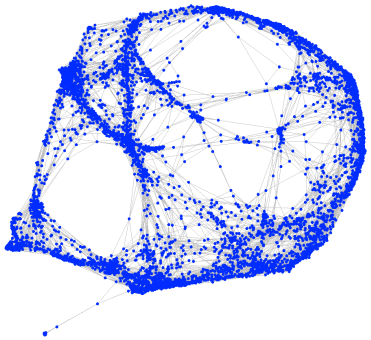
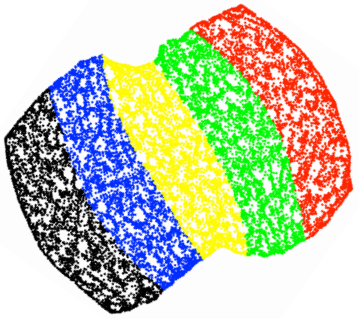


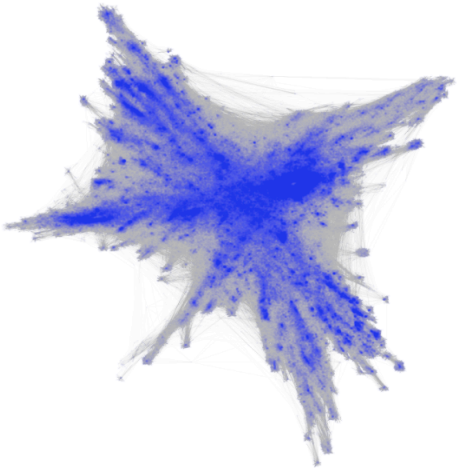


OpenOrd: An Open-Source Toolbox for Large Graph Layout

S. Martin[1], W. M. Brown[2],
R. Klavans[3], and K. Boyack[3,*]

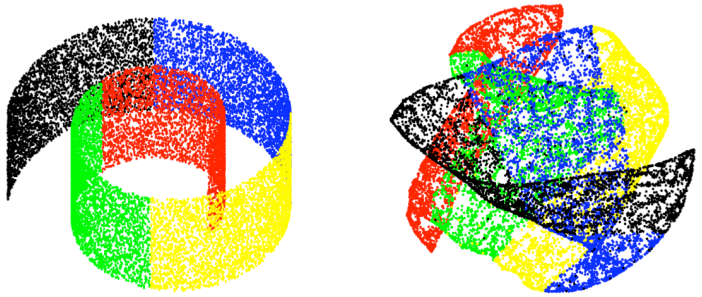
[1]Sandia National Laboratories, Albuquerque, NM
[2]Oak Ridge National Laboratories, Oak Ridge, TN
[3]SciTech Strategies, Inc., Berwyn, PA

*Presenter



Motivation for OpenOrd Toolbox

- Force-directed layout doesn't scale well to large graphs
  - Computational complexity $O(n^2)$
  - Poor global structure



  - Visually unappealing "hairball"

- Most existing algorithms don't work well on "real-world" data, e.g. not mesh, not scale-free, etc.

Basic Force-Directed Graph Layout

- Force-directed layout optimizes

$$\min_{\mathbf{x}_1,\ldots,\mathbf{x}_n} \sum_i \left(\sum_j \left(w_{ij} d(\mathbf{x}_i, \mathbf{x}_j)^2 \right) + D_{\mathbf{x}_i} \right),$$

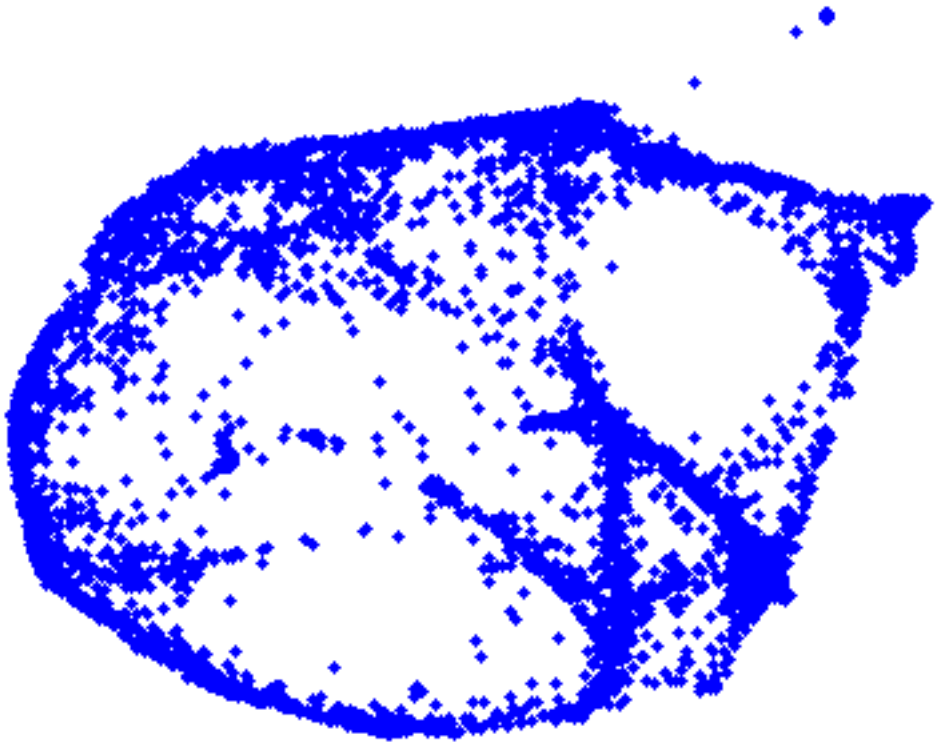
  where $\mathbf{x}_i$ are positions of vertices, w_{ij} are edge weights and $D\mathbf{x}_i$ is the density of edges near $\mathbf{x}_i$.
  - Large edge weights encourage vertices to group together.
  - High density is discouraged.

- OpenOrd is based on predecessor VxOrd and uses simulated annealing to solve this problem, with a five stage cooling schedule (liquid, expansion, cool-down, crunch, simmer).

- Density term $\mathrm{D}\mathbf{x}_i$ is costly, so we use a grid-based method to reduce computation from $O(n^2)$ to $O(n)$.

Basic Force-Directed Graph Layout

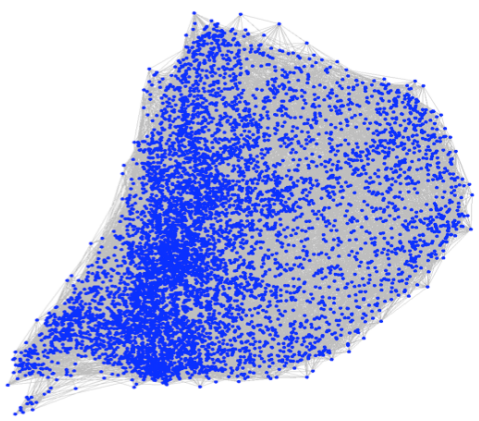
- Movie showing simulated annealing schedule.

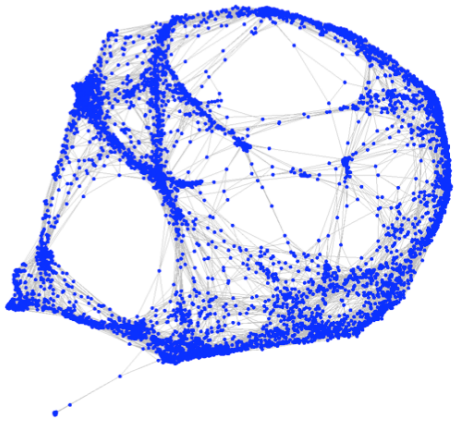


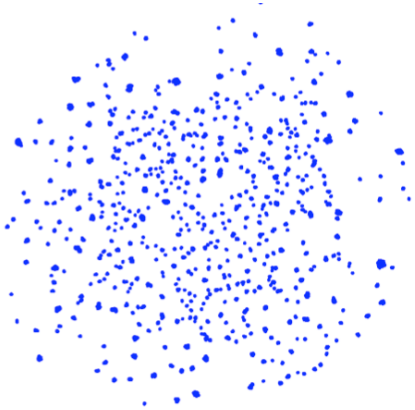
Liquid
Expansion
Cooldown
Crunch
Simmer

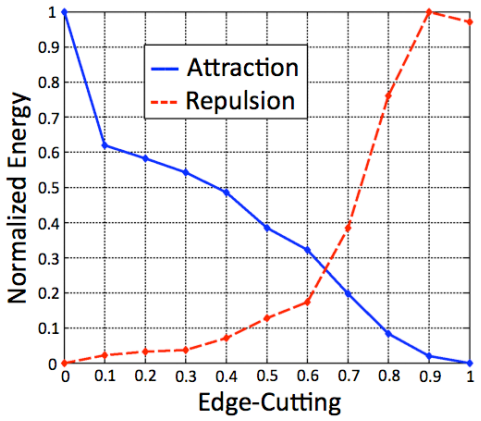
Edge-Cutting

- OpenOrd uses an edge-cutting heuristic in order to provide user control of amount of white space in layout.
  - Edges are cut if they are both long (in layout) and large weight.
- Edge-cutting allows trade-off between attractive $w_{ij}d(\mathbf{x}_i,\mathbf{x}_j)^2$ term and repulsive $D\mathbf{x}_i$ term in optimization.







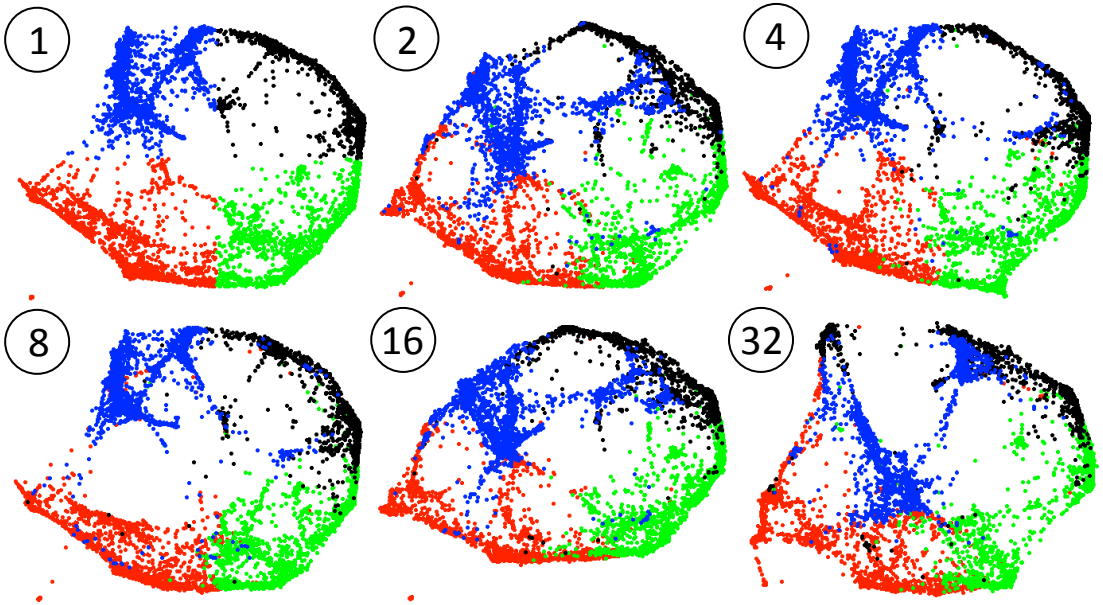


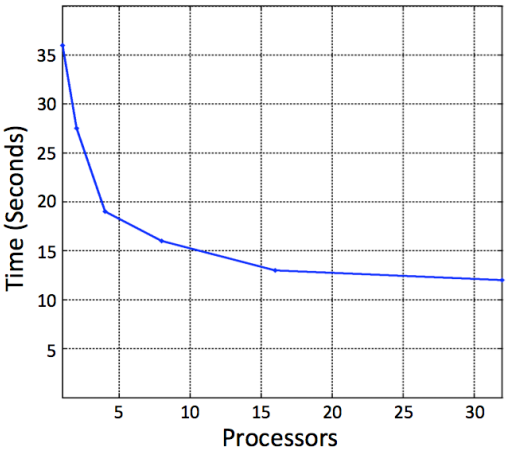
less ——— edge-cutting ——— more →

trade-off

Parallel Layout

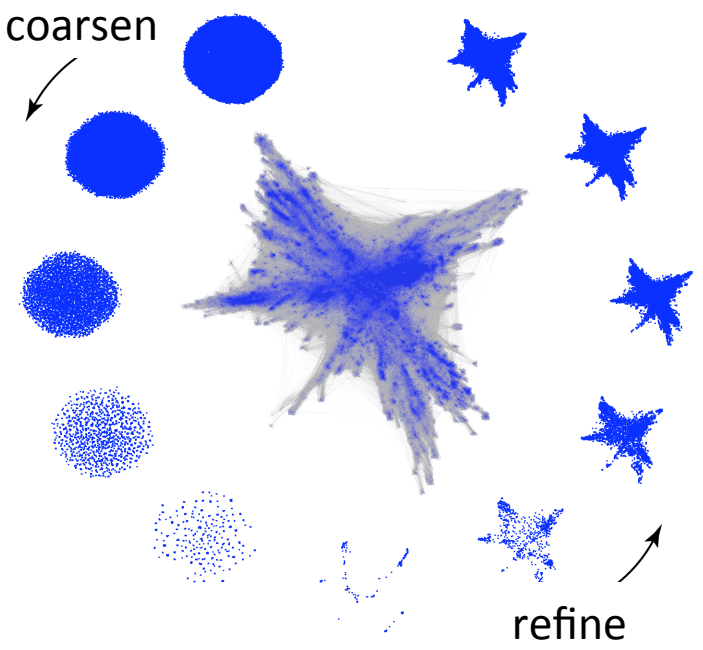
- OpenOrd can be run in paralell: each processor keeps track of all vertex positions, but is only responsible for moving a subset of vertices.
  - Maintains similar layout.
  - Increases effective memory of computer for truly enormous graphs.
  - Increases speed.



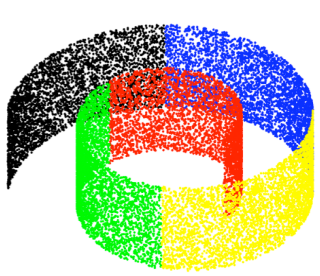


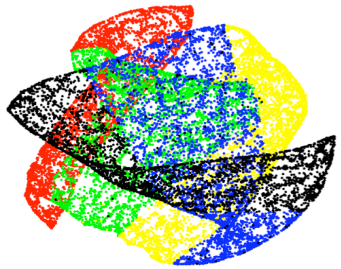
Multi-Level Layout

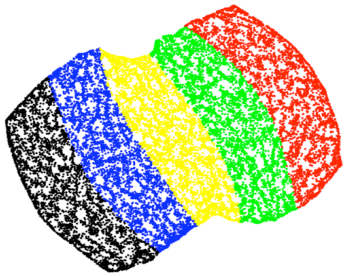
- Multi-level layout: cluster vertices, coarsen, repeat, layout, refine, repeat.



- OpenOrd uses multi-level layout to untangle global structure.

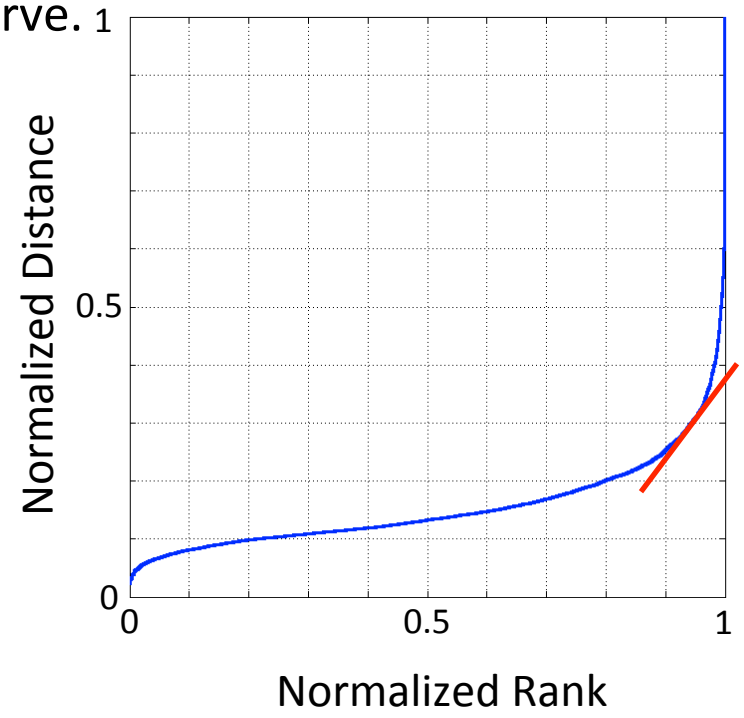

original data


single level layout


multi-level layout

Average-Link Clustering

- When we coarsen the graph for multi-level layouts, we use average-link heirarchical clustering.
  - Clustering is based on distance in a layout with maximum edge-cutting.
  - A distance threshold is chosen based for forming clusters.
  - Distance threshold can be chosen manually or automatically using a normalized curve.



Parameter Testing

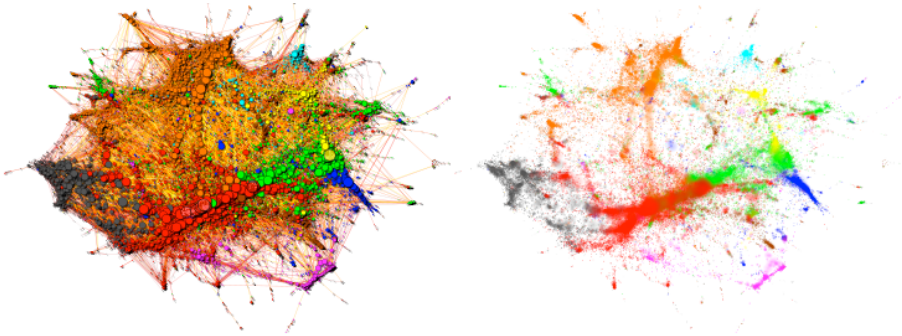
- OpenOrd has many parameters that must be chosen in order to function, such as edge-cutting and layout level.
- We identified good defaults using a variety of datasets as test cases.

Parameter Testing

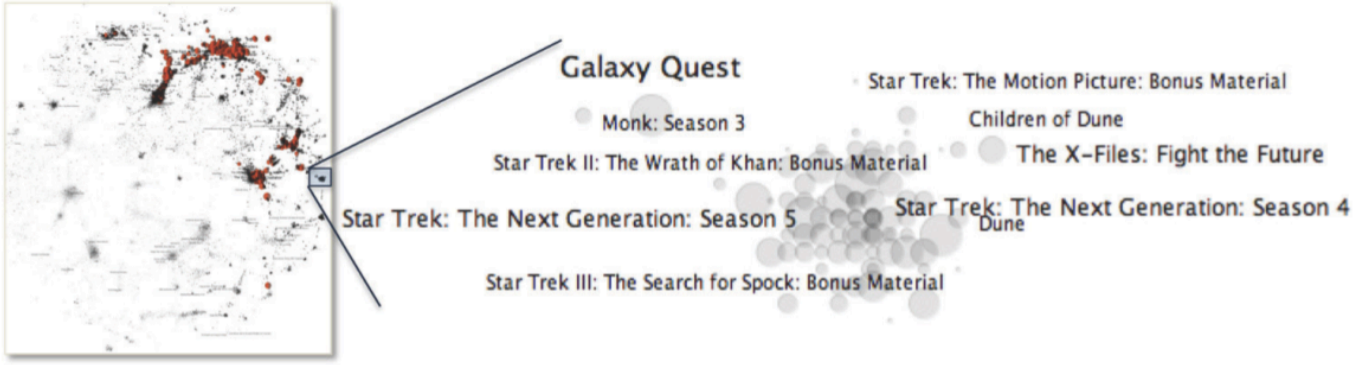
Dataset	Nodes	Edges	Level	Time
Yeast	6,147	61,646	3	1:29
Journals	8,712	98,705	3	2:13
Swiss Roll	20,000	400,000	9	4:01
Solid State Lighting	32,776	222,626	4	7:16
Quarter Year ISI 2003	218,716	1,821,976	5	1:09:36
Wikipedia	659,388	16,582,426	6	3:39:23
Full Year ISI 2004	849,888	5,843,729	7	3:40:23

Some Real-World Examples

- Map of Last.fm music database by Tamas Nepusz (Royal Holloway University)



- Netflix Movie Database by Todd Holloway (Indiana University)



- Maps of Science (lots of variations) by R. Klavans and K. Boyack

Conclusions

- OpenOrd combines many of the state-of-the-art ideas in large-scale graph layout in an easy-to-use open source package.
  - Edge-cutting (novel to OpenOrd and predecessor VxOrd)
  - Parallel operation.
  - "Smart" (using average-link clustering) multi-level graph partitioning.
- OpenOrd (and predecessor VxOrd) has been used successfully in a number of applications.
  - Scientific literature analysis.
  - Bioinformatics applications.
  - Music databases.
  - Movie databases.
- OpenOrd is available at www.cs.sandia.gov/~smartin.