pages 36–40, 2002.

[6] X. Bi and B. Li.
The visualization decision-making model of four objectives based on the balance of space vector.
IHMSC 2012, pages 365–368, 2014.

[7] X. Blasco, J. M. Herrero, J. Sanchis, and M. Martínez.
A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization.
Information Sciences, 178(20):3908–3924, 2008.

References iii

[8] X. Blasco, G. Reynoso-Mezab, E. A. Sanchez Perez, and J. V. Sanchez Perez.
Asymmetric distances to improve n-dimensional Pareto fronts graphical analysis.
Information Sciences, 340-341:228–249, 2016.

[9] P.-W. Chiu and C. Bloebaum.
Hyper-radial visualization (HRV) method with range-based preferences for multi-objective decision making.
Structural and Multidisciplinary Optimization, 40(1–6):97–115, 2010.

[10] K. Engel, M. Hadwiger, J. M. Kniss, C. Rezk-Salama, and D. Weiskopf.
Real-time Volume Graphics.
A. K. Peters, Natick, MA, USA, 2006.

References iv

[11] R. M. Everson and J. E. Fieldsend.
Multi-class ROC analysis from a multi-objective optimisation perspective.
Pattern Recognition Letters, 27(8):918–927, 2006.

[12] J. E. Fieldsend and R. M. Everson.
Visualising high-dimensional Pareto relationships in two-dimensional scatterplots.
EMO 2013, pages 558–572, 2013.

[13] A. R. R. de Freitas, P. J. Fleming, and F. G. Guimaraes.
Aggregation trees for visualization and dimension reduction in many-objective optimization.
Information Sciences, 298:288–314, 2015.

References v

[14] S. Greco, K. Klamroth, J. D. Knowles, and G. Rudolph.
Understanding complexity in multiobjective optimization (Dagstuhl seminar 15031).
Dagstuhl Reports, pages 96–163, 2015.

[15] V. D. Grunert da Fonseca, C. M. Fonseca, and A. O. Hall.
Inferential performance assessment of stochastic optimisers and the attainment function.
EMO 2001, pages 213–225, 2001.

[16] Z. He and G. G. Yen.
Visualization and performance metric in many-objective optimization.
IEEE Transactions on Evolutionary Computation, 20(3):386–402, 2016.

[17] P. E. Hoffman, G. G. Grinstein, K. Marx, I. Grosse, and E. Stanley.
DNA visual and analytic data mining.
Conference on Visualization, pages 437–441, 1997.

References vi

[18] A. Ibrahim, S. Rahnamayan, M. V. Martin, K. Deb.
3D-RadVis: Visualization of Pareto front in many-objective optimization
CEC 2016, pages 736–745, 2016.

[19] A. Inselberg.
Parallel Coordinates: Visual Multidimensional Geometry and its Applications.
Springer, New York, NY, USA, 2009.

[20] J. Knowles.
A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers.
ISDA '05, pages 552–557, 2005.

[21] T. Kohonen.
Self-Organizing Maps.
Springer Series in Information Sciences, 2001.

References vii

[22] R. H. Koochaksaraei, R. Enayatifar, and F. G. Guimaraes.
A new visualization tool in many-objective optimization problems.
HAIS 2016, pages 213–224, 2016.

[23] M. Köppen and K. Yoshida.
Visualization of Pareto-sets in evolutionary multi-objective optimization.
HIS 2007, pages 156–161, 2007.

[24] F. Kudo and T. Yoshikawa.
Knowledge extraction in multi-objective optimization problem based on visualization of Pareto solutions.
CEC 2012, 6 pages, 2012.

[25] M. López-Ibáñez, L. Paquete, and T. Stützle.
Exploratory analysis of stochastic local search algorithms in biobjective optimization.
Experimental Methods for the Analysis of Optimization Algorithms, pages 209–222, 2010.

[26] A. V. Lotov, V. A. Bushenkov, and G. K. Kamenev.
***Interactive Decision Maps: Approximation and Visualization of Pareto Frontier*.**
Kluwer Academic Publishers, Boston, MA, USA, 2004.

[27] D. Lowe and M. E. Tipping.
Feed-forward neural networks and topographic mappings for exploratory data analysis.
Neural Computing & Applications, 4(2):83–95, 1996.

[28] J. Meyer-Spradow, T. Ropinski, J. Mensmann, and K. H. Hinrichs.
Voreen: A rapid-prototyping environment for ray-casting-based volume visualizations.
IEEE Computer Graphics and Applications, 29(6):6–13, 2009.

[29] K. Miettinen.
Survey of methods to visualize alternatives in multiple criteria decision making problems.
OR Spectrum, 36(1):3–37, 2014.

[30] S. Obayashi and D. Sasaki.
Visualization and data mining of Pareto solutions using self-organizing map.
EMO 2003, pages 796–809, 2003.

[31] R. L. Pinheiro, D. Landa-Silva, and J. Atkin.
Analysis of objectives relationships in multiobjective problems using trade-off region maps.
GECCO 2015, pages 735–742, 2015.

[32] A. Pryke, S. Mostaghim, and A. Nazemi.
Heatmap visualisation of population based multiobjective algorithms.
EMO 2007, pages 361–375, 2007.

[33] J. W. Sammon.
A nonlinear mapping for data structure analysis.
IEEE Transactions on Computers, C-18(5):401–409, 1969.

[34] J. B. Tenenbaum, V. de Silva, and J. C. Langford.
A global geometric framework for nonlinear dimensionality reduction.
Science, 290(5500):2319–2323, 2000.

[35] K. Trawinski, M. Chica, D. P. Pancho, S. Damas, and O. Cordon.
moGrams: A network-based methodology for visualizing the set of non-dominated solutions in multiobjective optimization.
CoRR abs/1511.08178, 2015.

[36] J. Valdes and A. Barton.
Visualizing high dimensional objective spaces for multiobjective optimization: A virtual reality approach.
CEC 2007, pages 4199--4206), 2007.

[37] Voreen, Volume rendering engine.
`http://www.voreen.org/`

[38] D. J. Walker, R. M. Everson, and J. E. Fieldsend.
Visualisation and ordering of many-objective populations.
CEC 2010, 8 pages, 2010.

[39] D. J. Walker, R. M. Everson, and J. E. Fieldsend.
Visualizing mutually nondominating solution sets in many-objective optimization.
IEEE Transactions on Evolutionary Computation, 17(2):165–184, 2013.

[40] D. J. Walker.
Visualising multi-objective populations with treemaps.
GECCO 2015, pages 963–970, 2015.

[41] J. W. Wallis, T. R. Miller, C. A. Lerner, and E. C. Kleerup.
Three-dimensional display in nuclear medicine.
IEEE Transactions on Medical Imaging, 8(4):297–230, 1989.

[42] M. Yamamoto, T. Yoshikawa, and T. Furuhashi.
Study on effect of MOGA with interactive island model using visualization.
CEC 2010, 6 pages, 2010.