Interactive Problem Solving via Algorithm Visualization

Pearl Pu Faltings

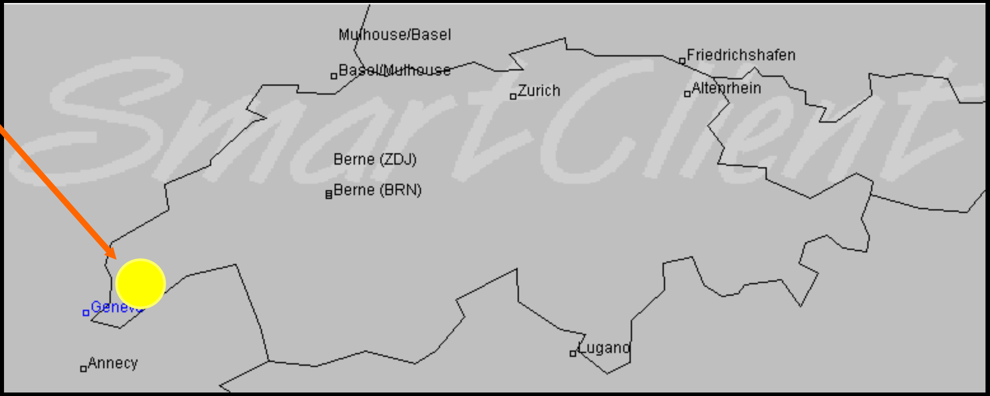
Ergonomics & Database Lab

Ecole Polytechnique Fédérale de Lausanne

Denis Lalanne

LIA-CERI, Université d'Avignon

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE



Motivation

- powerful algorithms fail to explain the results
- problems difficult for both human and machine

Problem statement

- interactive problem solving using algorithm visualization - a case study in conceptual design with constraint satisfaction techniques

Outline of talk

- conceptual design
- constraint satisfaction algorithms
- visualization metaphors
  - kaleidoscope
  - MAP
  - Lattice
- Related works
- Conclusion

Conceptual design

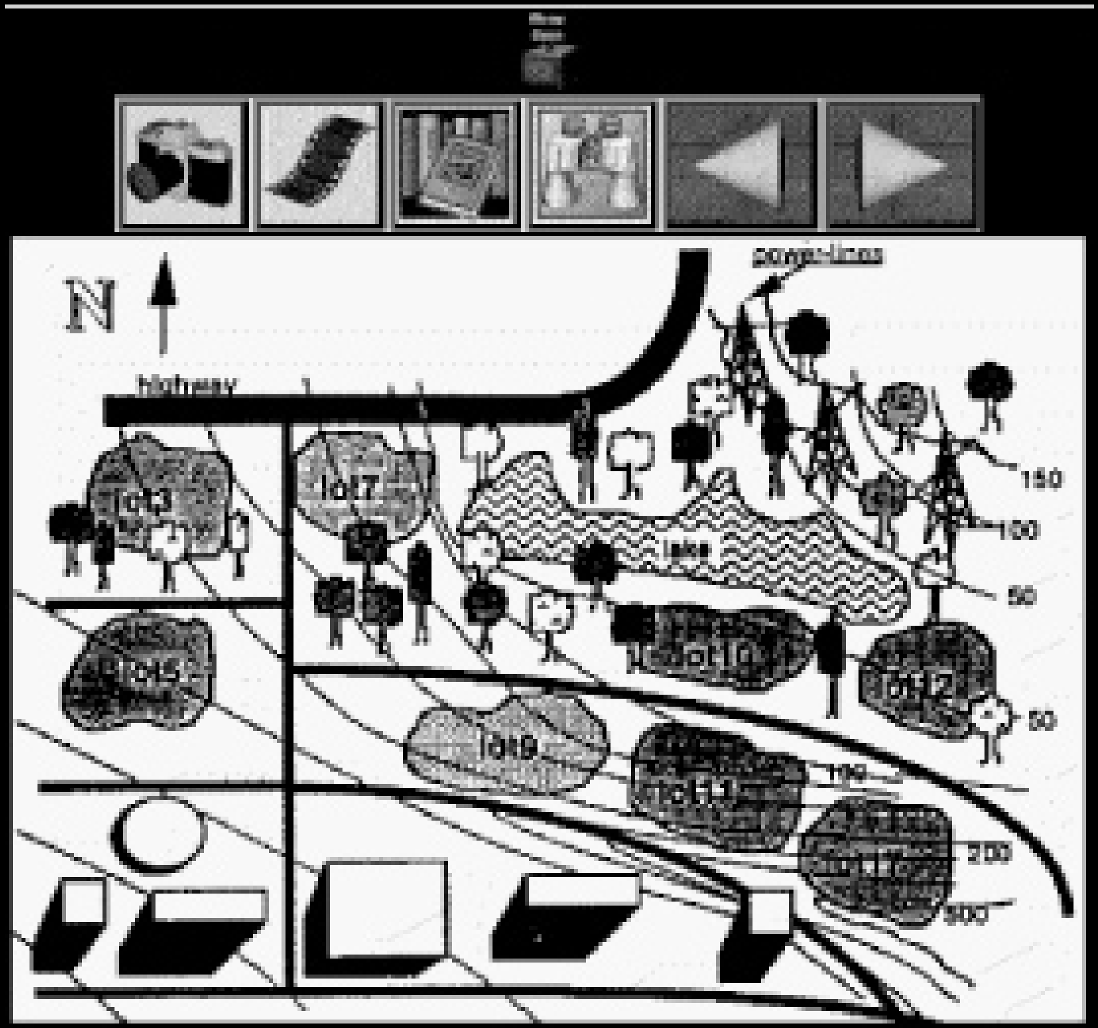
- “Design is not description of what is, it is exploration of what might be” -- Bill Mitchell
- Computational approximation -- define search space, automatic search, constraint satisfaction
- Product conceptual design, configuration design, land use design….

Constraint satisfaction problem (CSP)

- a set of variables
- each variable has a domain - a set of permissible values
- a set of constraints
- simple examples: map coloring, n-queen

Design example defined as CSP

land use



Lot3, lot5, lot7 and lot9 are all relatively flat sites with fairly good soil conditions. Lot10 and lot12 are moderately sloped sites in a nice wooded location, but have poor soil conditions. Lot17 is a very steep site. Lot11 ant lot17 are elevated sites facing southwest and down into a valley that has a lake and some wooded area.

The problem solver's task is to come up with assignments of land uses to sites. A complete design is one in which each land use has been assigned to a lot. The final design should be one which complies with a given set of criteria.

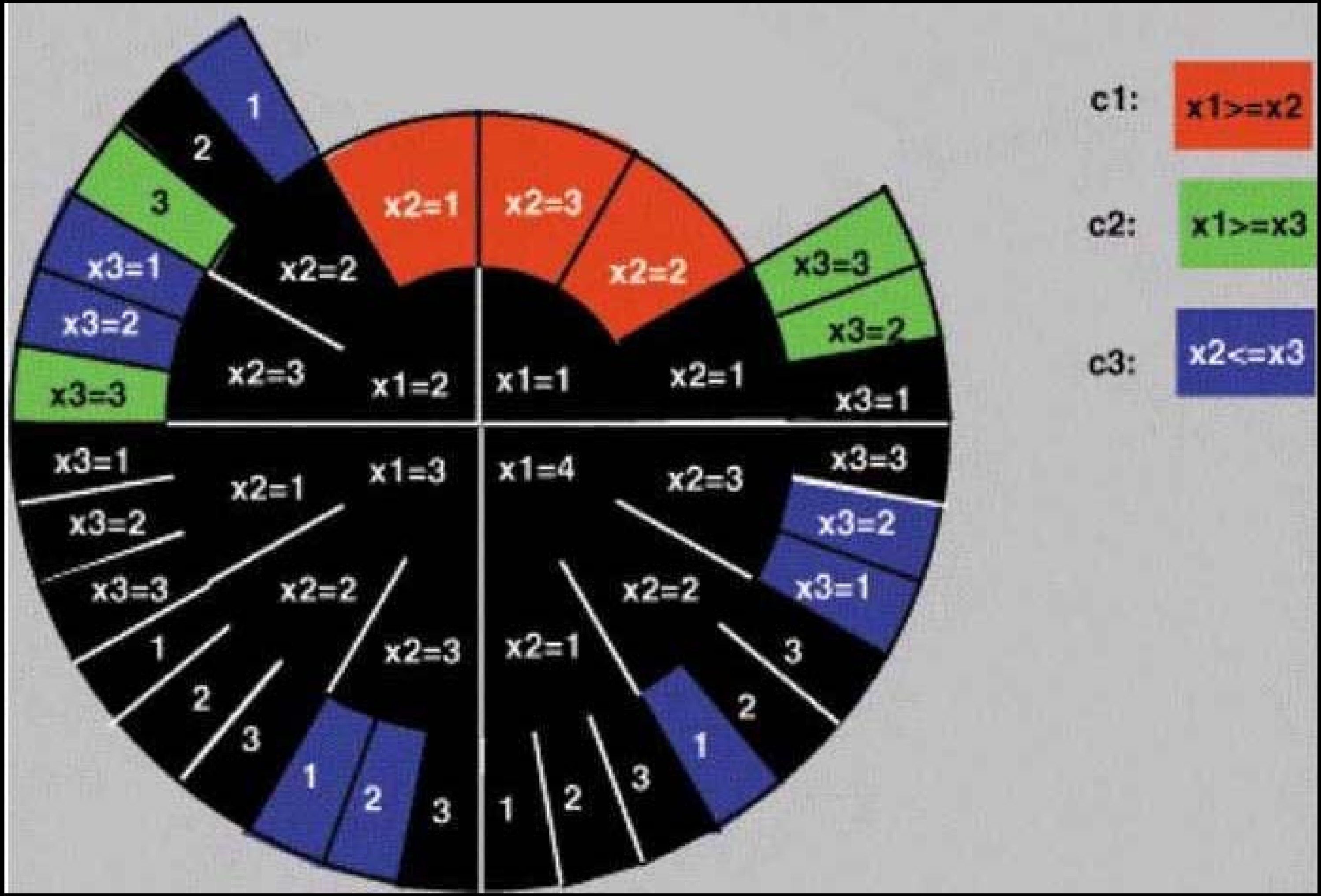
- The dumpsite and the cemetery should not be visible from either houses, nor apartment building.
- Steep slopes are to be avoided for building.
- Poor soil should be avoided for those land uses that involve
- construction.
- The recreational area has to be near the lake.
- The highway is noisy and ugly and should be avoided when locating the apartments, the single-family housing complex and the recreational areas.
- The supermarket cannot be in front of the single-family houses, the dumpsite, and of the cemetery, mainly for esthetical reason.

((dumpsite == 3) or (cemetery == 3)) -> ((apartment != 5) and (apartment != 7))

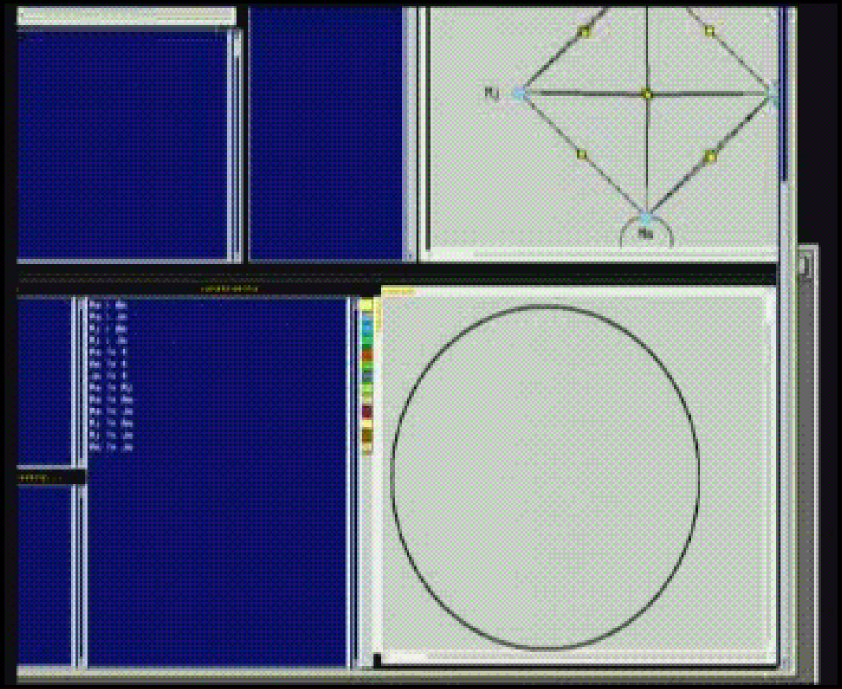
Search algorithms for CSP

- simple backtracking
- pre-processing and SB
- Monte Carlo method by Knuth
- algorithms are np-complete in general

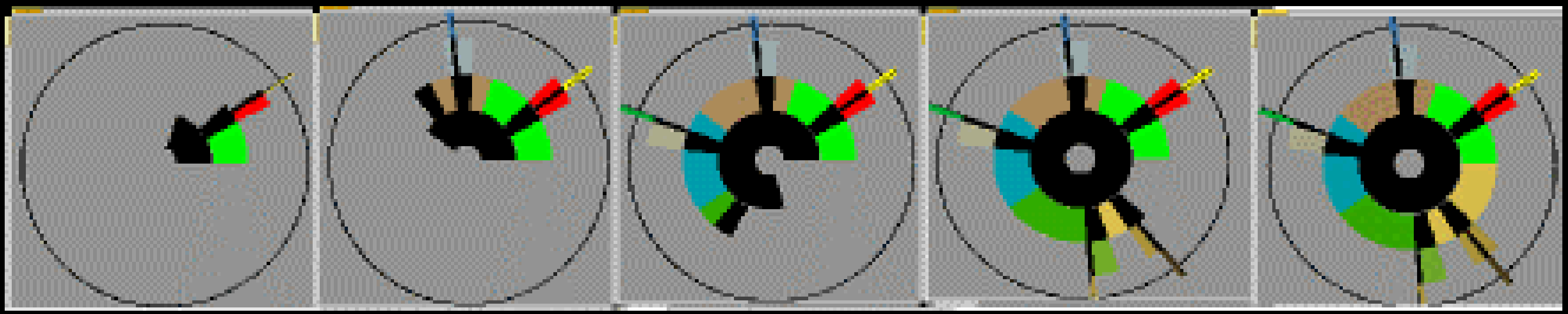
Kaleidoscope – visualizing search



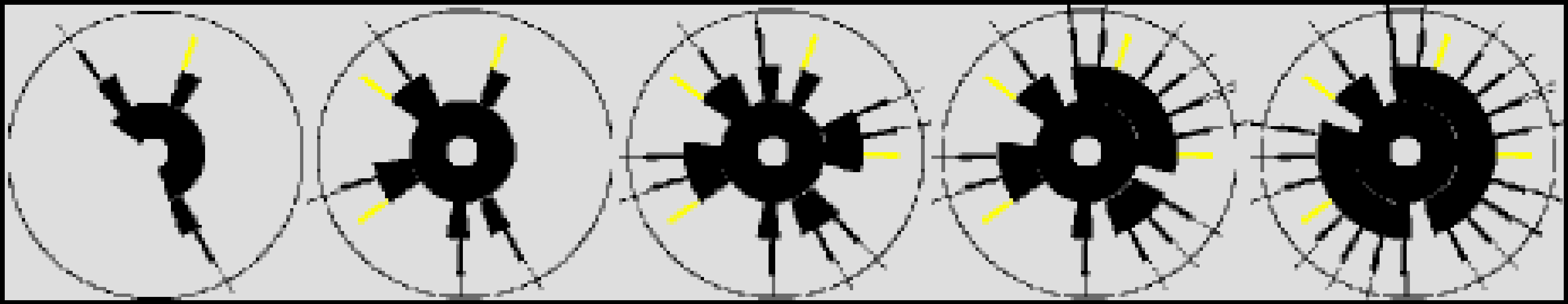
Constraint Editor



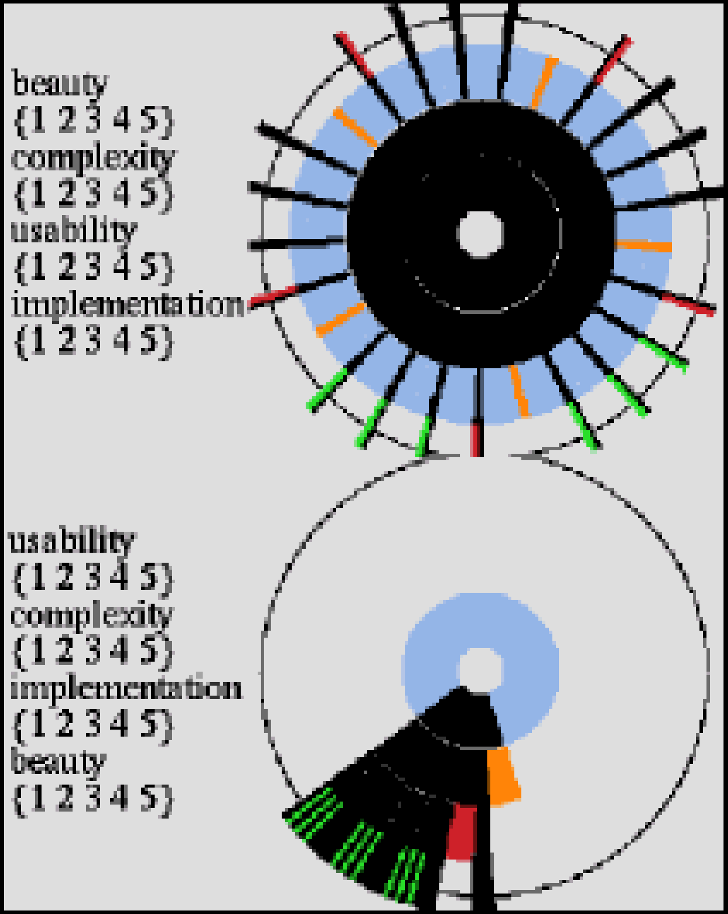
Visualizing simple backtracking



Visualizing Knuth algorithm



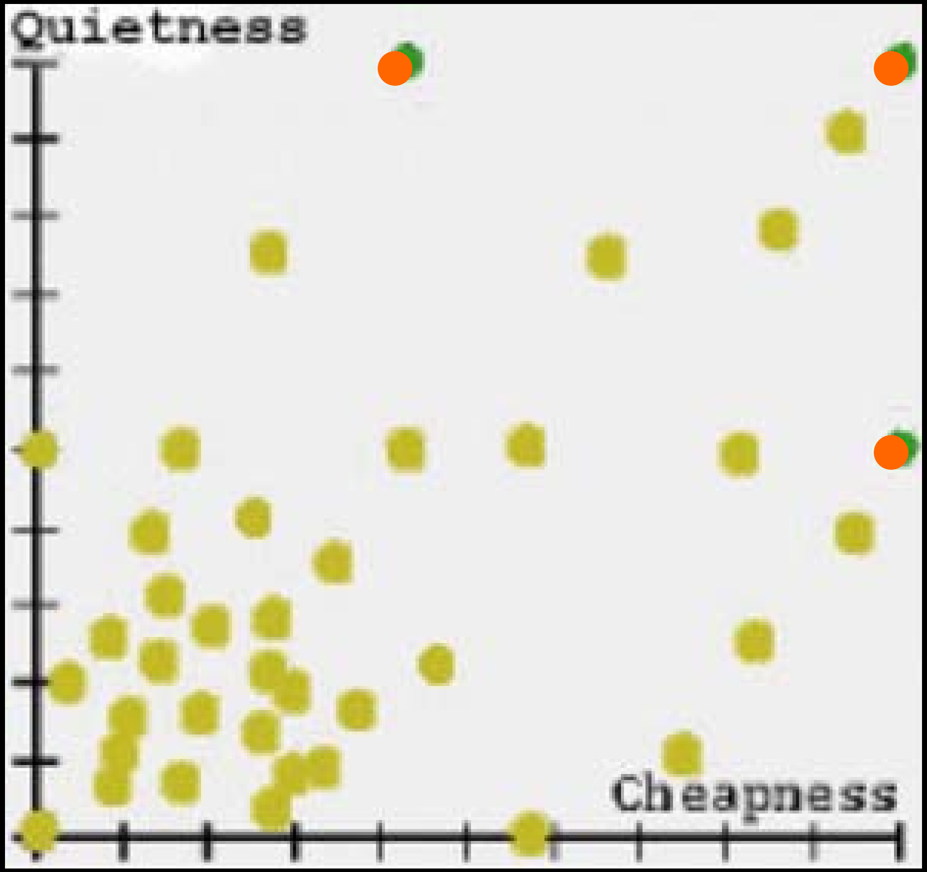
Visualizing variable re-ordering



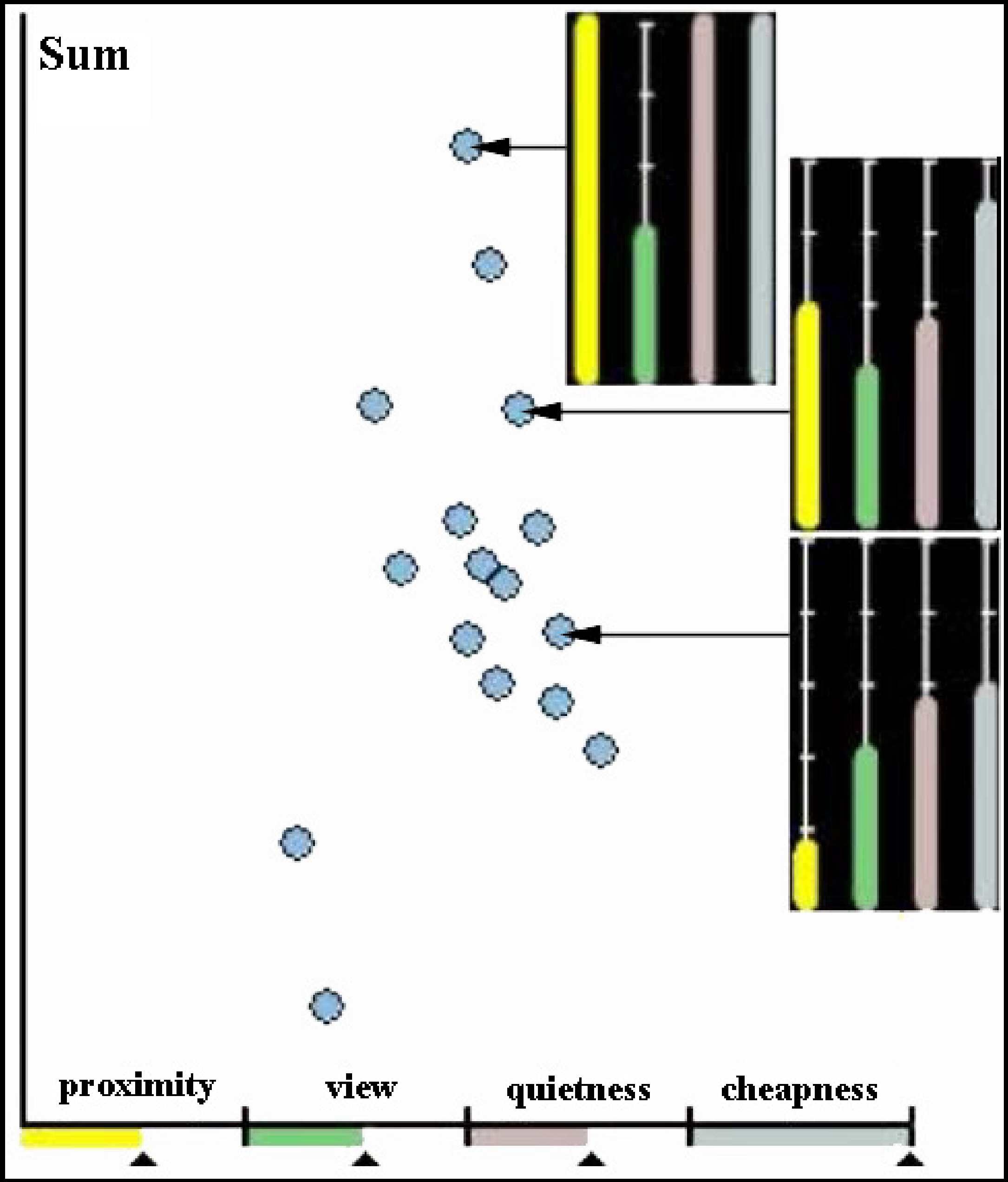
Discovery with Kaleidoscope

- does thrashing occur, frequently?
- Are solutions diversified or concentrated in clusters?
- Are solutions abundant or futile
- if variables re-ordered, does solution generation become faster?

Search in under-constrained spaces



Visualizing land assignment problem



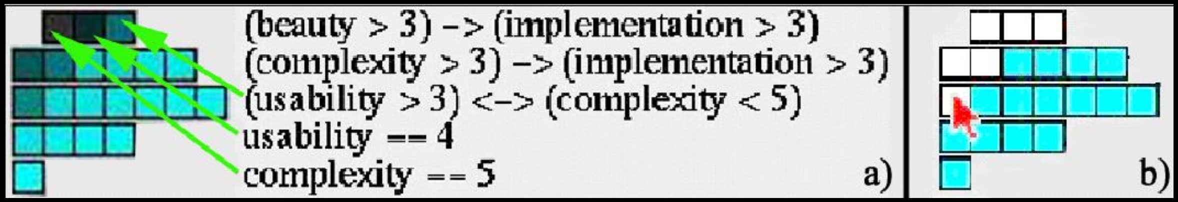
Visualizing tradeoffs of solutions in MAP: multiple attribute Pareto

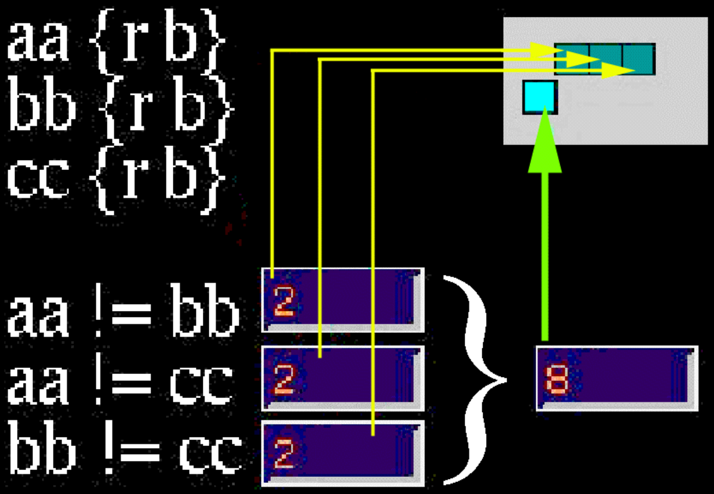
Search in over-constrained space

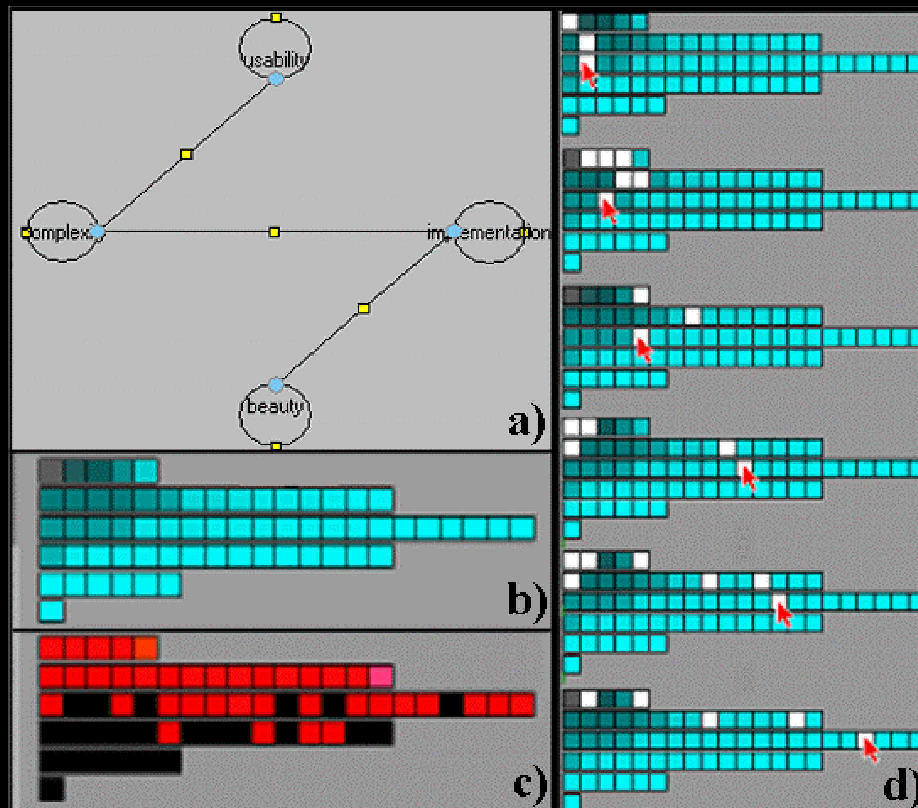
- one or several sets of constraints contain no solution
- diagnosing them is hard without visualization
- Lattice visualization

Minimal conflict set(s)

- A constraint set is a conflict set if it does not allow any partial solutions
- a constraint set is the minimal conflict set if no smaller set is a constraint set
- a constraint set cannot allow any solutions iff it contains at least one minimal conflict set


(beauty > 3) -> (implementation > 3)
(complexity > 3) -> (implementation > 3)
(usability > 3) <-> (complexity < 5)
usability == 4
complexity == 5
a)
b)


aa {r b}
bb {r b}
cc {r b}
aa != bb
aa != cc
bb != cc
2
2
2
8

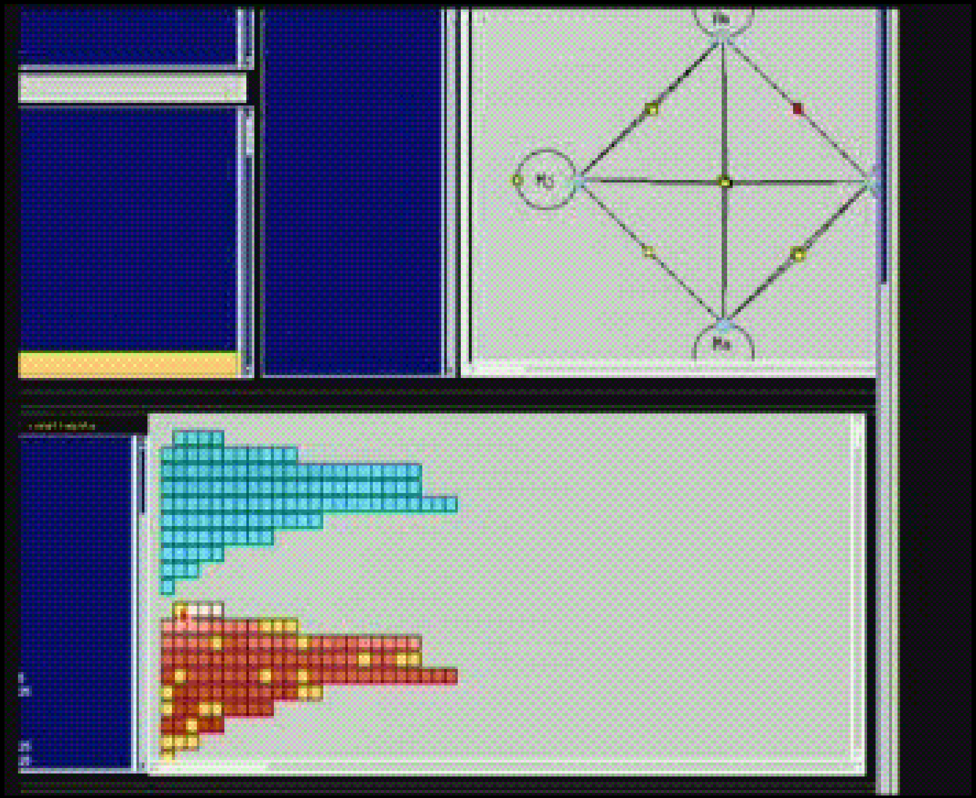


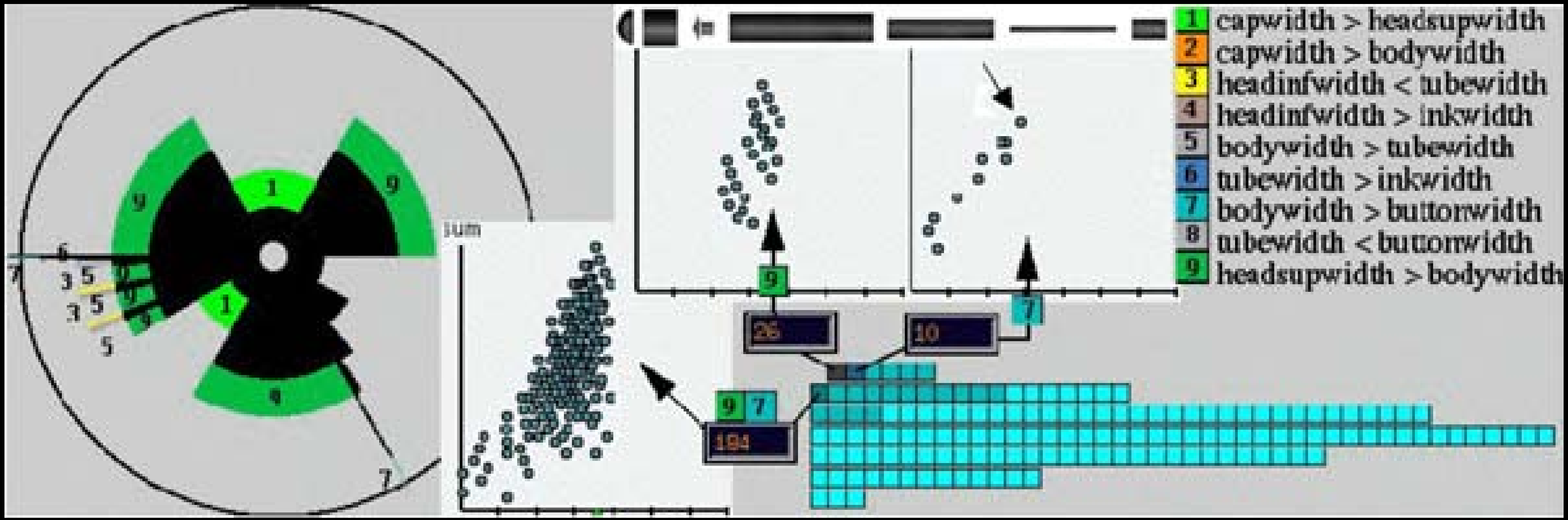

```
beauty {1 2 3 4 5}, implementation {1 2 3 4 5},
complexity {1 2 3 4 5},usability {1 2 3 4 5}

(beauty > 3) -> (implementation > 3)
(complexity > 3) -> (implementation > 3)
(usability > 3) <-> (complexity < 5)
beauty == 4, complexity == 5,
usability == 4, implementation == 1
```

e)

Lattice visualization





Kaleidoscope, MAP, and Lattice in one interface

Related works

- algorithm visualization
- perceptual inference
- attribute and influence explorers
- Tilebars
- dynamic aggregation (radial visualization)

Usability study

- what to test? - more than usability of system
- Help designers discover new solutions via algorithm visualization

Conclusion

- Presented Kaleidoscope, MAP, Lattice
- Interact with designers to explore, evaluate, and discover new design solutions
- visualization as a means for "interactive intelligence"