Network visualization techniques and evaluation

Mohammad GHONIEM

The Charlotte Visualization Center
University of North Carolina, Charlotte

March 15th 2007

Outline

Outline

Outline

Outline

Information Visualization

Definition

- A compact graphical representation
- A graphical user interface
- For the visualization and interaction with large numbers of items
  that may be a subset of an even larger dataset.

Information Visualization

Scope and usefulness

- Bears on data that are:
  - abstract, multi-dimensional, structured or unstructured.
- in order to:
  - Make discoveries, decisions, find explanations, or communicate about
  - visual patterns
    (trends, clusters, outliers, . . .)
  - groups of items,
  - individual items.

Information Visualization

Historical background

- Considerable leap in hardware technology and capabilities
  - Production and storage of large volumes of data;
  - Increased processing capabilities;
  - Improved display capabilities.
- Birth of data mining
  - Mathematical and statistical data analysis;
  - Visual information exploration.

Information Visualization

Principles and role

- Make the best use of human vision
  - detect patterns,
  - sense correlations,
  - confirm an intuition or formulate hypotheses.
- Make the best use of user's expertise
  - interactive exploration,
  - interactive construction of views.
- Precedes but does not replace classical *data mining*.
- Provides robust solutions for the most frequent data structures.

Organisation

Tree Visualization Techniques

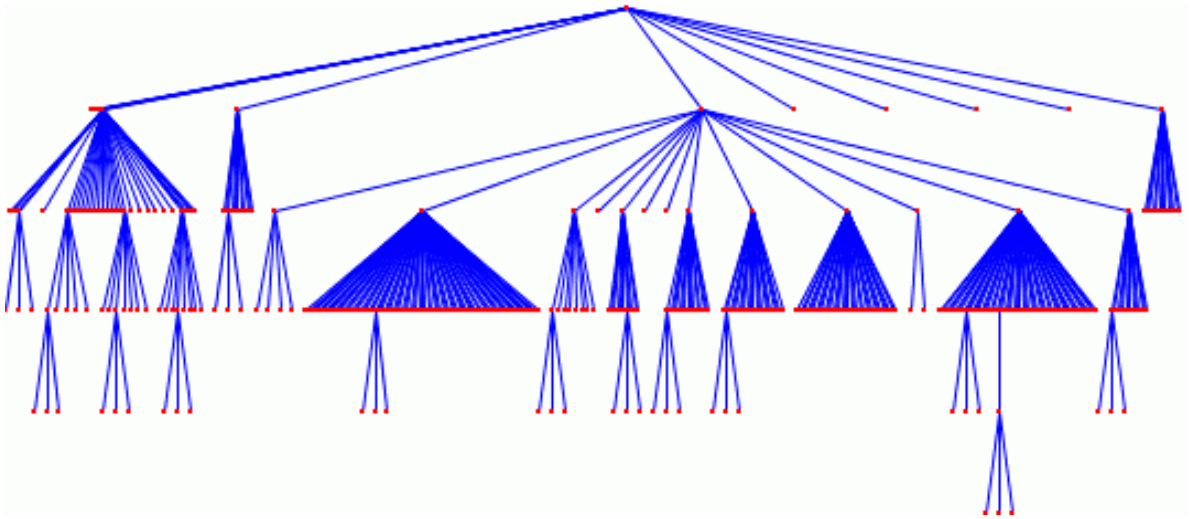
2D representations

- Node-link diagrams (cartesian / circular)
- Space-filling techniques (treemaps, icicle trees, circular)
- Non-euclidian space (hyperbolic trees)

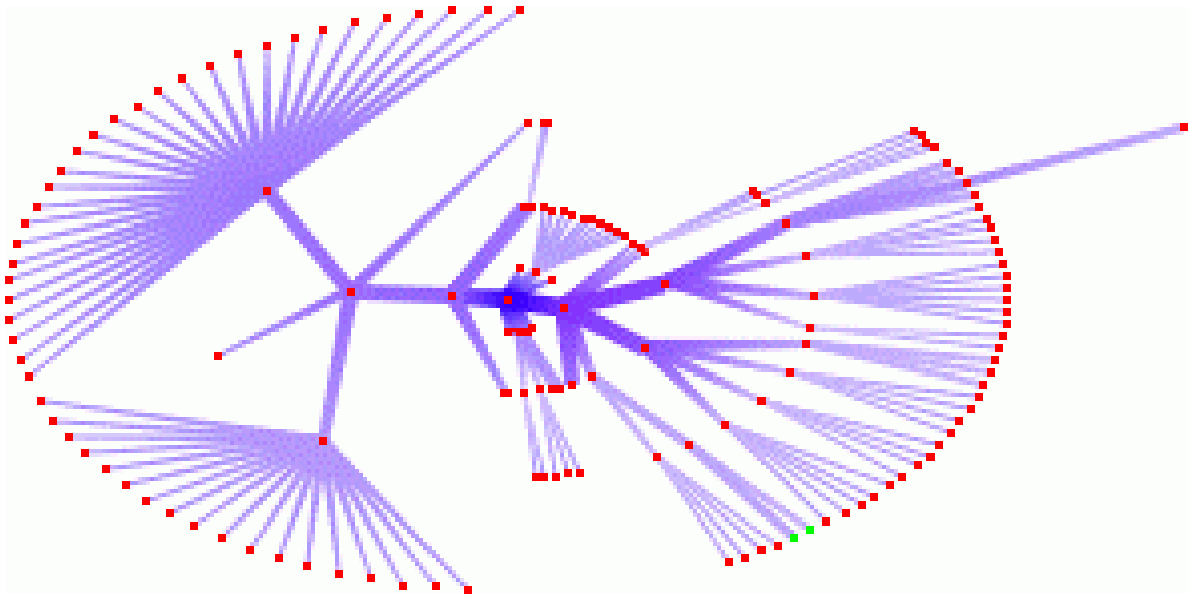
3D representations

- Node-link diagrams (cone trees)
- Non-euclidian space (hyperbolic trees)

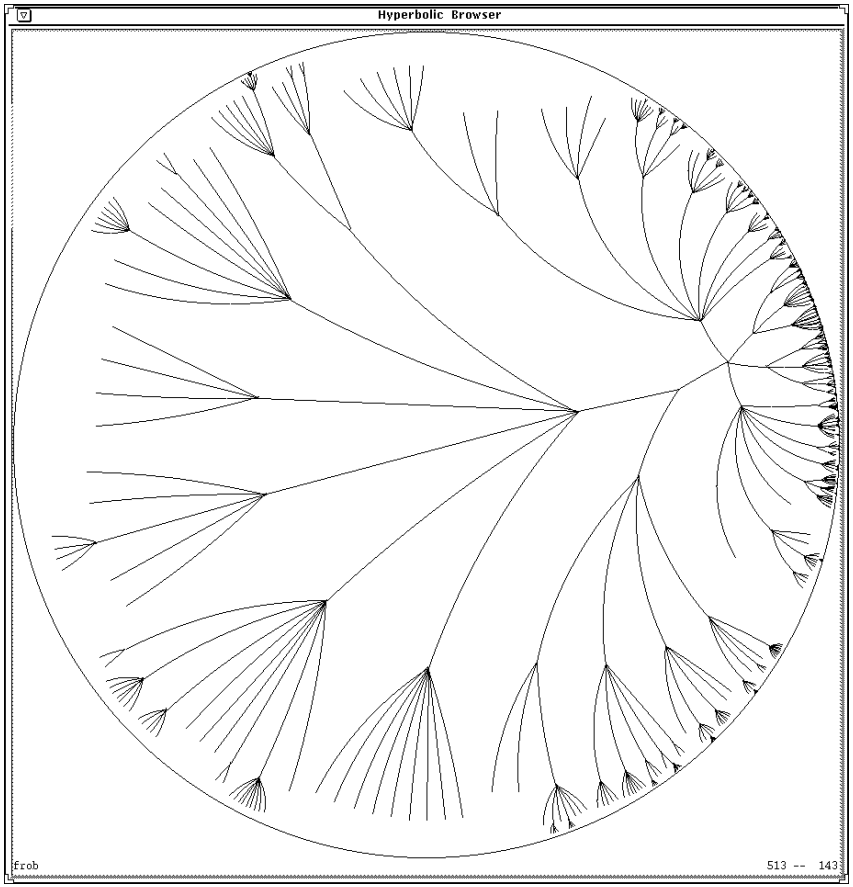
Overview of Tree Visualizations



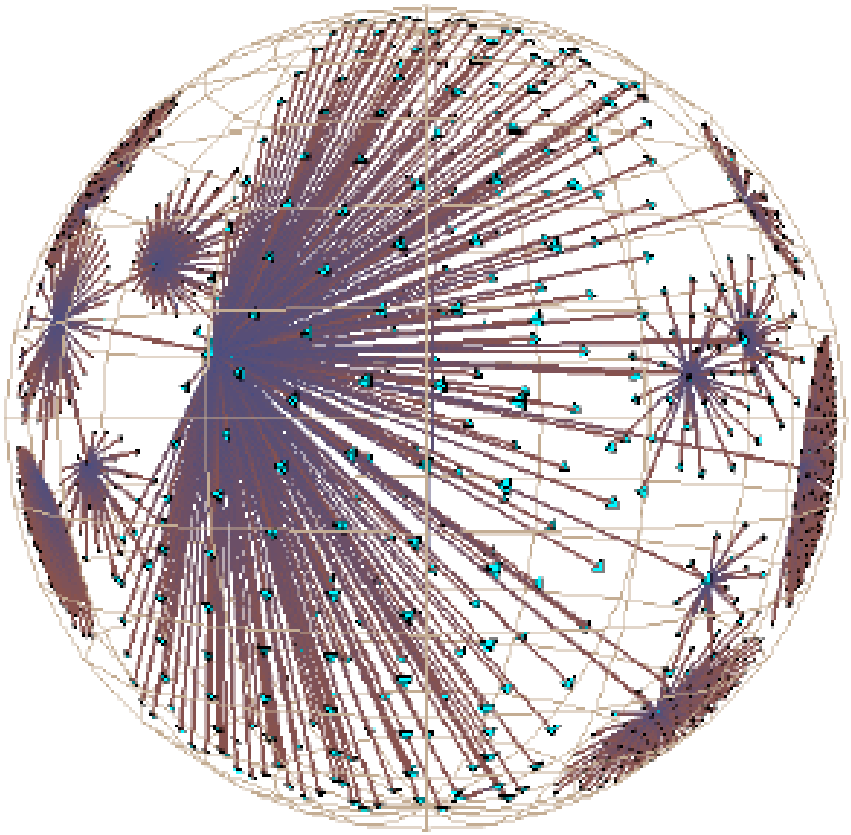
Overview of Tree Visualizations



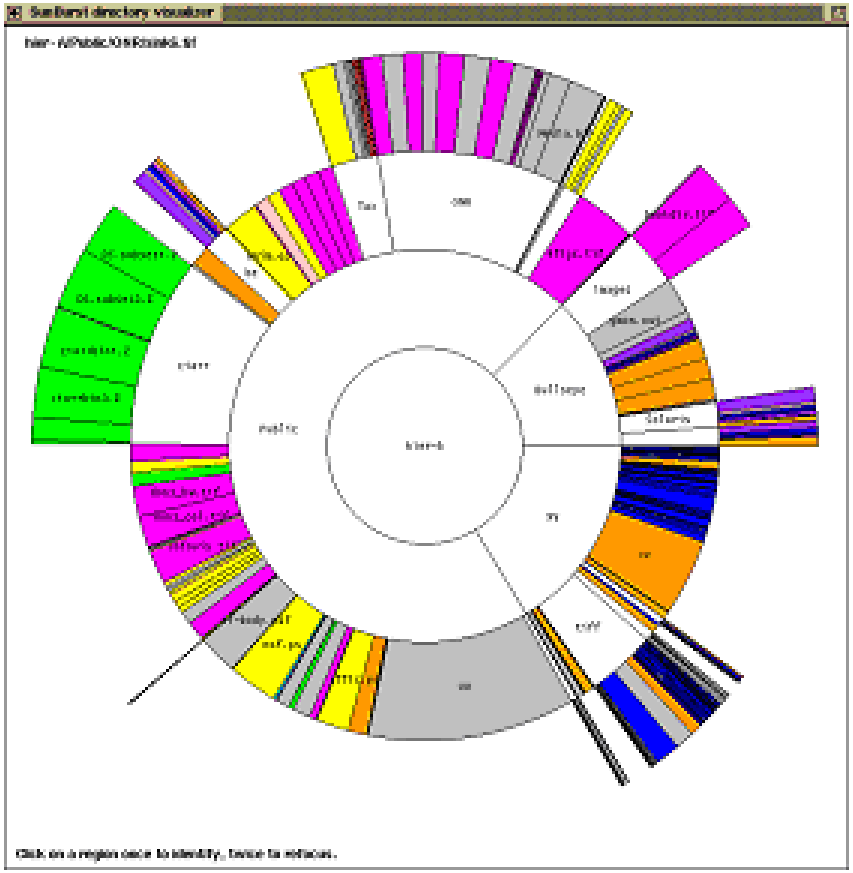
Overview of Tree Visualizations



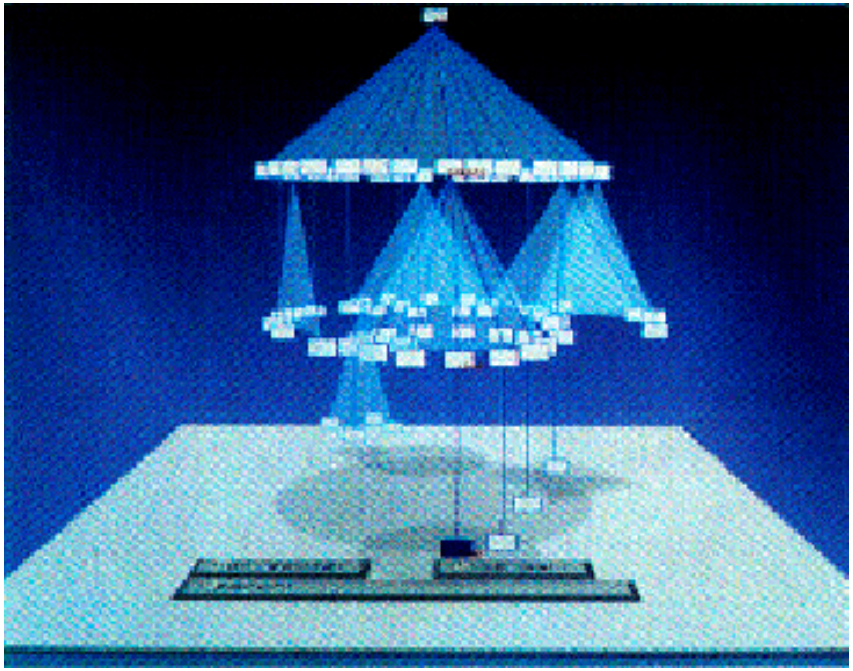
Overview of Tree Visualizations



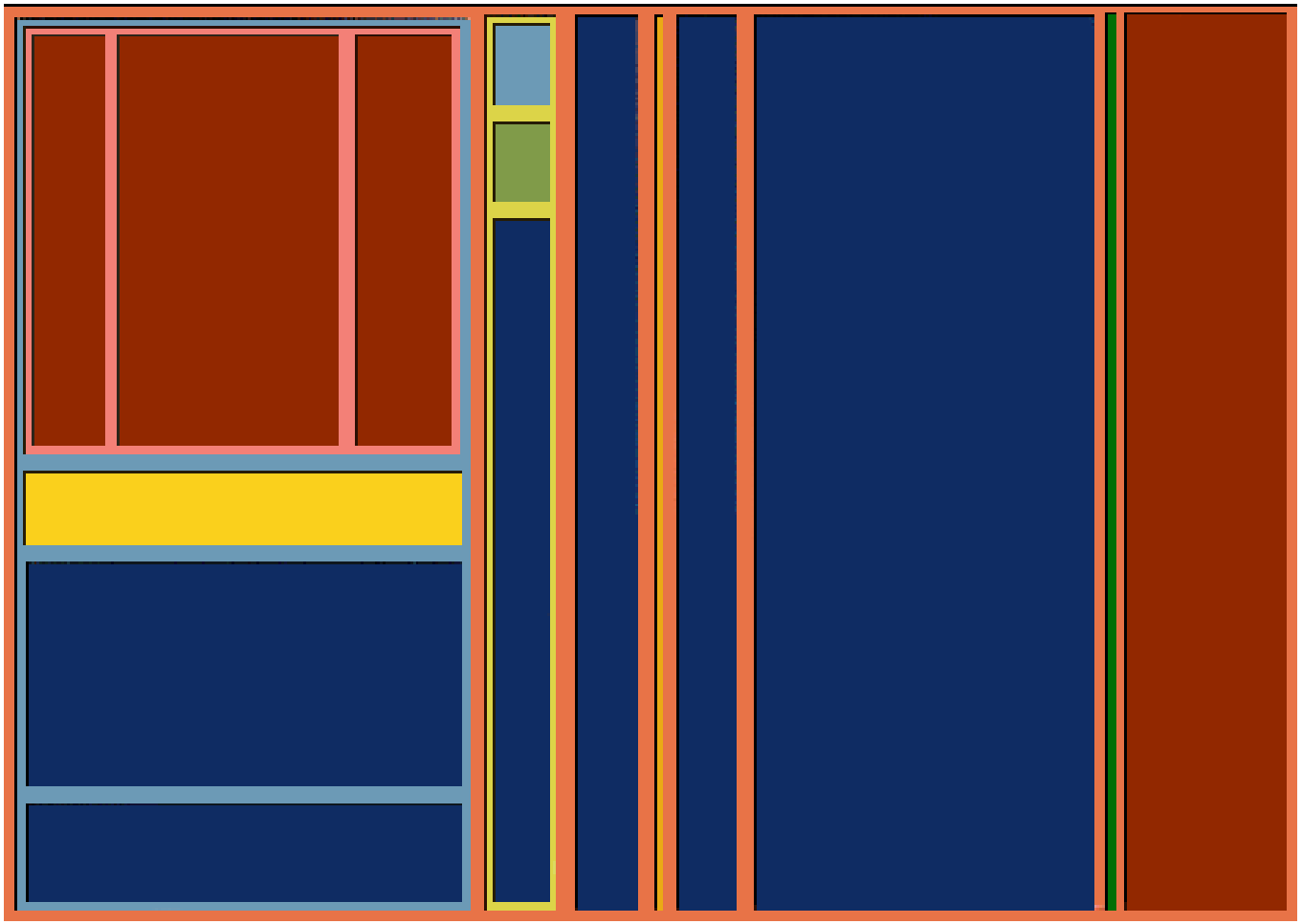
Overview of Tree Visualizations



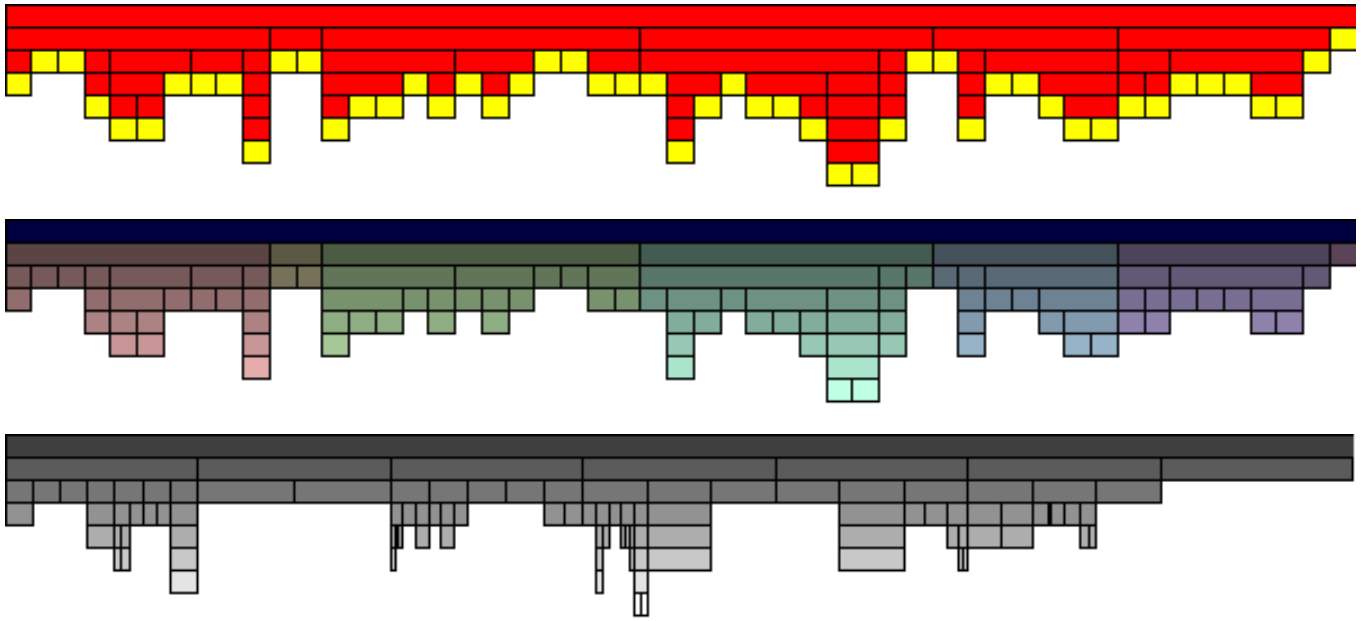
Overview of Tree Visualizations



Overview of Tree Visualizations



Overview of Tree Visualizations



Organisation

Graph theory basics

Definitions

- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$,
- $\mathcal{E} = \{(u, v)\}$, where $(u, v) \in \mathcal{V} \times \mathcal{V}$,
- two vertices are *adjacent* if they are connected by an edge,
- an edge is *directed* if an order is defined on its extremities,
- a graph is *directed* if its edges are directed,
- a *directed path* is a sequence of vertices $(v_1, \cdots, v_k)$ where $(v_i, v_{i+1}) \in E \; \forall i < k$,
- a directed path is a *cycle* if $(v_k, v_1) \in \mathcal{E}$,
- a directed graph is *acyclic* if it is cycle free,
- a graph is *planar* if it has an intersection free 2D drawing.

Graphs are ubiquitous

Examples

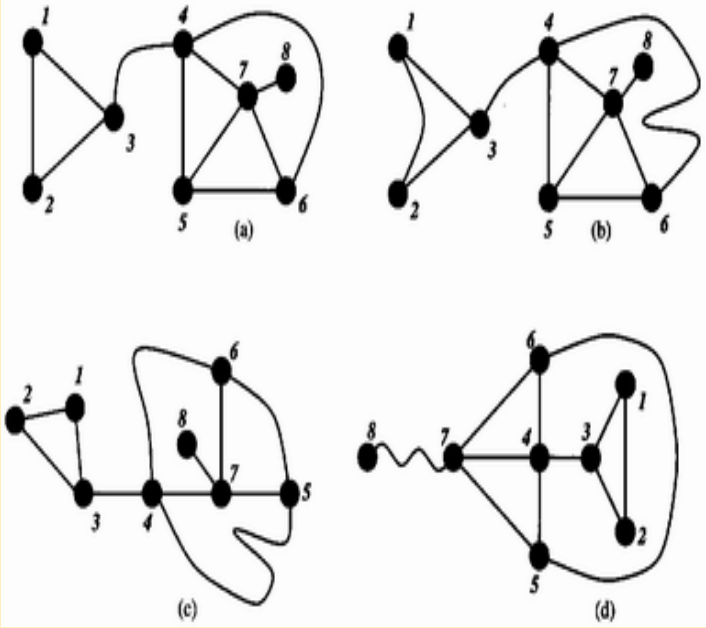
- Infrastructure networks (telecommunications, power, roads).
- Social networks (acquaintances, crime networks).
- Co-citation network, etc.

Graphs and representations

Definition

- a graph is an abstract entity $\neq$ its representations.

Layout



Graph visualization techniques

Visualization techniques

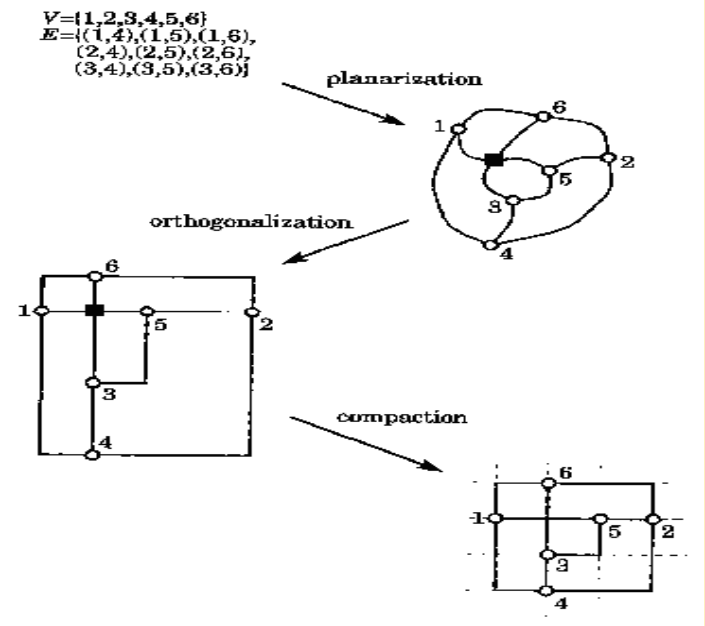
- Node-link diagrams,
- Adjacency matrices.

Aesthetic rules and drawing conventions

- Drawing conventions: polyline drawing, grid drawing, upward/downward etc.
- Aesthetic rules: min. intersections, min. edge length, min. area etc.
- Some rules are conflicting.
- Some user studies and experiments (H. Purchase, C. Ware).

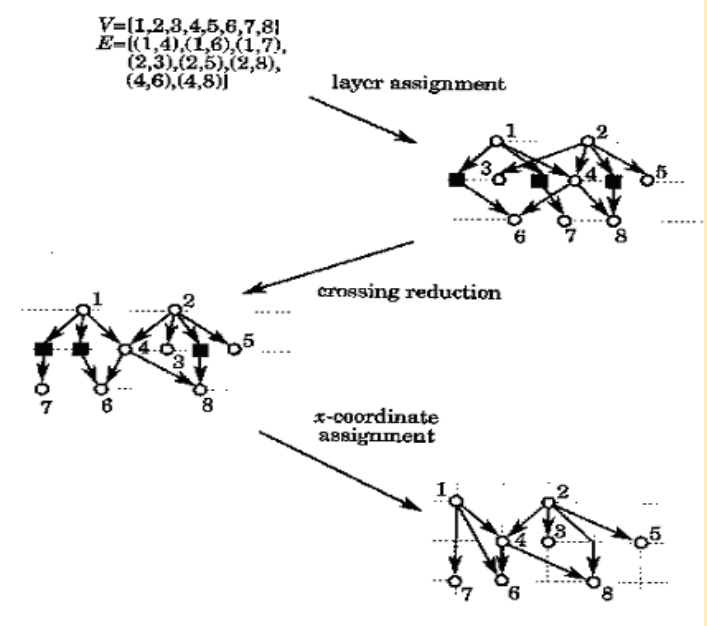
Graph visualization techniques

Graph drawing approaches



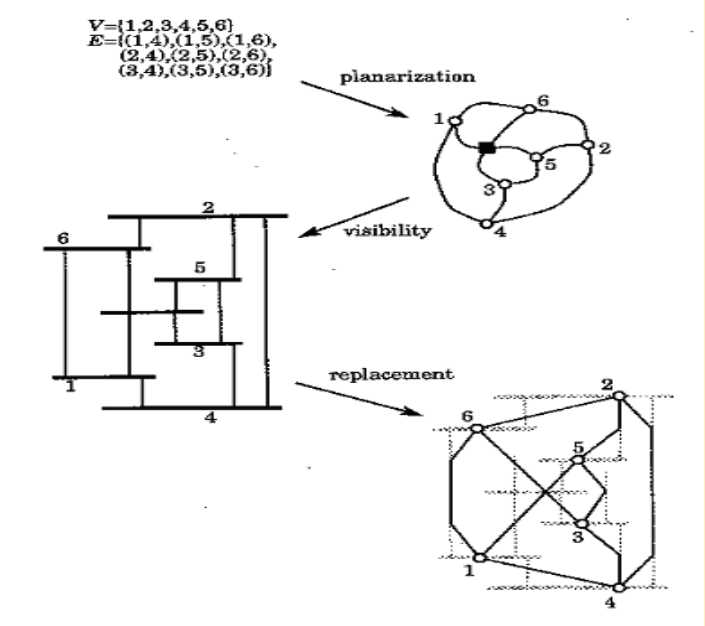
Graph visualization techniques

Graph drawing approaches



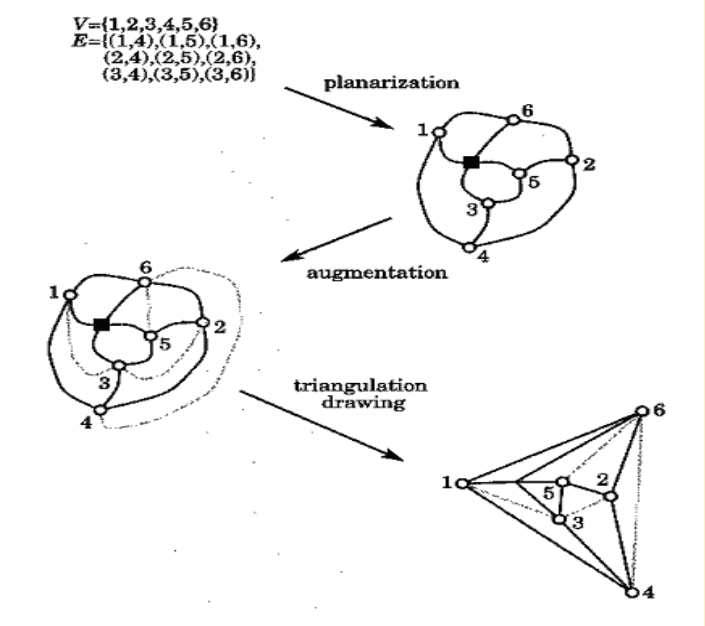
Graph visualization techniques

Graph drawing approaches

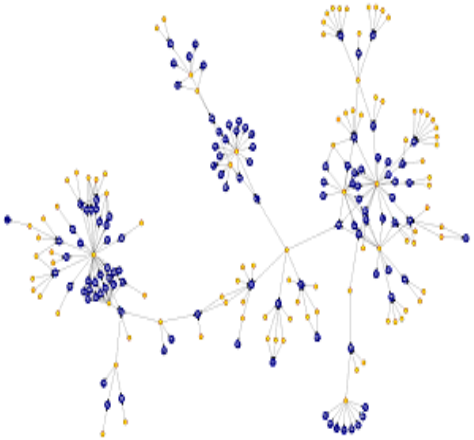


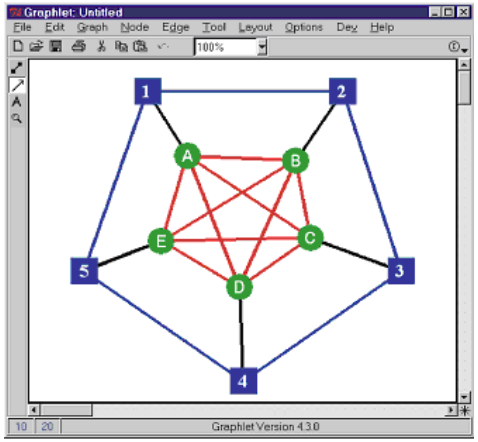
Graph visualization techniques

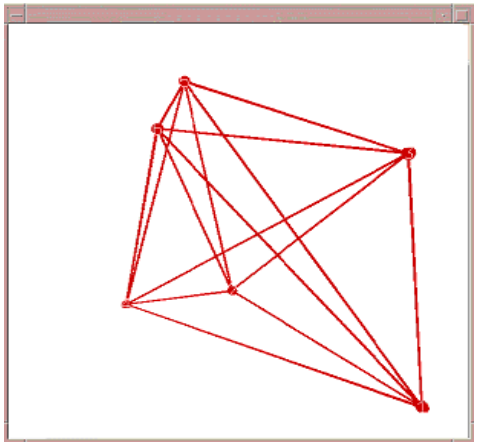
Graph drawing approaches

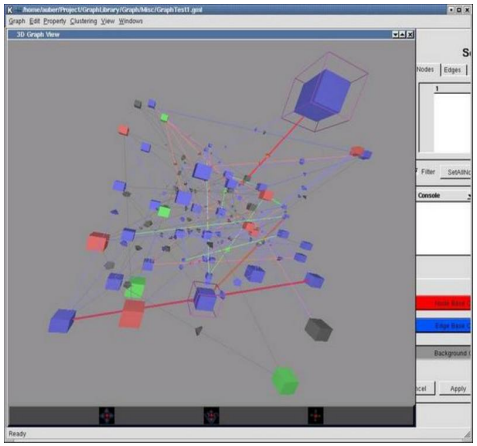


Graph visualization overview









Graph visualization frameworks

Existing frameworks

- Tulip (University of Bordeaux – France),
- Pajek (University of Ljubljani – Slovenia),
- GraphViz (AT&T),
- JUNG (University of California, Irvine).

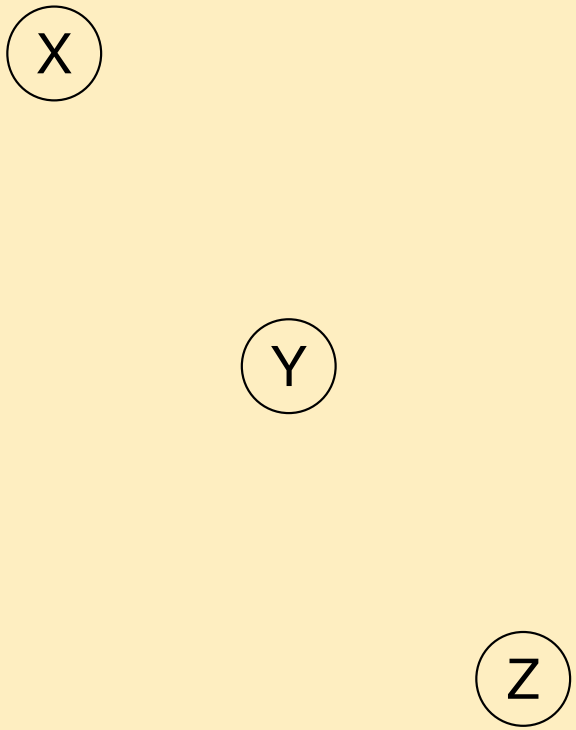
Graph visualization techniques

Node-link diagrams

- Widespread and well studied wrt sparse graphs.
- Cluttered views when link density increases.
- Instable layout algorithms
- Unusable for dynamic graphs.
- Begs for an alternate representation.

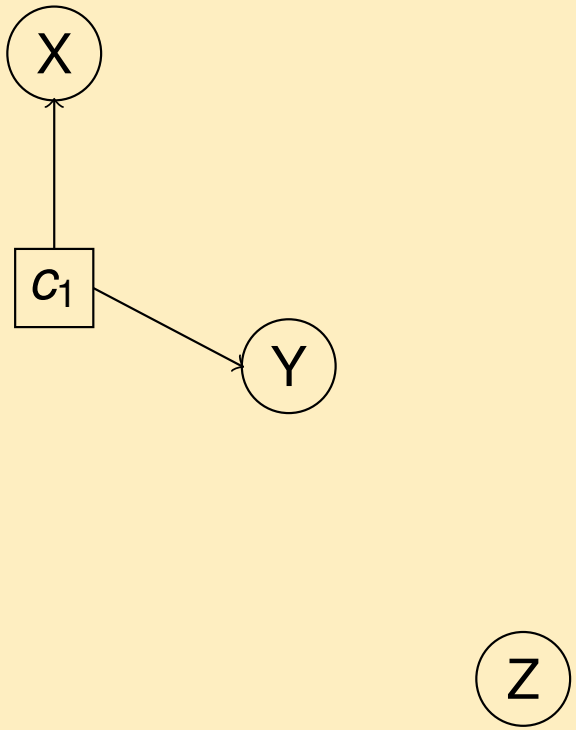
Monitoring co-activity graphs

Variables + Constraints



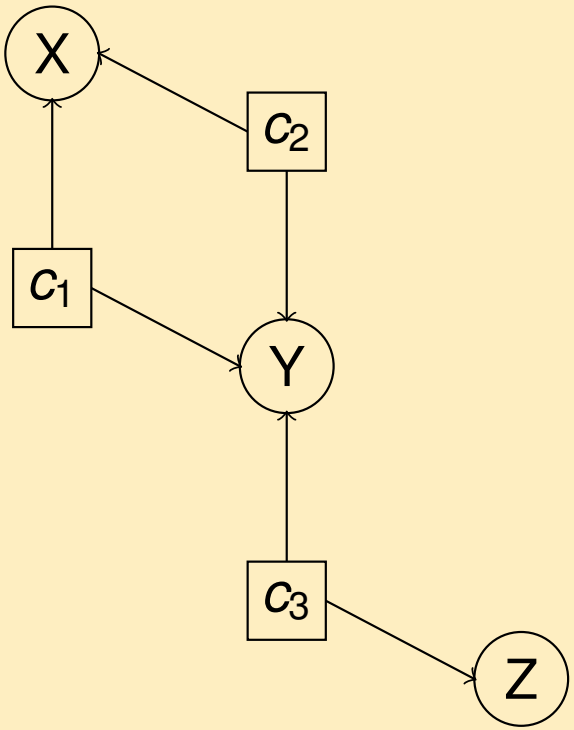
Monitoring co-activity graphs

Variables + Constraints



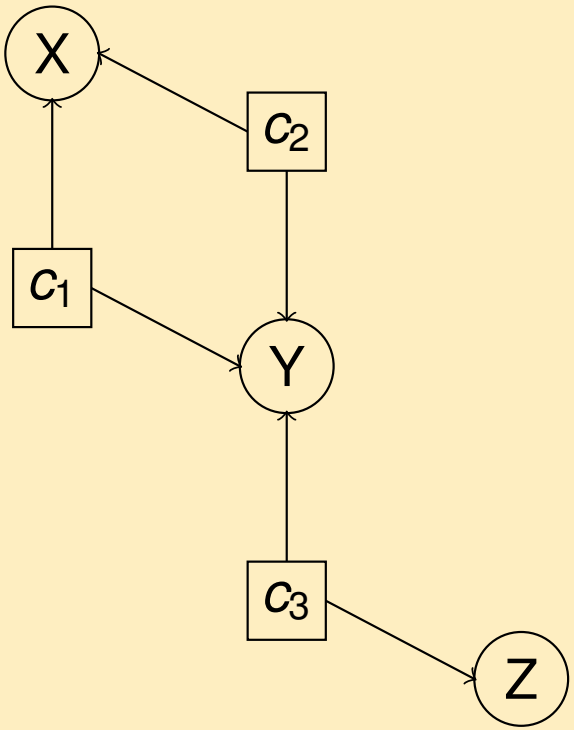
Monitoring co-activity graphs

Variables + Constraints

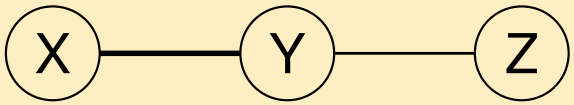


Monitoring co-activity graphs

Variables + Constraints

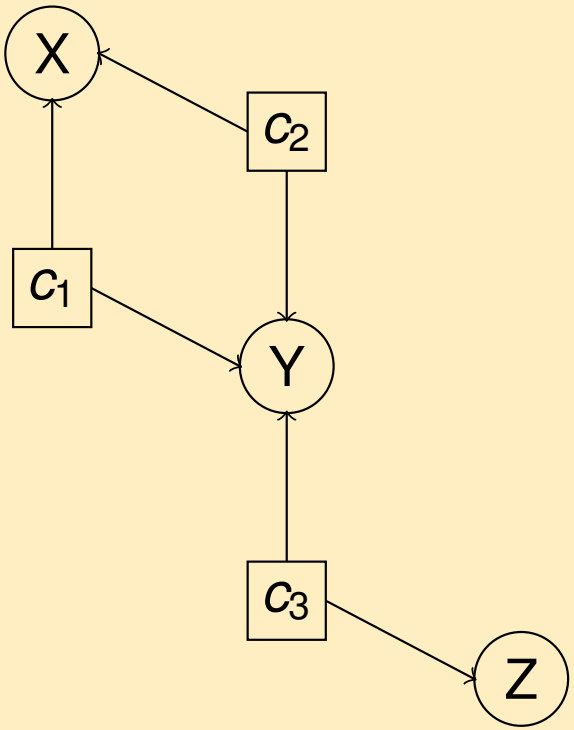


Variables

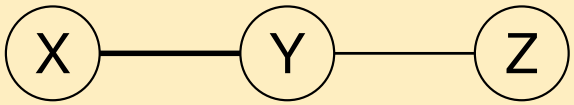


Monitoring co-activity graphs

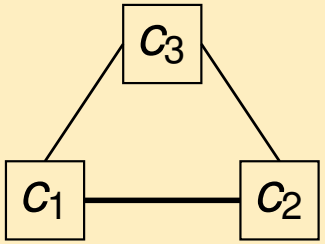
Variables + Constraints



Variables

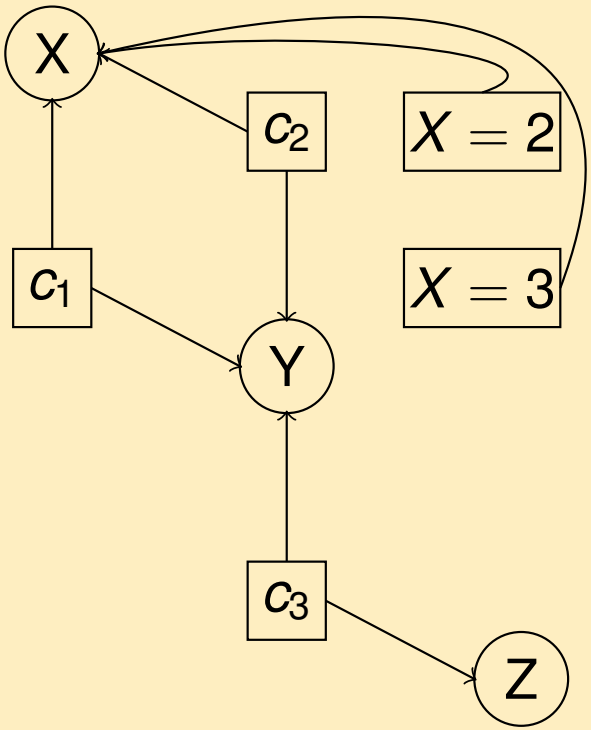


Contraintes

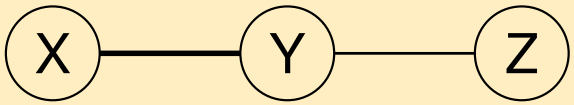


Monitoring co-activity graphs

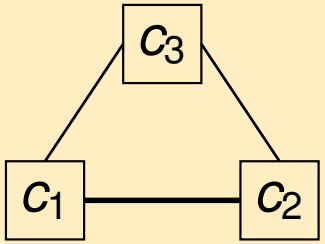
Variables + Constraints



Variables

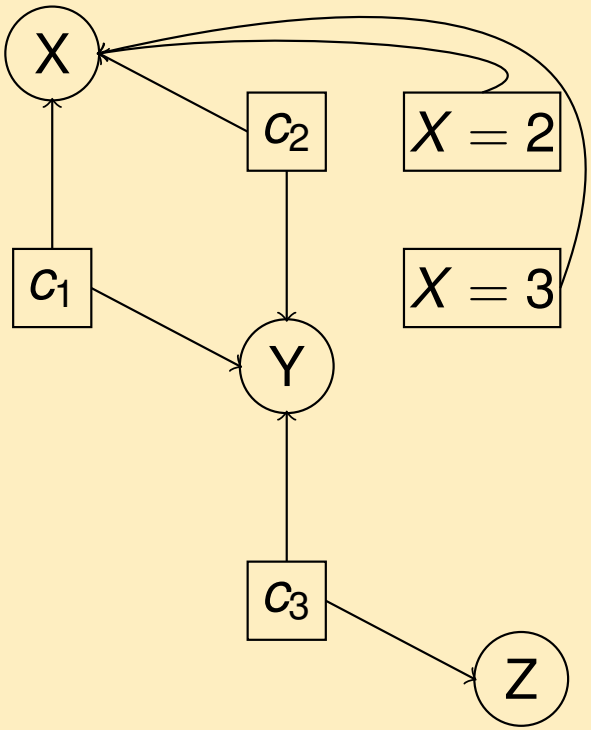


Contraintes

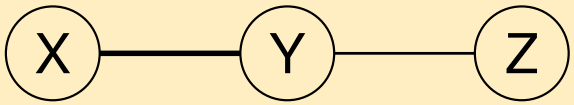


Monitoring co-activity graphs

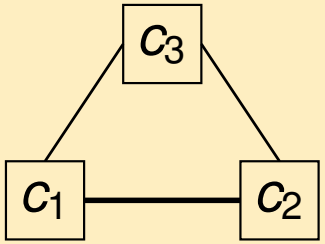
Variables + Constraints



Variables

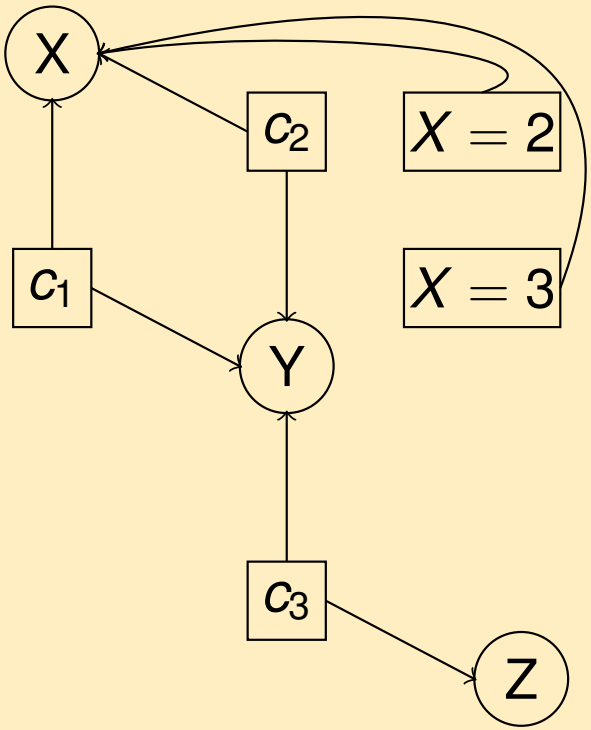


Contraintes

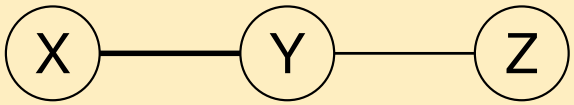


Monitoring co-activity graphs

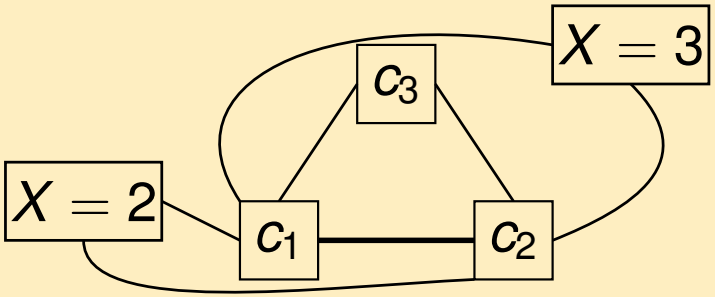
Variables + Constraints



Variables

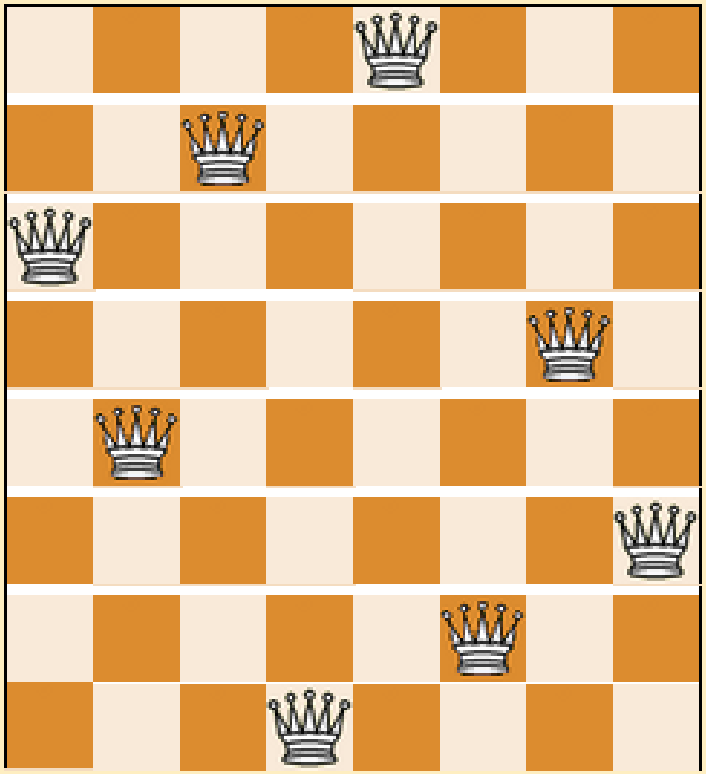


Contraintes



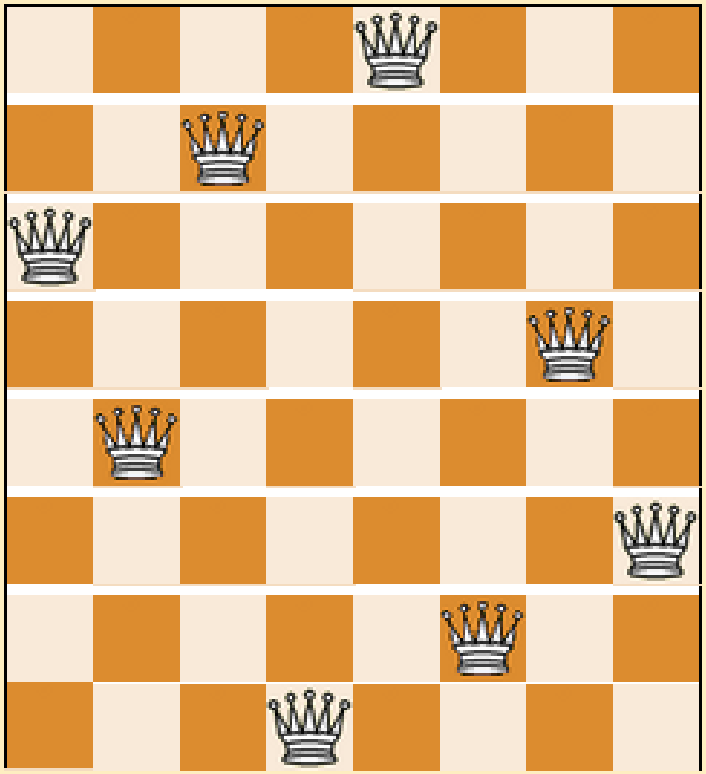
Co-activity graphs

The 8 queen problem

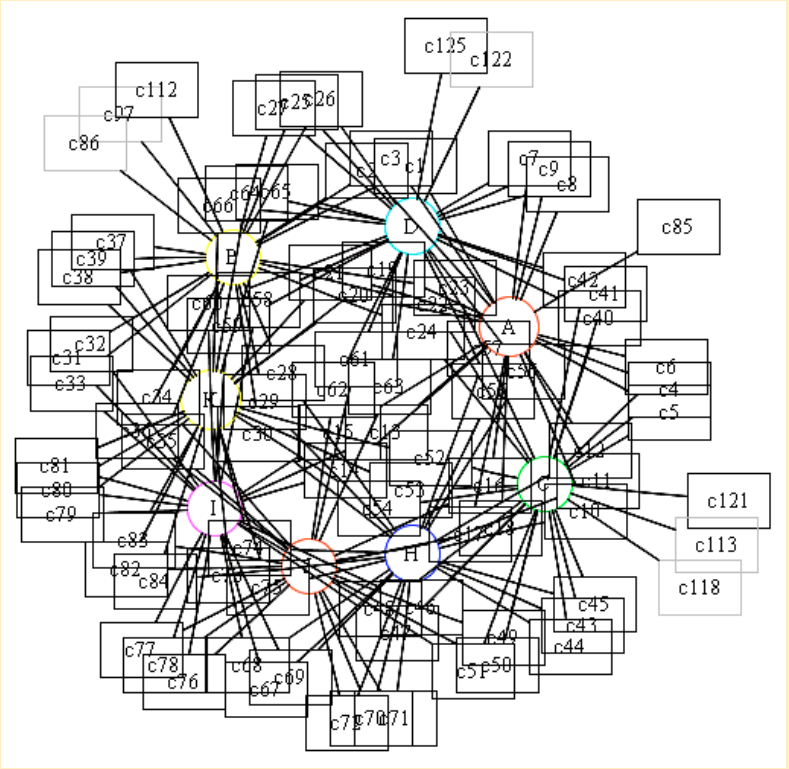


Co-activity graphs

The 8 queen problem

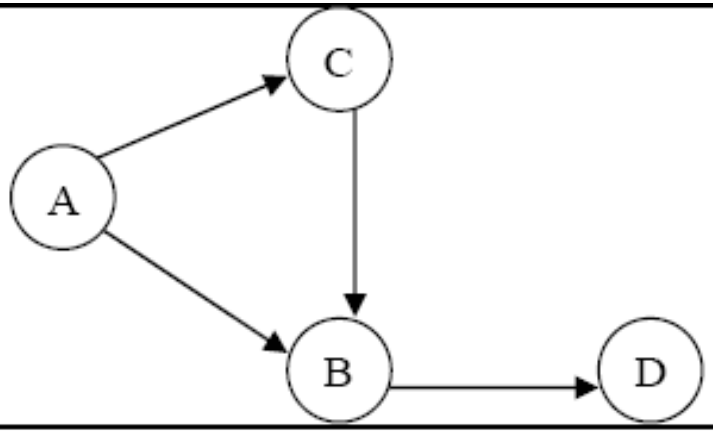


Co-activity graph



Organisation

Matrix-based representation of graphs



Un graphe orienté comportant quatre sommets

	A	B	C	D
A	0	0	0	0
B	1	0	1	0
C	1	0	0	0
D	0	1	0	0

Matrice d'adjacence équivalente

	A	B	C	D
A				
B	■		■	
C	■			
D		■		

Représentation matricielle

Matrix-based representation of graphs

Strengths

- Relies on a well-known mathematical representation.
- No clutter nor occlusion.
- Orderable, predictible for most common order relations.
- Displays existing and missing links.

Shortcomings

- Unfamiliar visualization.
- Not effective for path related tasks.

Matrix-based representation of graphs

Strengths

- Relies on a well-known mathematical representation.
- No clutter nor occlusion.
- Orderable, predictible for most common order relations.
- Displays existing and missing links.

Shortcomings

- Unfamiliar visualization.
- Not effective for path related tasks.

Matrix-based representation of graphs

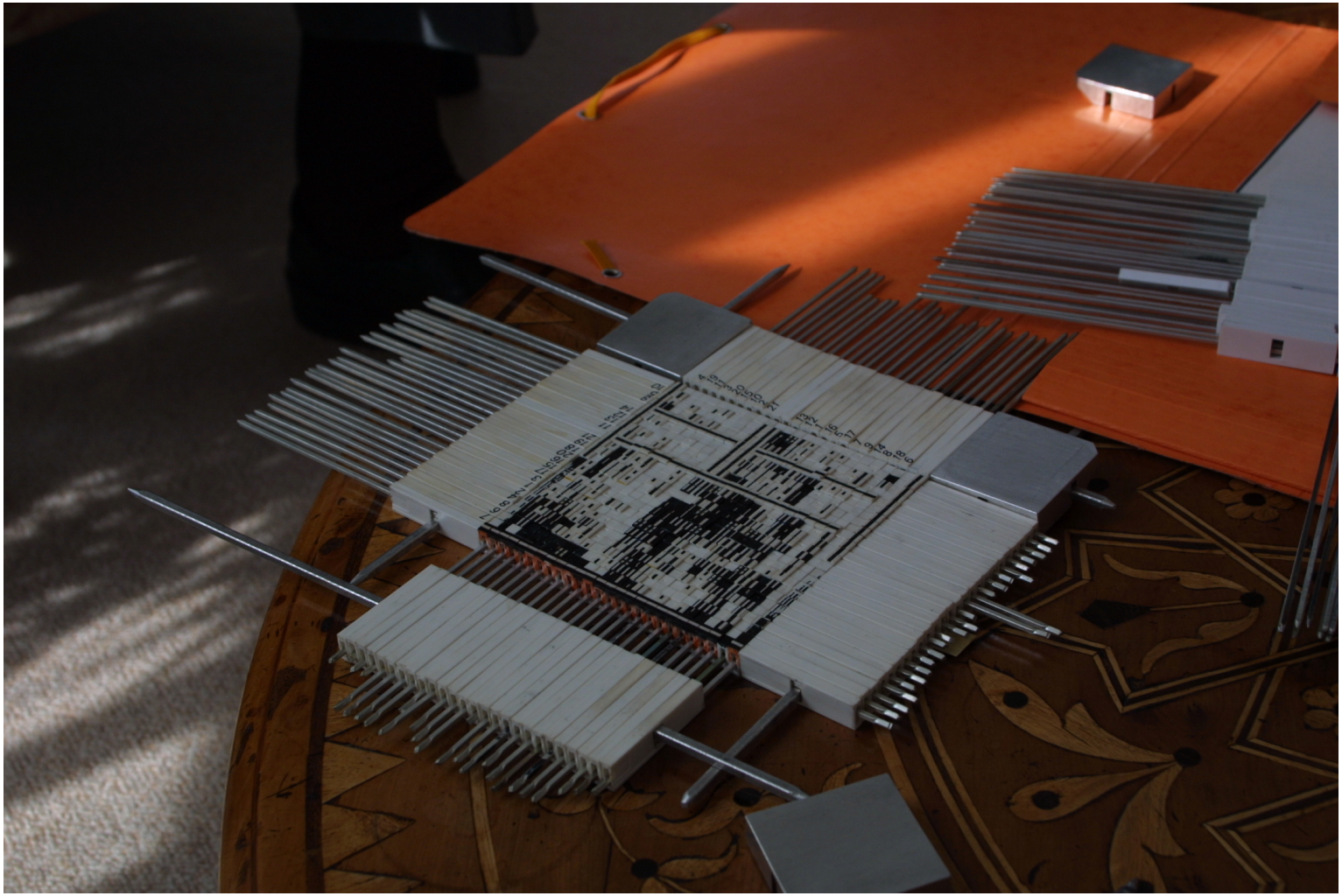
Strengths

- Relies on a well-known mathematical representation.
- No clutter nor occlusion.
- Orderable, predictible for most common order relations.
- Displays existing and missing links.

Shortcomings

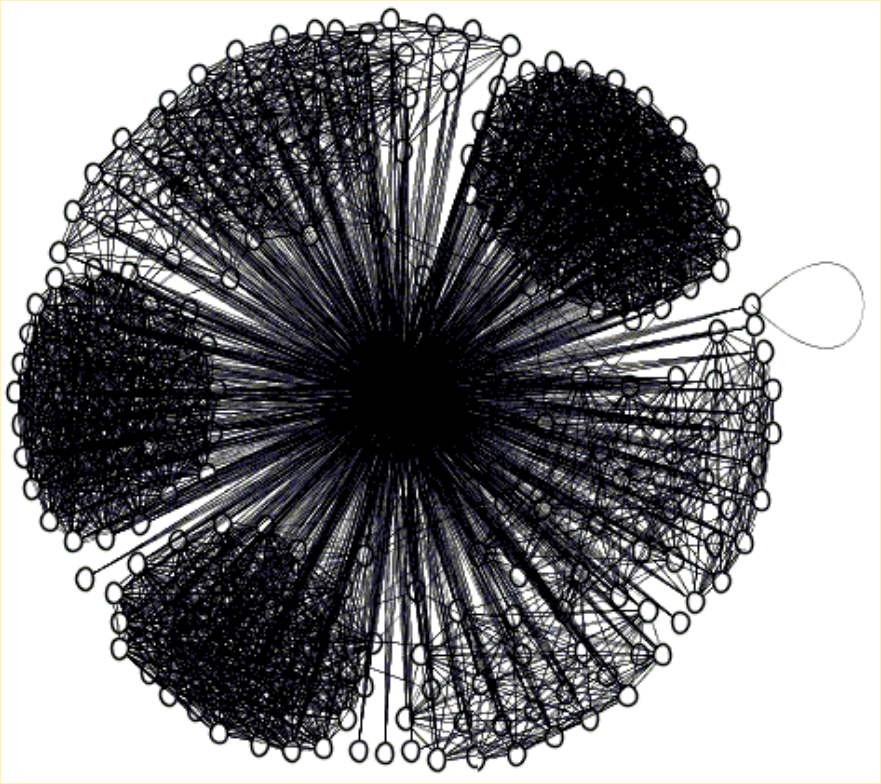
- Unfamiliar visualization.
- Not effective for path related tasks.

The matrix of Bertin



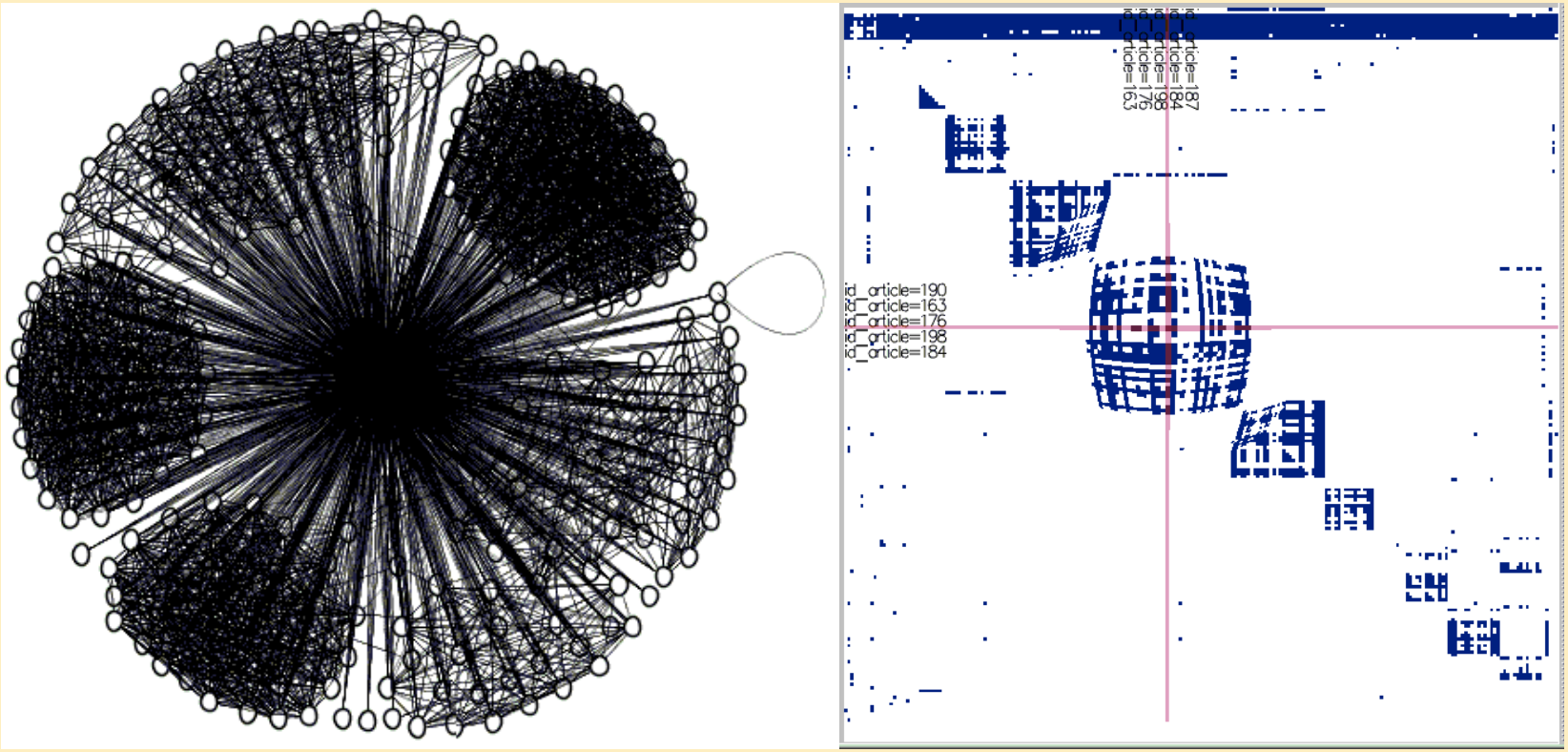
Cluster revelation through permutations

Orderability magic



Cluster revelation through permutations

Orderability magic



Matrix-based representation of graphs

Properties of matrices

Matrix-based representation of graphs

Multi-scale self-contained visualization

Matrix-based representation of graphs

Multi-scale self-contained visualization

Matrix-based representation of graphs

Multi-scale self-contained visualization

Matrix-based representation of graphs

But...

How readable is the matrix-based representation ?

Evaluation of the readability of matrices

Principe

- Compare two alternative representations of graphs wrt
- A predefined list of common tasks,
- A set of graphs of variable sizes and link density.
- The readability is measured according to two indicators:
  1. answer time;
  2. error rate.

Evaluation of the readability of matrices

Taxonomy of tasks

- Tasks related to the overview.
- Tasks related to vertices.
- Tasks related to links.
- Tasks related to paths.
- Tasks related to subgraphs.

Evaluation of the readability of matrices

Seven tasks

1. Estimate the number of vertices.
2. Estimate the number of links.
3. Find the most connected node.
4. Find a node by its name.
5. Say whether two links are connected.
6. Say whether two nodes have a common neighbor.
7. Find a path between two nodes.

A few observations about the tasks

- A minimal selection of tasks.
- Primitive tasks.

Evaluation of the readability of matrices

Experimental precautions

- Preliminary demonstration and training stage.
- Guidelines : answer fast and right, may skip questions if too difficult.
- Two two-fold sessions.
- Three breaks (5 min, 10 min, 5 min).
- Equalize weariness effects.
- Equalize learning effects.
- Bounded answer time (45 sec. max.)

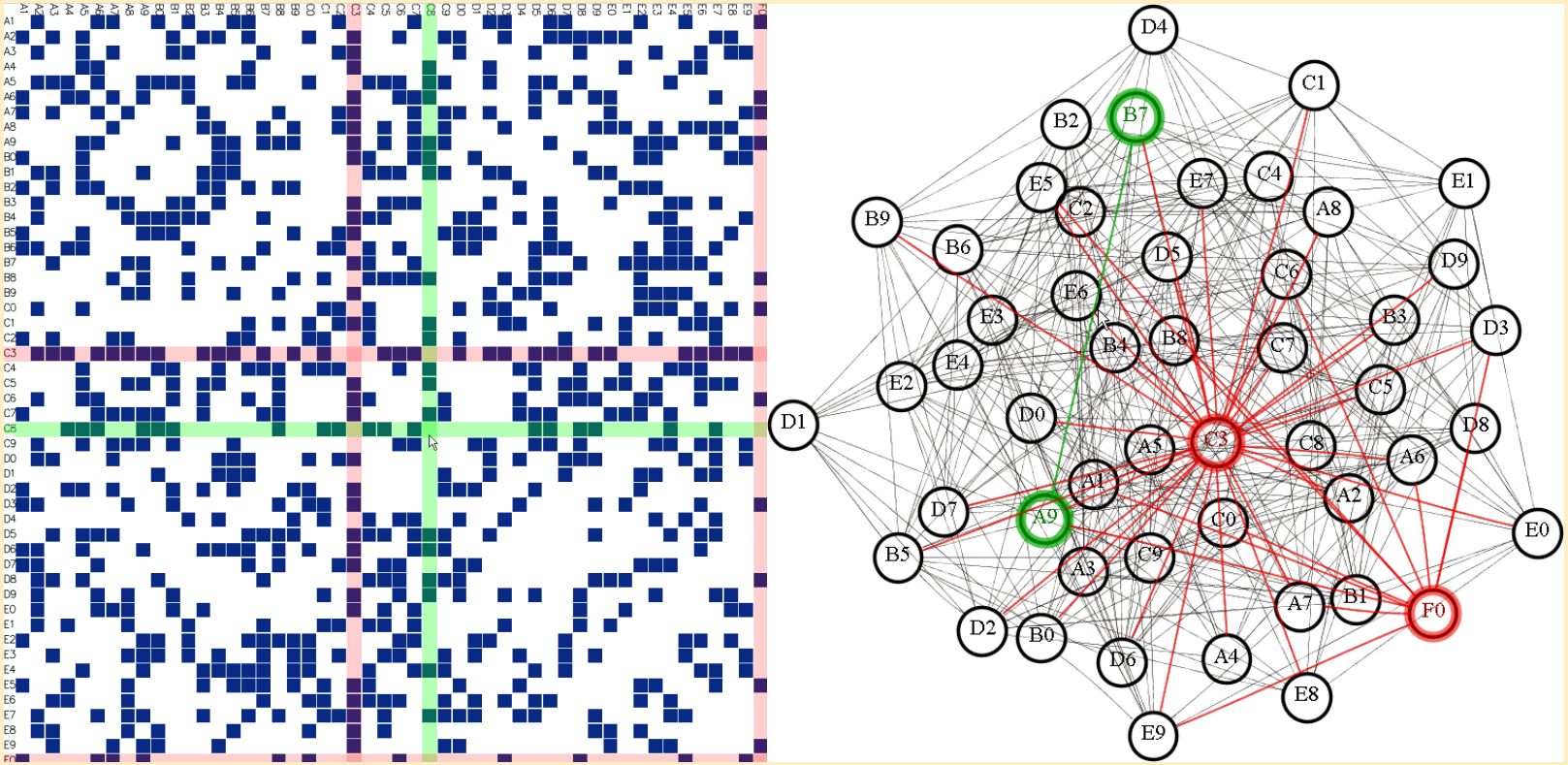
Evaluation of the readability of matrices

Setup

- Nine random graphs with increasing size and link density.
- Use neato/graphviz package for node-link diagrams.
- Matrix-based representation in OpenGL/Java.
- Presets tothe best advantage of each representation.
- Minimal interaction (node and link highlights).

Evaluation of the readability of matrices

Highlight and selection of nodes and links



Evaluation of the readability of matrices

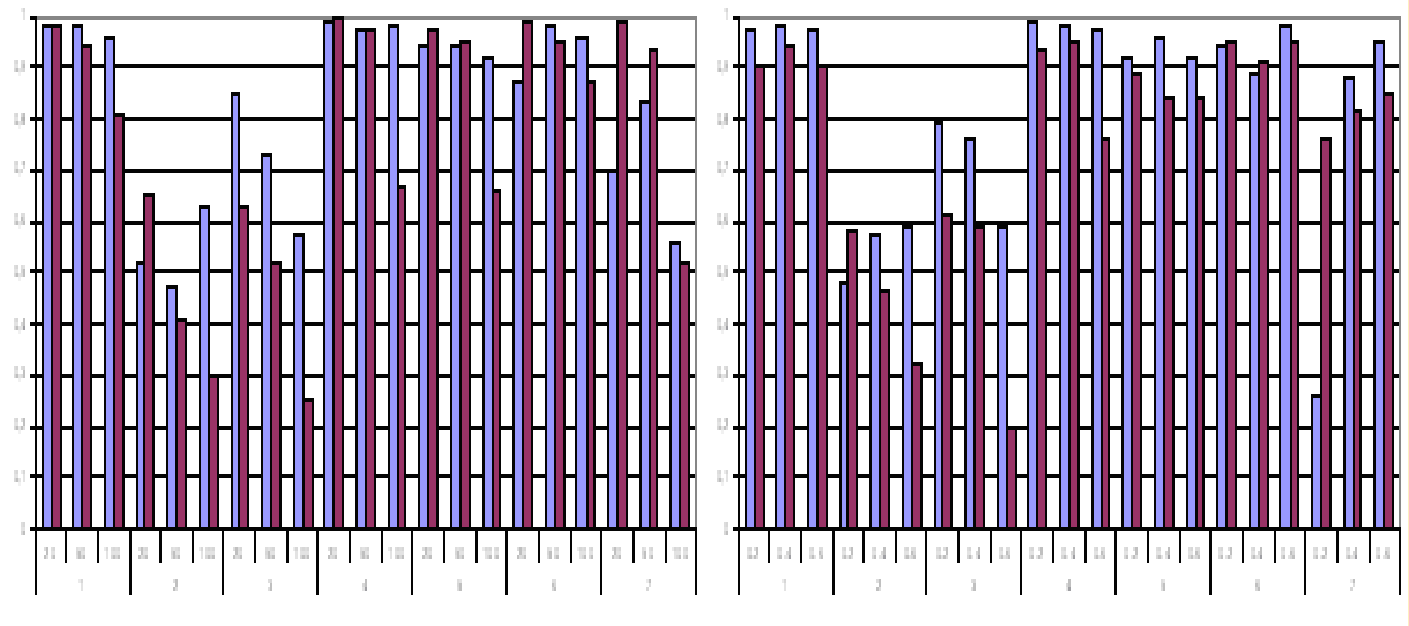
User population and result analysis

- 36 Ph.D. students or assistant professors in computer-science.
- Analysis: quantitative

Evaluation of the readability of matrices

User population and result analysis

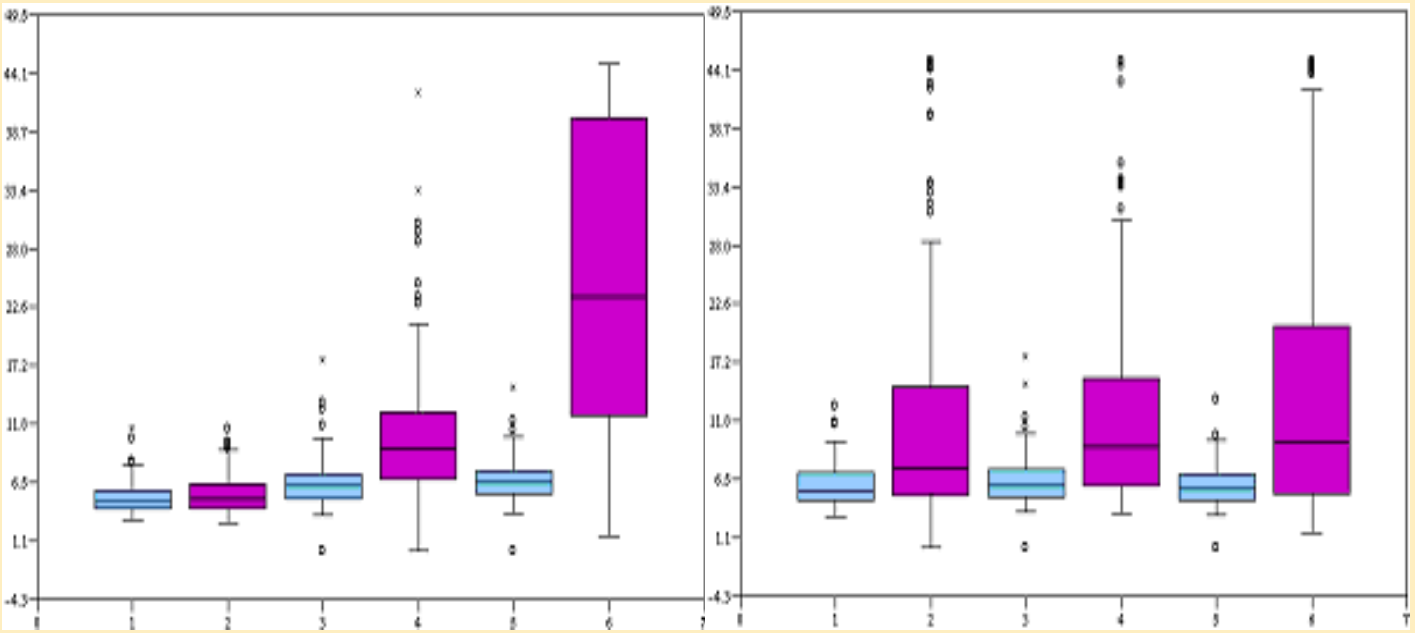
- 36 Ph.D. students or assistant professors in computer-science.
- Analysis: quantitative



Evaluation of the readability of matrices

User population and result analysis

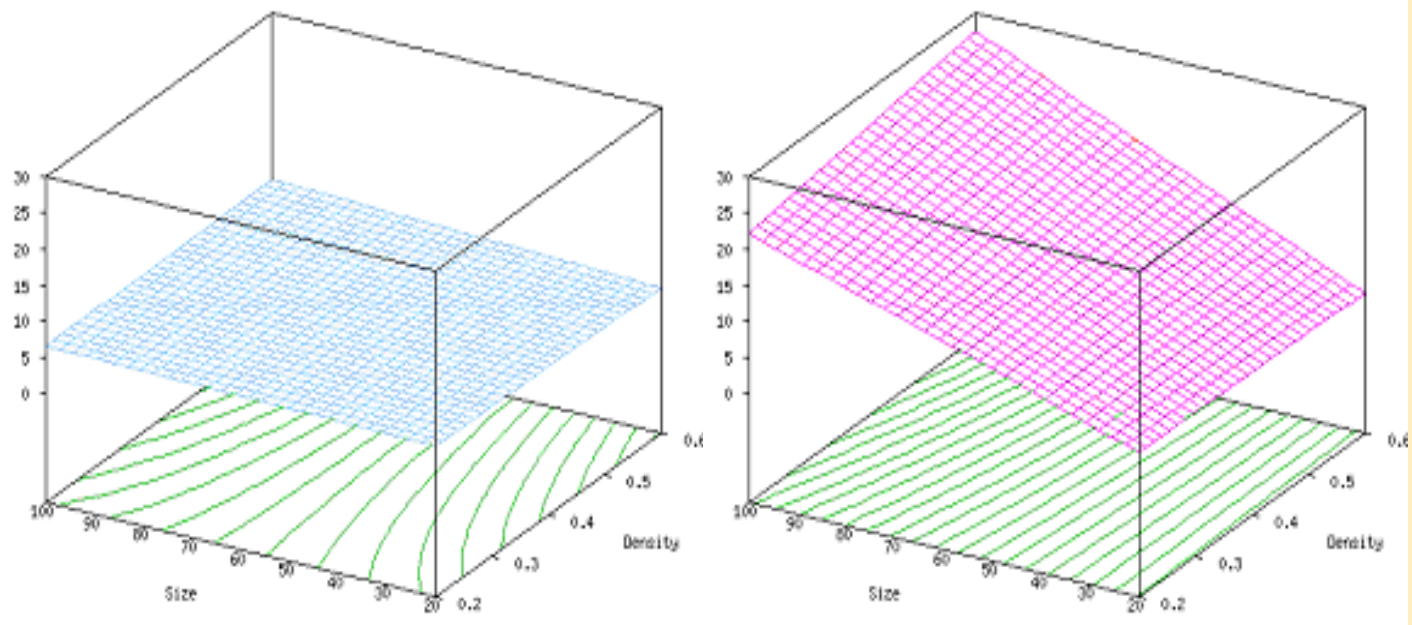
- 36 Ph.D. students or assistant professors in computer-science.
- Analysis: quantitative and qualitative



Evaluation of the readability of matrices

User population and result analysis

- 36 Ph.D. students or assistant professors in computer-science.
- Analysis: quantitative and qualitative + 3D model.



Evaluation of the readability of matrices

Results

- Better results with the matrix-based representation for 6/7 tasks wrt
  - dense graphs,
  - large graphs.
- Node-link diagrams are preferable for small sparse graphs.
- Path related tasks are difficult to carry out.

User reactions

- Not so enthusiastic in the beginning, except a few.
- Very positive at the end of the evaluation, except a few.

Organisation

Visualization of constraint-oriented programming graphs

Properties of such graphs

- Very densely connected.
- Large.
- Complex temporal relations between nodes.

Visualization solution

- Matrix-based representation:
  - no clutter,
  - space-filling technique,
  - adapted for focus + context techniques (fisheye),
  - multi-scale representation (clustering + agregation).
- History animation through dynamic queries,

Monitoring of dynamic graphs: example 1

Pedagogical problem (sort 100 variables)

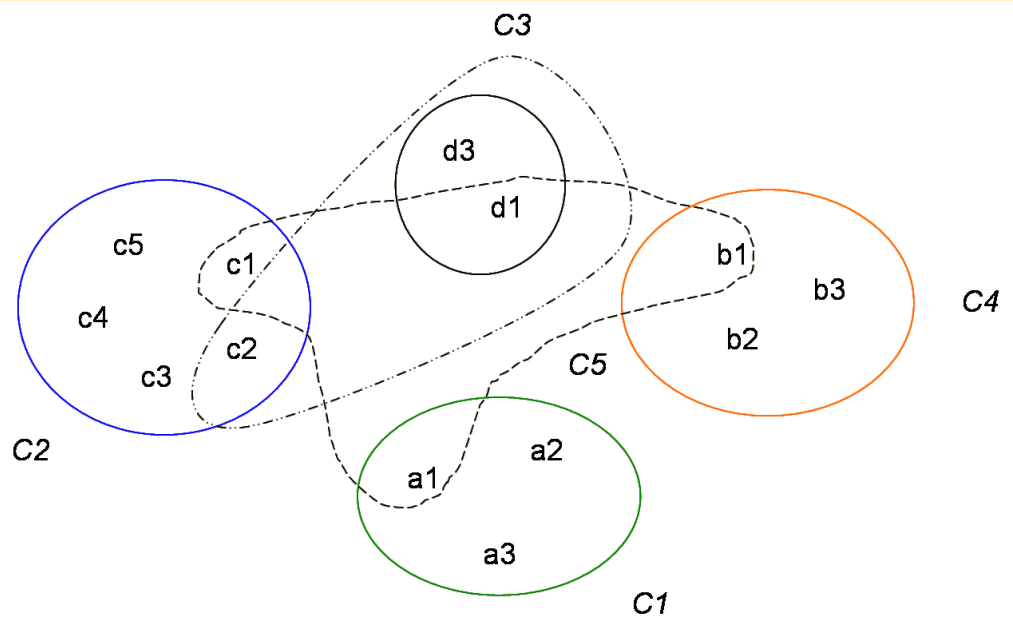
- 100 variables $x_{i \in [1,100]}$ in $[1, 100]$.
- 99 constraints : $\forall i \in [1, 99], x_i < x_{i+1}$.

Monitoring of dynamic graphs: example 1

Constraint graph of the sort problem

Monitoring of dynamic graphs: example 2

alldiff constraints problem



Monitoring of dynamic graphs: example 2

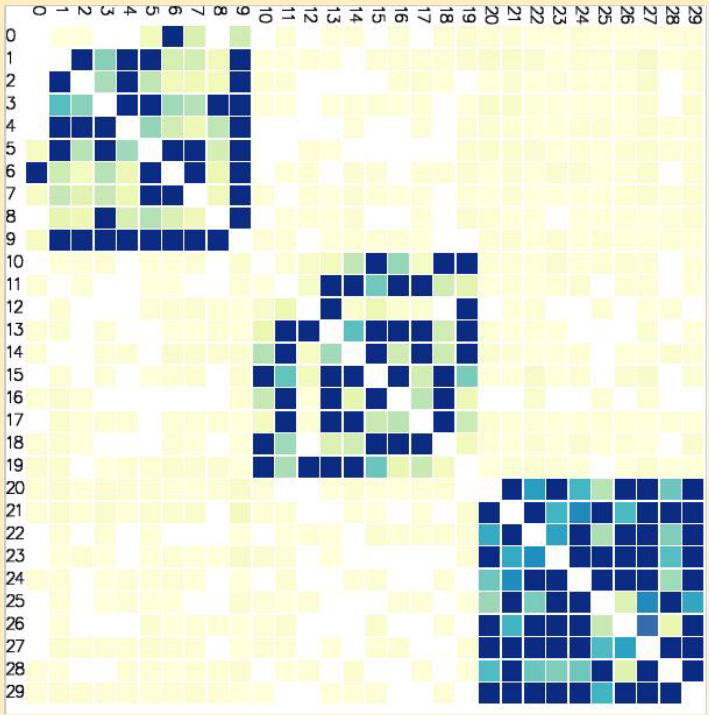
Automatic segmentation and animation of the solving process

Problem structure investigations

Aid for debugging and fine-tuning of programs

- 3 sets of variables with strong internal links.
- Problem structure is visible at the end of propagation phase.
- Link density has no negative impact on the clarity of the overview.

Variables × Variables

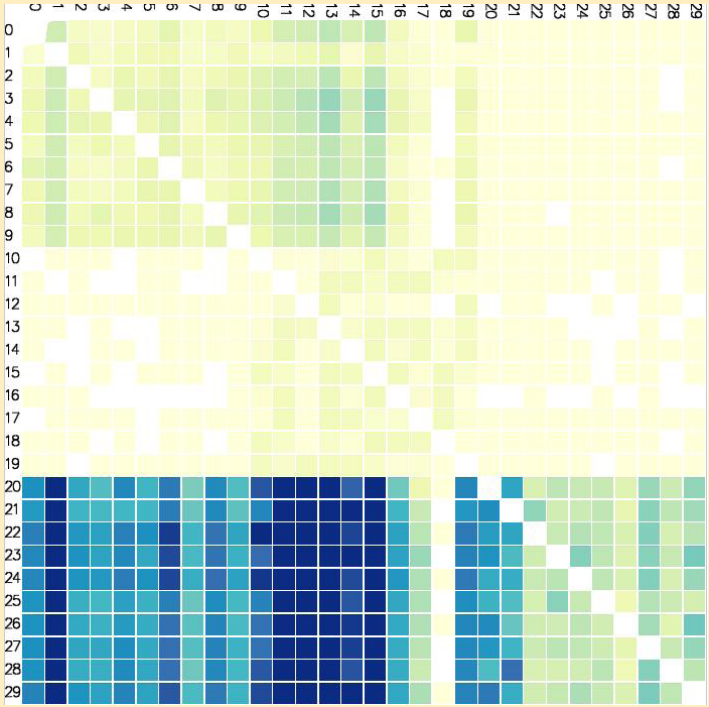


Problem structure investigations

Aid for debugging and fine-tuning of programs

- *Mindom* heuristic executed during 2 minutes and interrupted.
- Sets #1 and #2 have a strong impact on set #3.
- Weak links mislead the solver into bad decisions.

Variables × Variables

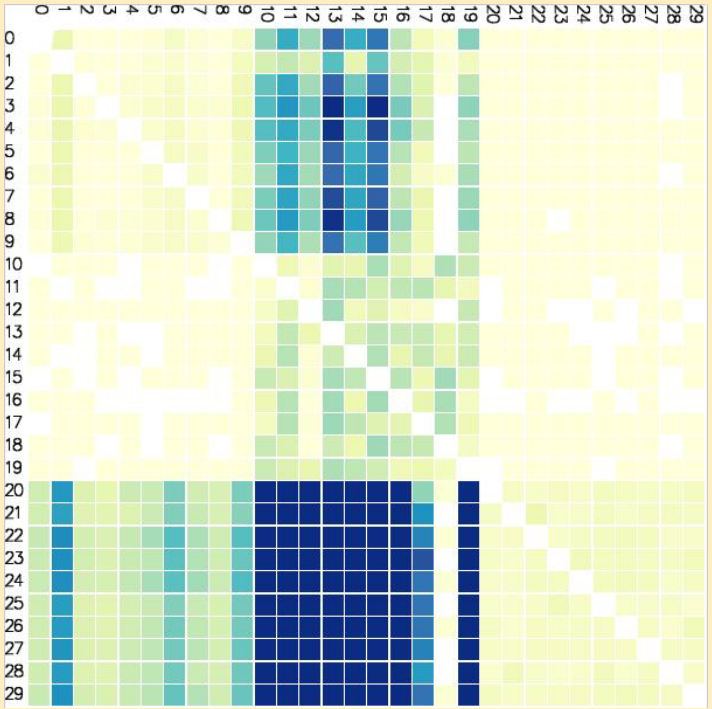


Problem structure investigations

Aid for debugging and fine-tuning of programs

- Normalized view.
- Initial infrequent decisions appear in dark.
- Poor decisions involve set #2.

Variables × Variables

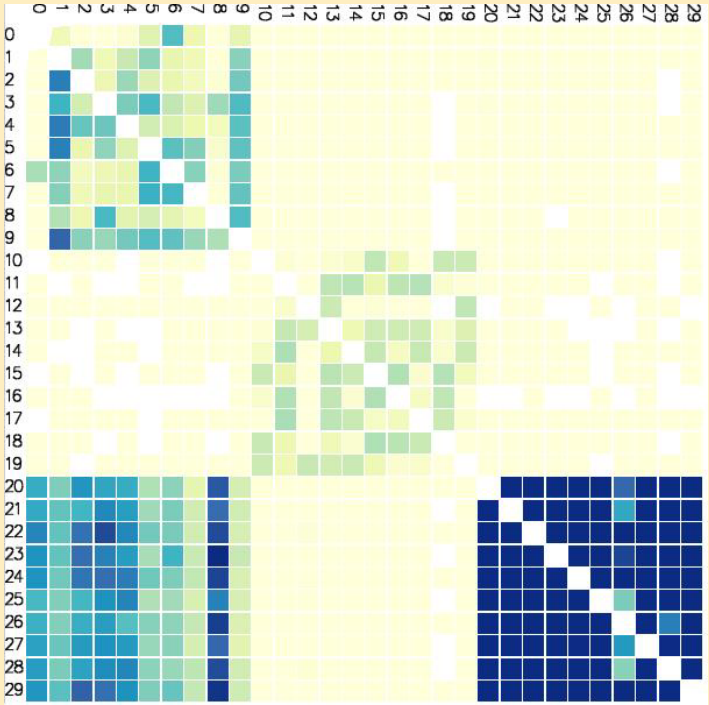


Problem structure investigations

Aid for debugging and fine-tuning of programs

- Discard gradually the effects of old decisions.
- The solver is trapped into a costly enumeration on sets #1 and #3 because of a bad decision taken wrt to set #2.

Variables × Variables

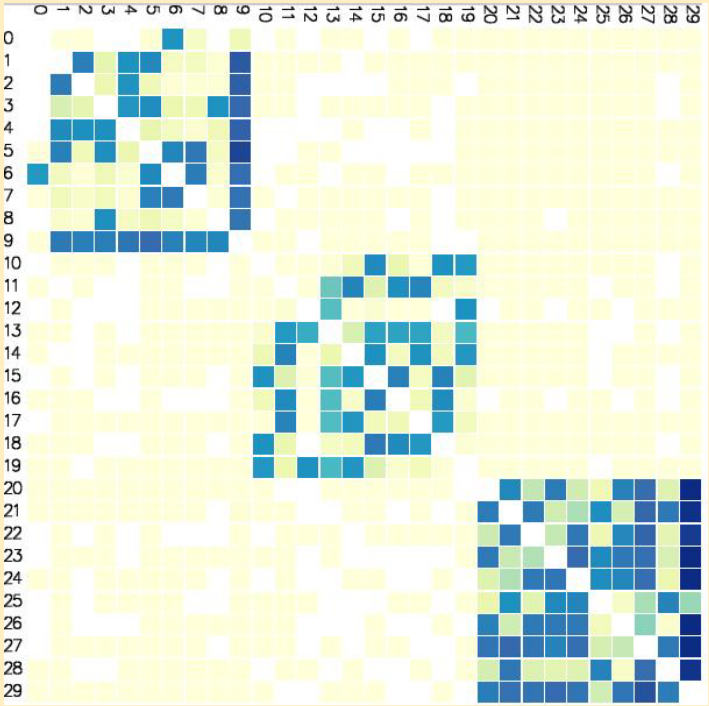


Problem structure investigations

Aid for debugging and fine-tuning of programs

- Adapt the *mindom* heuristic.
- Discard constraints causing an abnormal volume of activity.
- The problem is solved immediately.

Variables × Variables

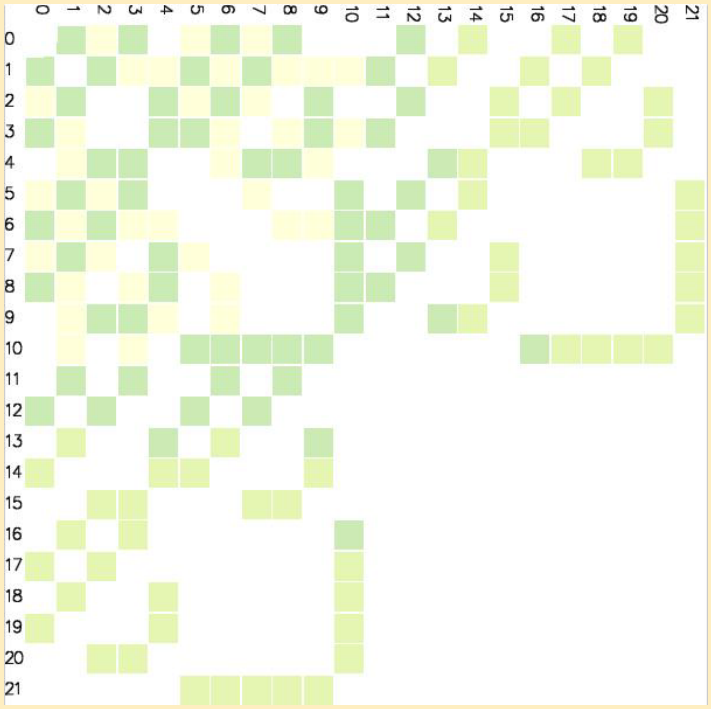


Investigation of complex problem : graph coloring

Instance 4 of Mycielski

- Variables 11 to 21 are not linked, obvious on a matrix.
- The problem can be split into two sub-problems.

Variables × Variables

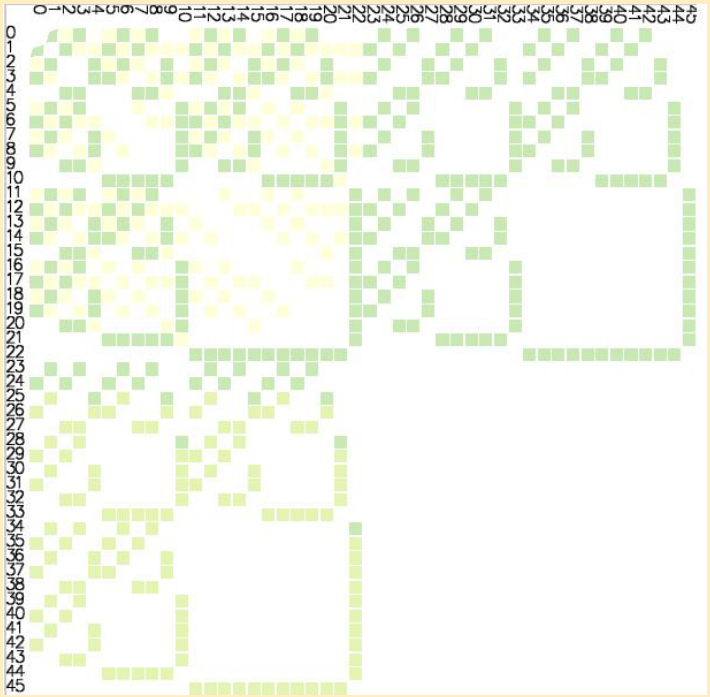


Investigation of complex problem : graph coloring

Instance 5 of Mycielski

- The structure of the problem is recurrent.
- An incremental resolution strategy can be very effective.

Variables × Variables



Conclusion

Information visualization

- The purpose of information visualization is insight.
- The choice of a visualization metaphor depends on the nature of the data and the tasks at hand.
- Large and/or dense networks are best viewed as adjacency matrices.
- Dynamic graphs are best monitored as adjacency matrices.