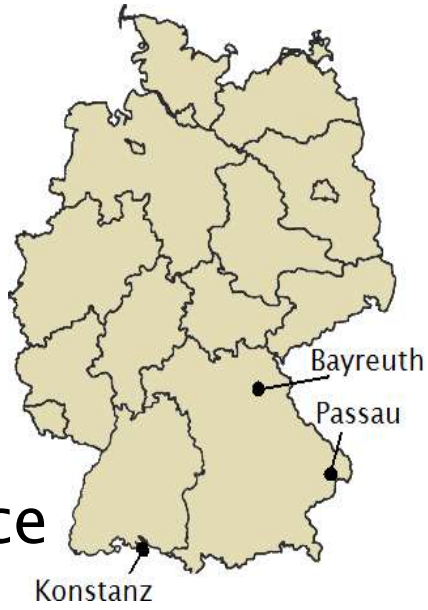
Sketchy Graph Drawing

Christian Pich
GK Summer School, St. Vigil
September 16, 2004

whoami



since 05/2004: Algorithmics group,
U. Brandes, University of Konstanz

1999-2004: *Diplom*, computer science
University of Passau

Interests:

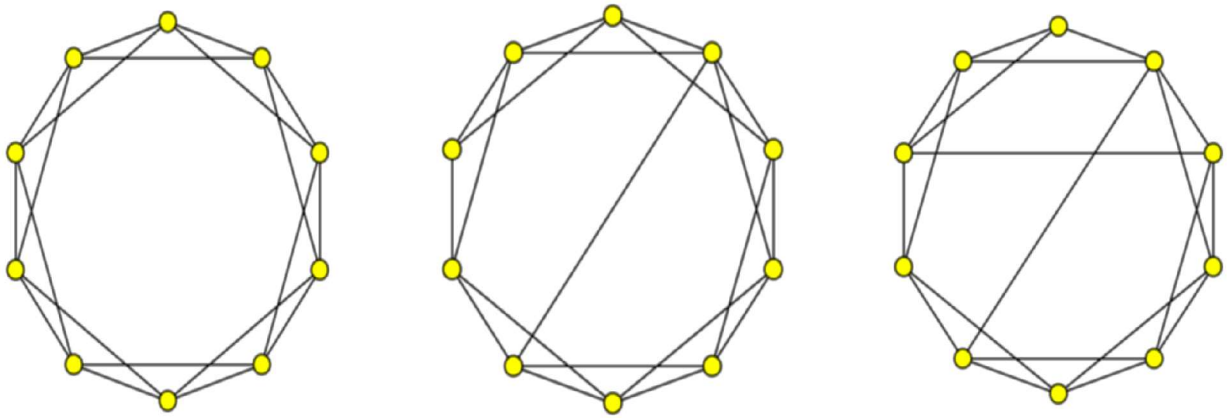
- Graph Drawing
- Graph Algorithms and Network Analysis
- Algorithmic Geometry
- Efficient Algorithms

Fun Part: Small Worlds I

Small World networks: Model for empirical networks, characterized by

- Limited node *degree* (number of acquaintances)
- Locally *dense subgraphs* (a friend of a friend is a friend)
- Low characteristic *path length* ("six degrees of separation")

Modeled by a *torus* and random *short-cuts*:

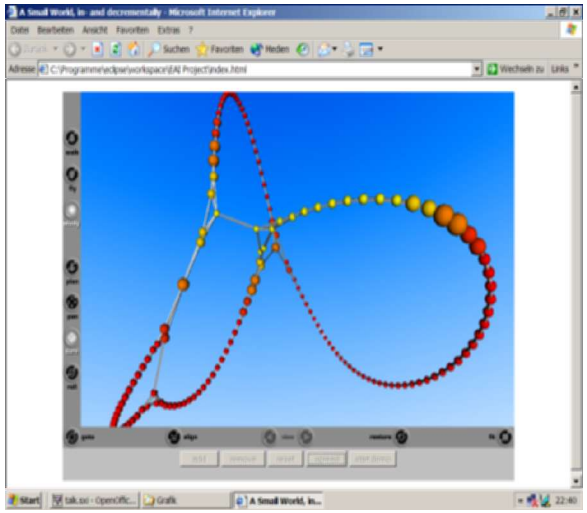


Fun Part: Small Worlds II

Six Degrees of Separation (with U. Brandes and M. Broghammer)

Installation at the Ars Electronica Festival, “Language of Networks”, Sept 1-7, Linz, Austria

http://www.aec.at/



Also see http://www.cs.virginia.edu/oracle/

Graph Drawing? Sketchy?

~~Sketchy~~ Graph Drawing:

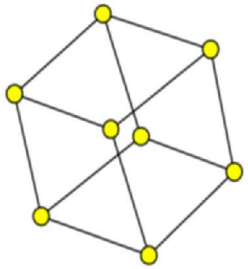
depicting a graph *G=(V,E)*

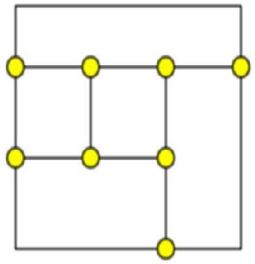
Objective: *aesthetic criteria*

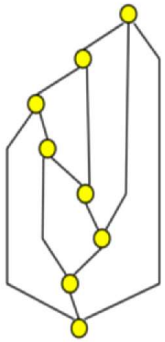
- few crossings
- display symmetries
- equal edge lenghts
- reveal graph-theoretic properties
- ...

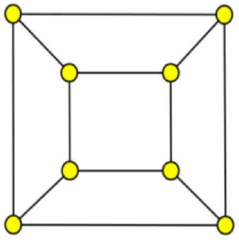
Problems:

- criteria are contradictory
- optimization problems are typically *NP*-hard
- ...









Sketchy Graph Drawing I

Problem 1:

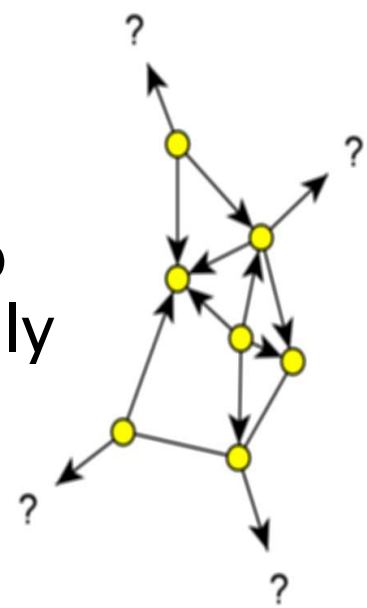
Some graphs are too large to be completely *processed* (analyzed, drawn)

- information overflow
- Runtime problems

Problem 2:

Some graphs are even too large to be *read* completely

- very very very large
- only given implicitly



Sketchy Graph Drawing II

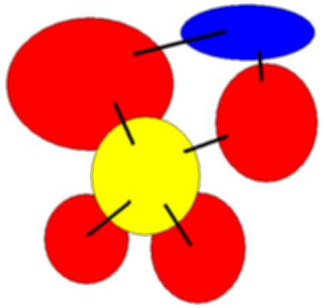
Idea:

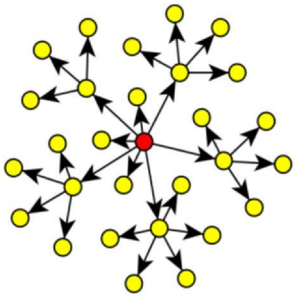
- Find a way to draw a graph *without knowing it in advance*
- Sketch the graph with what we know as yet
- Provide means for navigation (zooming into regions of interest, thereby providing smooth transitions)

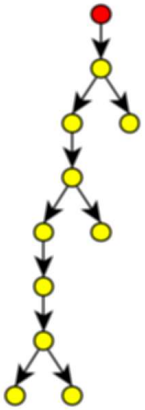
Sketchy Graph Drawing III

Issues:

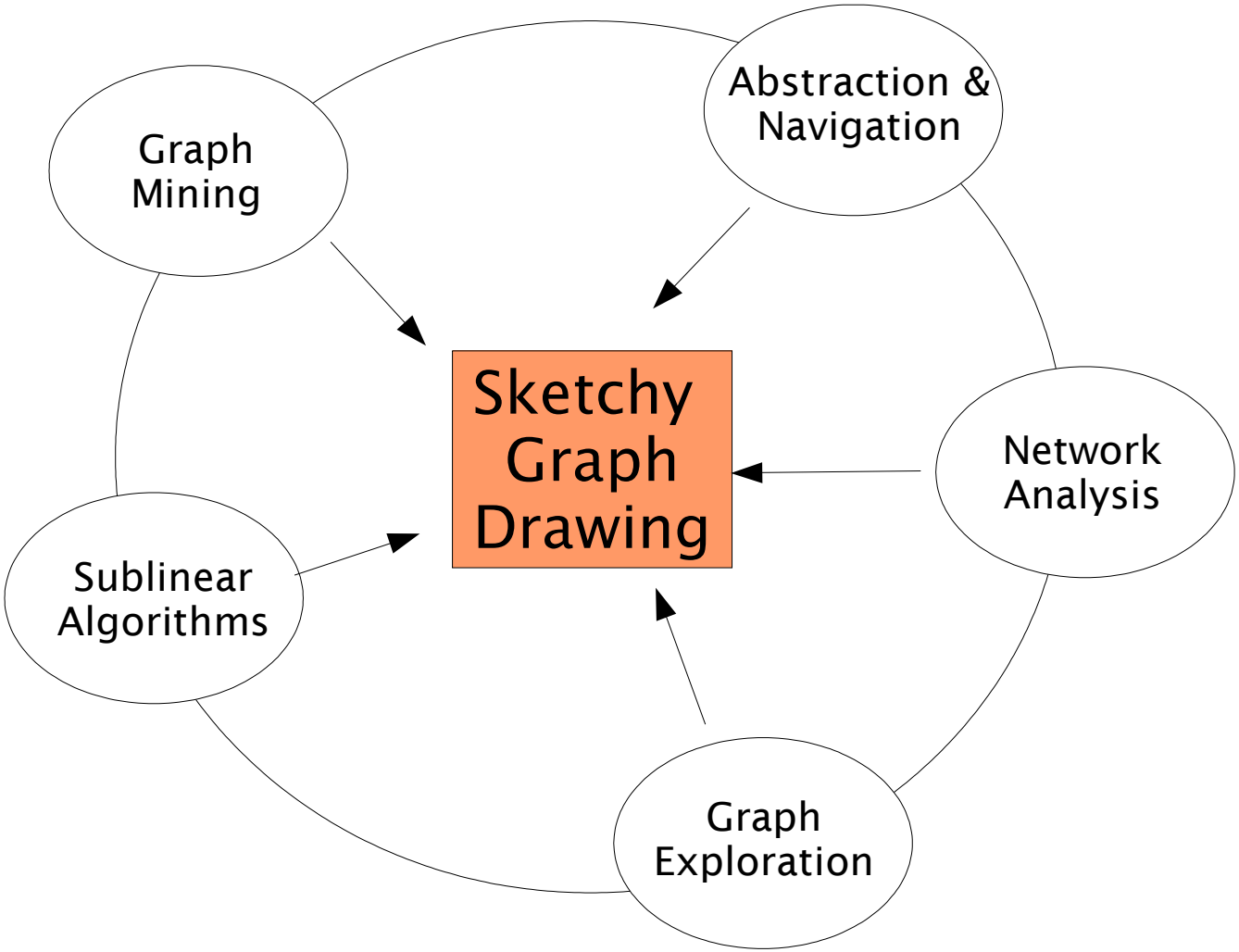
- How shall we draw a sketch? (D. Knuth, 1970: "*How shall we draw a tree?*")
- What is the best way to explore an unknown graph? DFS? BFS? A hybrid search? Something else? Use randomization?
- How can we ensure that the transitions between navigation steps be smooth?



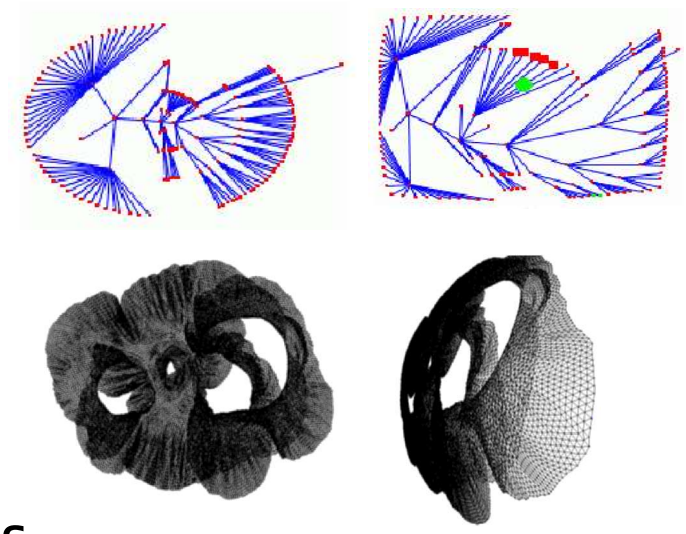




Sketchy Graph Drawing IV



Abstraction & Navigation I



Zooming
- geometric
- semantic

Focus&Context
- Fisheye Distortion
- Topological Fisheye Views

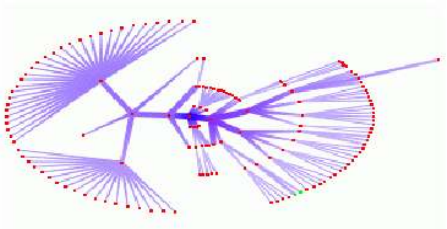
Incremental Exploration
- display only central parts of the graph
- enhance when demanded

Herman, Melançon, Marshall: Graph Visualization and Navigation in Information Visualization. *Proc. Eurographics*, 1999.

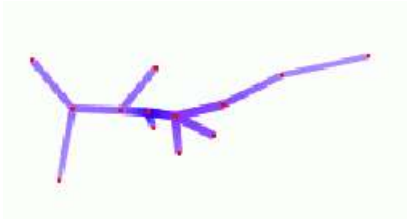
Gansner, Koren, North: Topological Fisheye Views for Visualizing Large Graphs, *IEEE Trans. Visualization and Computer Graphics,* 2004.

Abstraction & Navigation II

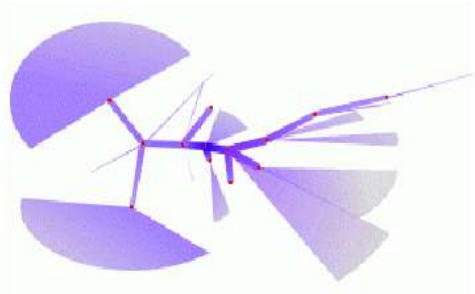
Display Abstraction in a graph:



ghosting



hiding



grouping

Decompose a graph into hierarchical layers (graph slices, graph sketches), clustering

Abello, Korn: MGV: A System for Visualizing Massive Multi-Digraphs. *Proc. IEEE InfoVis 2000*.

Kimelman et. al.: Reduction of Visual Complexity in Dynamic Graphs. *Proc. Graph Drawing 1993*.

Graph Exploration I

Original Problem: Complete traversal of an *unknown graph*/environment, motion planning

- Create a map of the partial traversal from a given starting node

- Use low number of edge traversals
- Variation: tethered to a rope

- Variation: limited fuel capacity

It's an *on-line problem*: We know nothing about the graph beforehand, but we gain knowledge

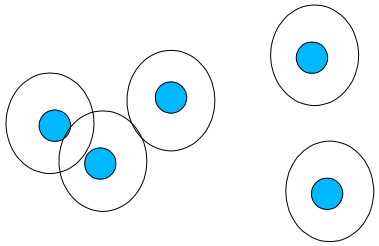
Deng, Papadimitriou: Exploring an Unknown Graph. *Proc. 31th IEEE FOCS*, 1990.
Duncan, Kobourov: Optimal Constrained Graph Exploration. *Proc. SODA 2001.*

Graph Exploration II

Search Strategies
- Breadth First
- Depth First
- Best First
- A*
- ...

Relaxation
- Only *partial* traversal, but
- sufficient amount of information to work with

Parallel Graph Algorithms
- pivot set of starting nodes
- combine the partial results



Quinn, Deo: Parallel Graph Algorithms, *ACM Computing Surveys* 16(1), 1984.

Sublinear Time Algorithms I

Often even O(n) is too slow

- very large graphs ("large information spaces")
- can/want to view only at a small fraction of the input

Goal: Sublinear Time Algorithms

- O(log n), O(sqrt n), independent of n
- typically: use sampling, randomization
- relaxation of an exact problem to an approximation, estimation

(sublinear *memory*: streaming algorithms)

Kumar, Rubinfeld: Algorithms Column: Sublinear Time Algorithms. *ACM SIGACT News 43(4)*, 2003.

Sublinear Time Algorithms II

Application: Property Testing

- Exact problem: yes/no
- Relaxated problem: is the object ε-*far* from having a property, with constant probability?

Example: Is the list $x=\{x1,\ldots,xn\}$ ε-far from being monotone with prob. 2/3, i.e. does it have an increasing subsequence of length $(1-\varepsilon)n$?

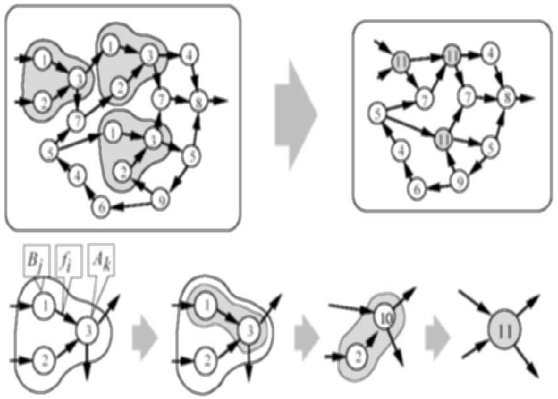
- Select i at random and perform ordinary binary search as if to find xi
- Repeat $O(1/\varepsilon)$ times, report FAIL if the search does not find xi. Eventually report PASS.
- Running time in $O((1/\varepsilon) \log n)$.

Ron: Property Testing – A Tutorial. In *Handbook of Randomized Computing*, 2001.

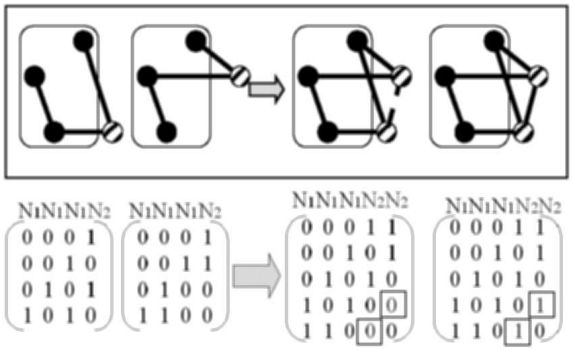
Graph Mining

Find frequent relevant subgraphs in a large graph

Uses methods from Classical Data Mining (finding regularities, basket analysis)



Graph compression



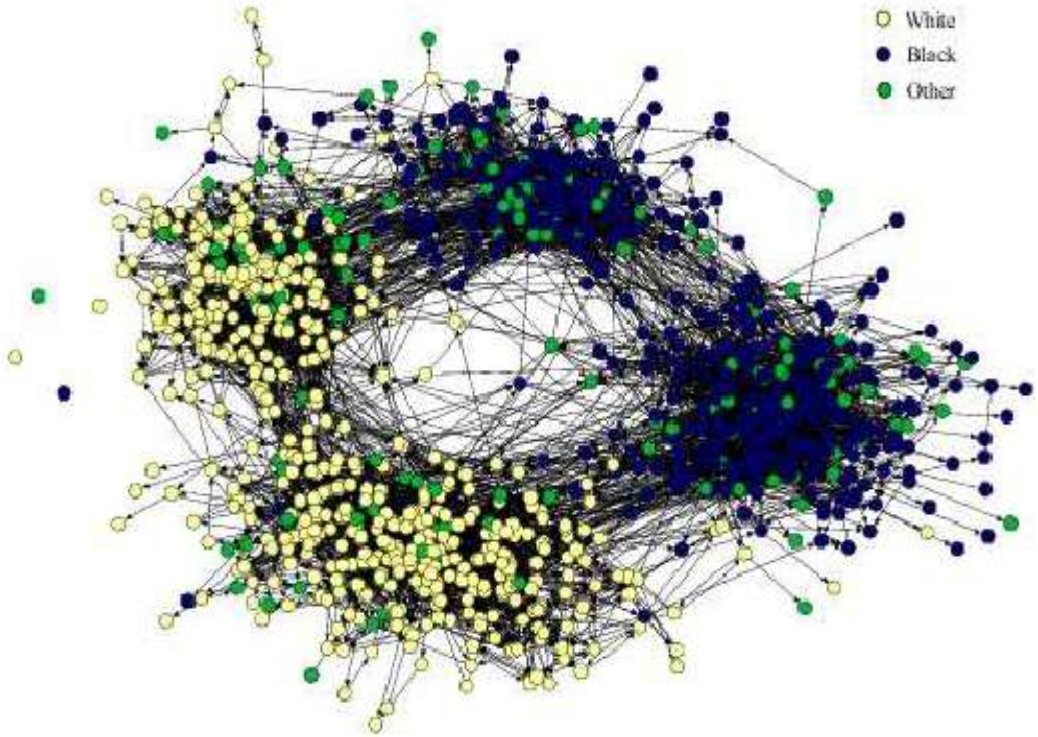
Frequent subgraphs

Washio, Motoda: State-of-the-Art of Graph-based Data Mining, SIGKDD Explor. News, 2003.

Network Analysis I

Friendships
in a US school

[Picture by James Moody]

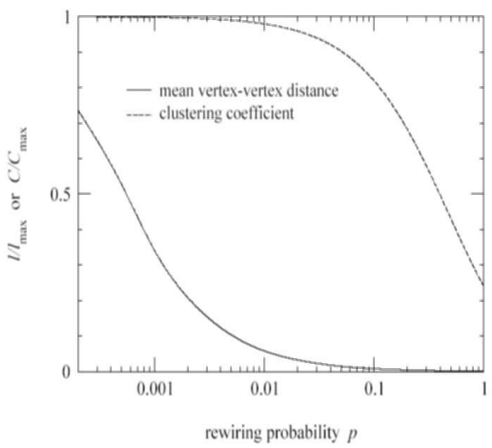


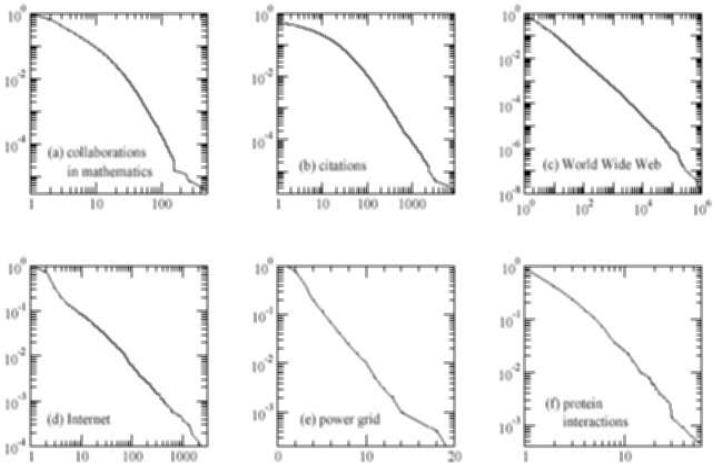
Find out:

- Who is the most important actor in a network? Who is central? Who isn't? Why?
- Are there clusters?
- How did the network evolve over time?
- What will it look like in the future?

Network Analysis II

Graph models, network statistics





Newman, 2003: *"How can I tell what this network looks like when I can't actually look at it?"*

Newman: Structure and Function of Complex Networks. *Computer Physics Communications*, 2003.

Summary

Sketchy Graph Drawing should employ methods from manifold areas, including

- Visual Abstraction and Navigation, Multilevel Visualization
- Graph Exploration and Network Searching
- Sublinear Time Algorithms
- Graph Mining
- Network Analysis

Thanks for your attention!