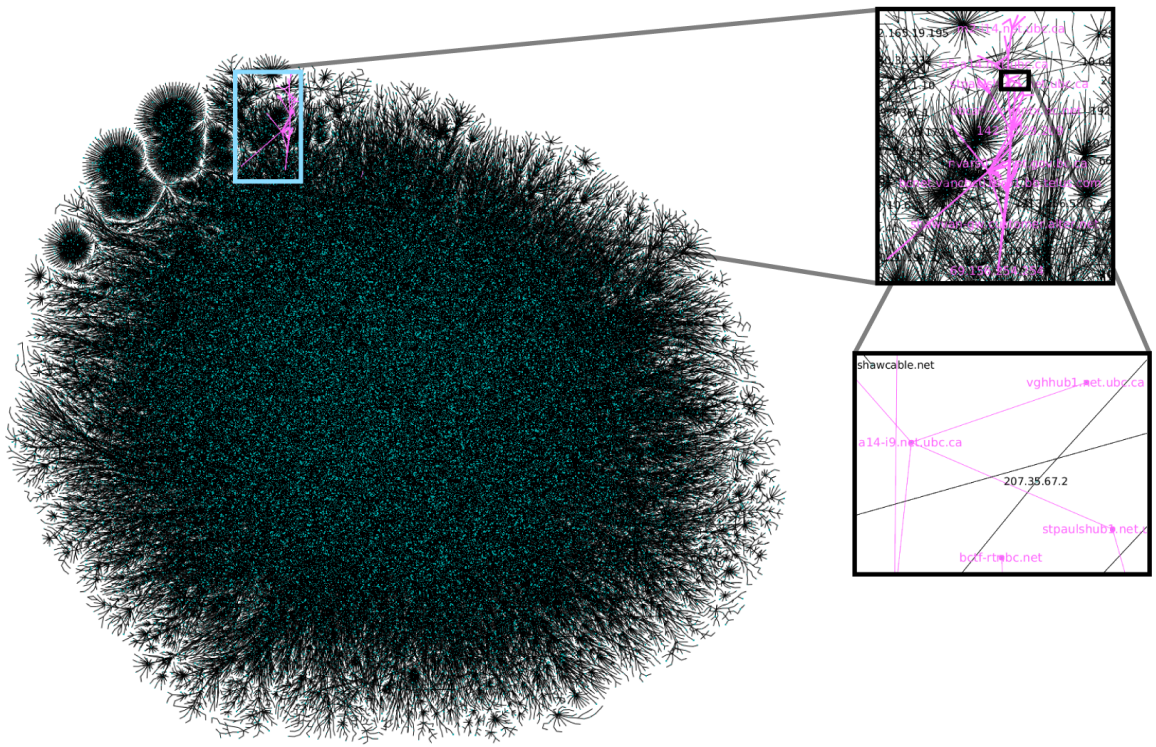
# TugGraph: Path-Preserving Hierarchies for Browsing Proximity and Paths in Graphs

Daniel Archambault[1,2], Tamara Munzner[2], and David Auber[1,3]

[1]GRAVITE, INRIA Bordeaux Sud-Ouest, France
[2]University of British Columbia, Canada
[3]CNRS UMR 5800 LaBRI, Université de Bordeaux I, France

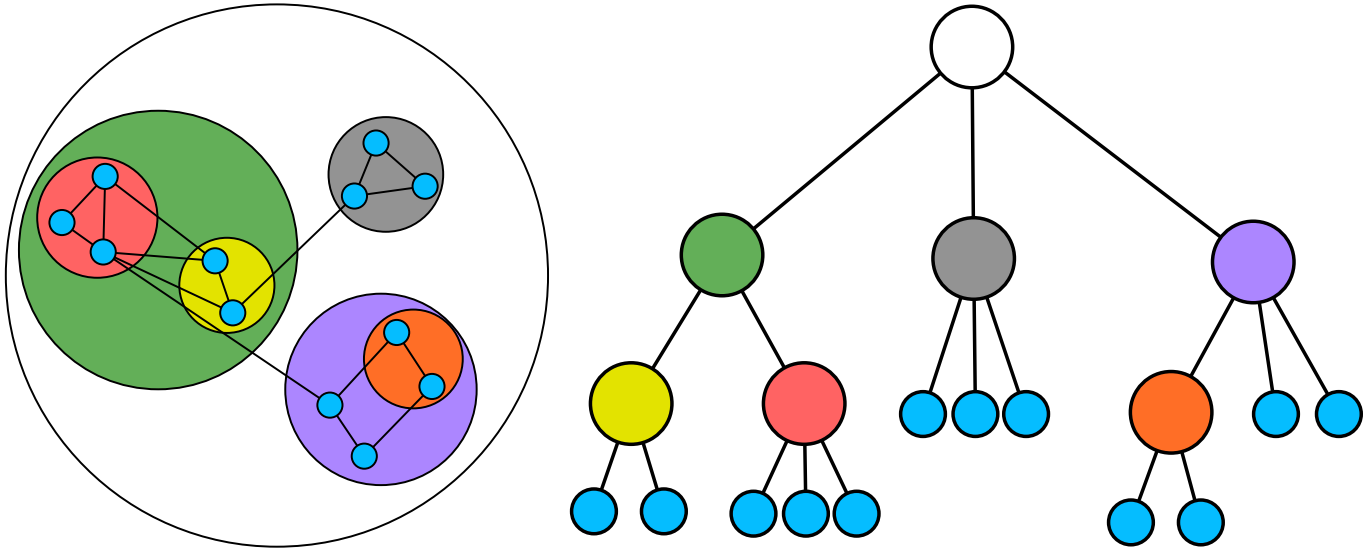
# Drawing Large Graphs




(a) LGL: 12hrs

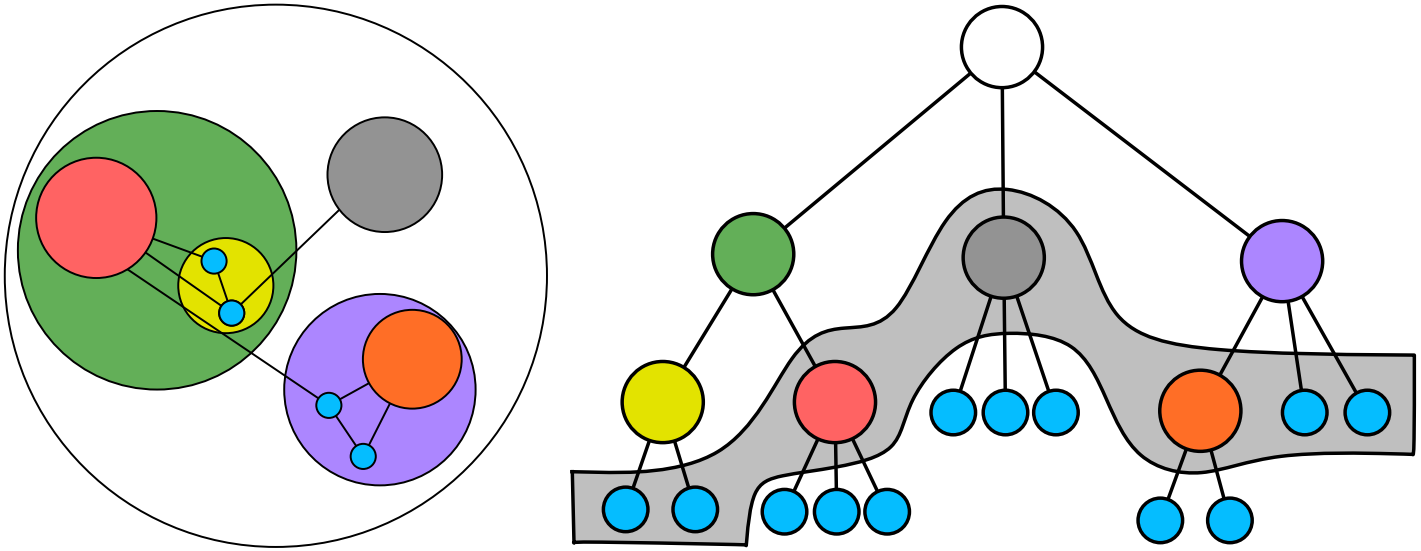
- Drawing algorithms have high asymptotic complexity
- Layouts suffer from visual clutter

# Hierarchy of Coarse Graphs



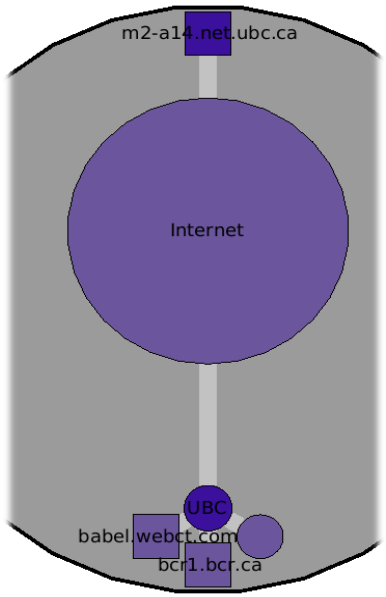
- Usual approach: decompose the graph recursively
  - a subgraph is replaced by a single **metanode** at its parent
  - process is repeated on graph forming a cluster tree
- The structure is a **multi-level hierarchy** in previous work

# Hierarchy Cut

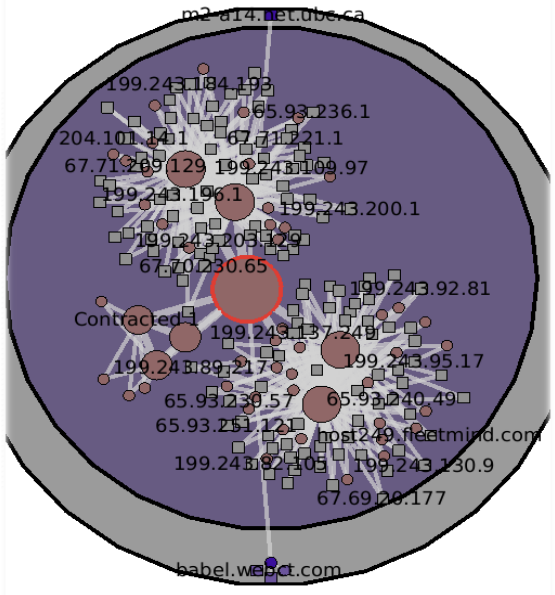


- A **cut** defines which nodes are visible or hidden
  - nodes on and above the cut are visible in the graph view
- Parts of graph can be drawn on demand
- foundation for steerable exploration of a graph
  - drawing algorithm only applied to selected regions

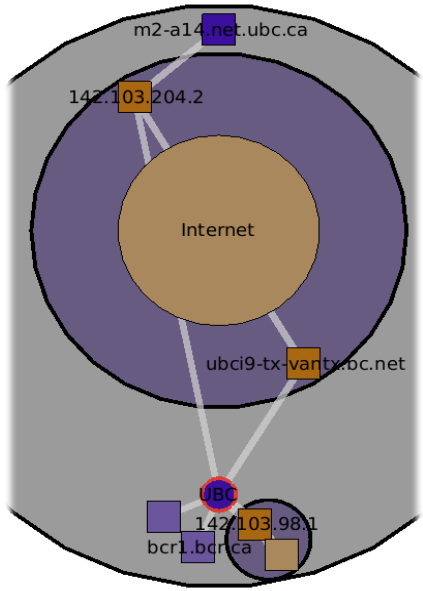
# Limitations of Steerable Graph Drawing




(c) Large Metanode



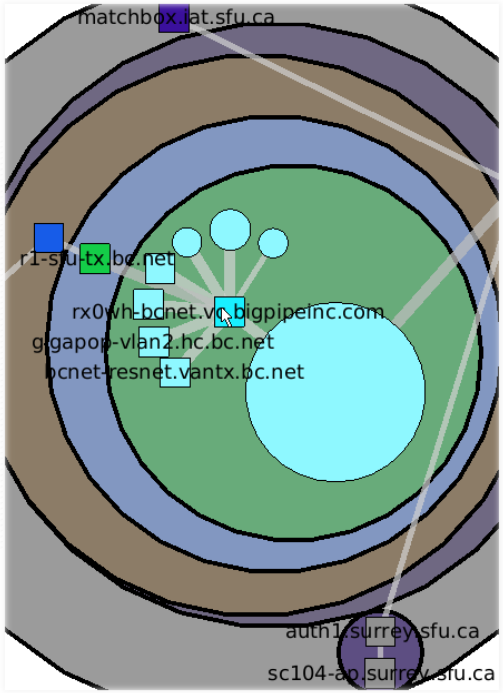

(d) Previous Work



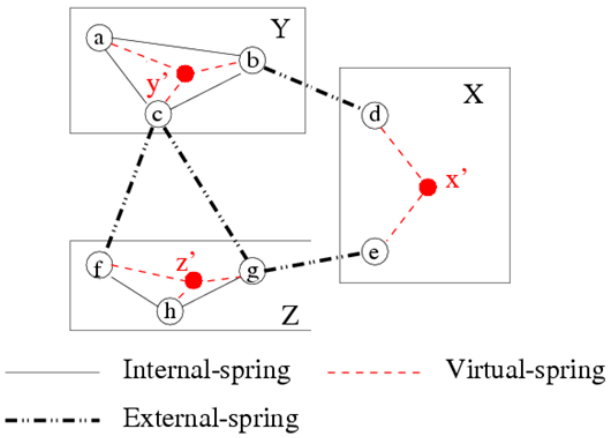

(e) TugGraph

- **Steerable** multi-level graph drawing
  - drawing computed on demand
- Problem: opening large metanodes (+190,000 nodes)
  - even automatic coarsening not adequate
- Solution: tug out structure nearby a small, interesting subgraph

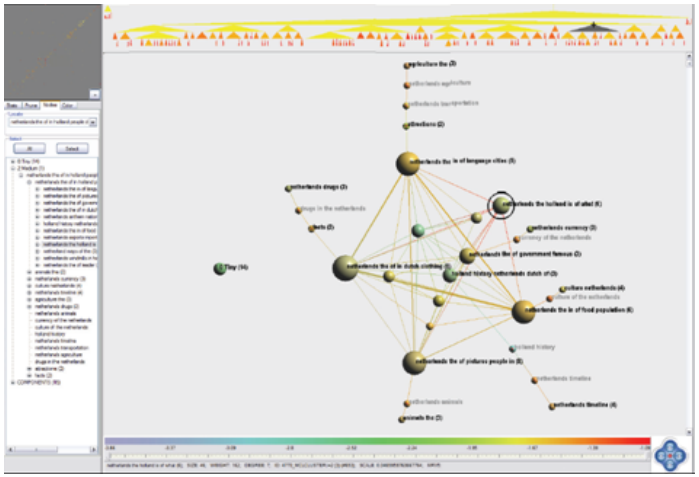
# Video


matchbox.iat.sfu.ca
r1-sfu-tx.bc.net
rx0wh-bcnet.vc.bigpipeinc.com
gigapop-vlan2.hc.bc.net
bcnet-resnet.vantx.bc.net
auth1.surrey.sfu.ca
sc104-ap.surrey.sfu.ca

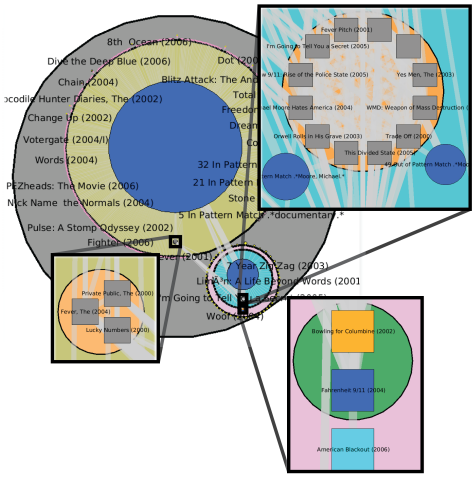
# Steerable Graph Drawing Systems




(a) DA-TU, 2000



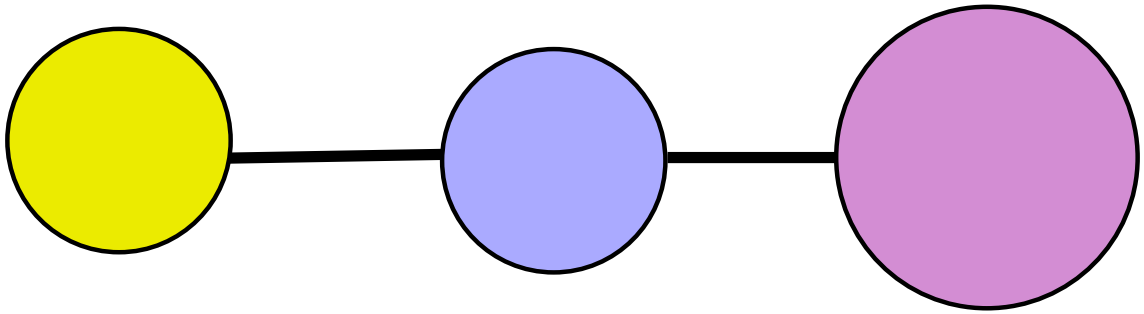
(b) ASK-GraphView, 2006



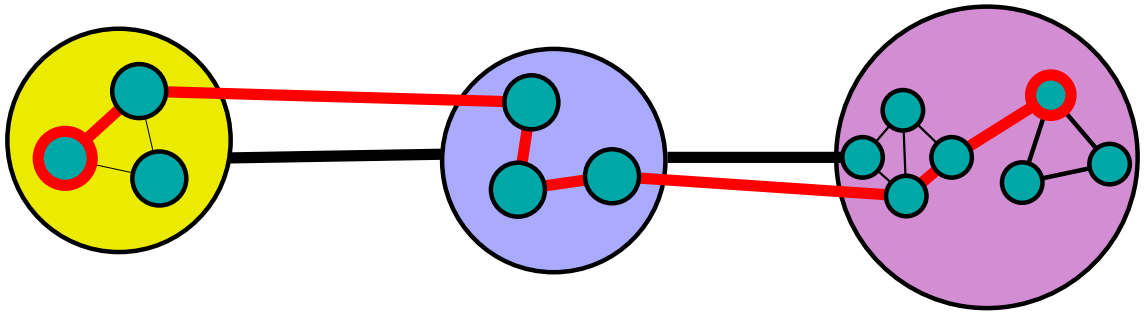
(c) GrouseFlocks, 2008

- Explore hierarchy by drawing metanodes on demand
- Hierarchies created by subgraph selection in input graph
- Limitation:
  - Interaction does not take into account subgraph of interest

# Path-Preserving Hierarchy



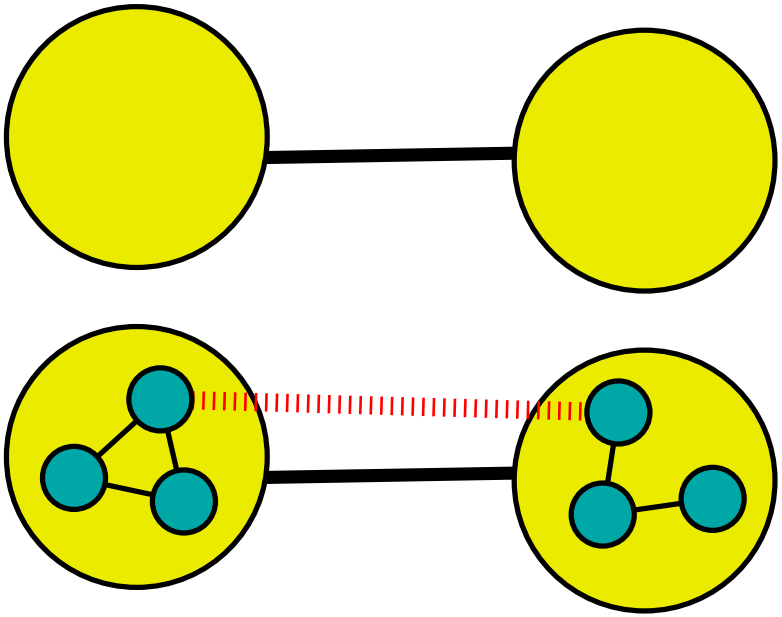
(a)



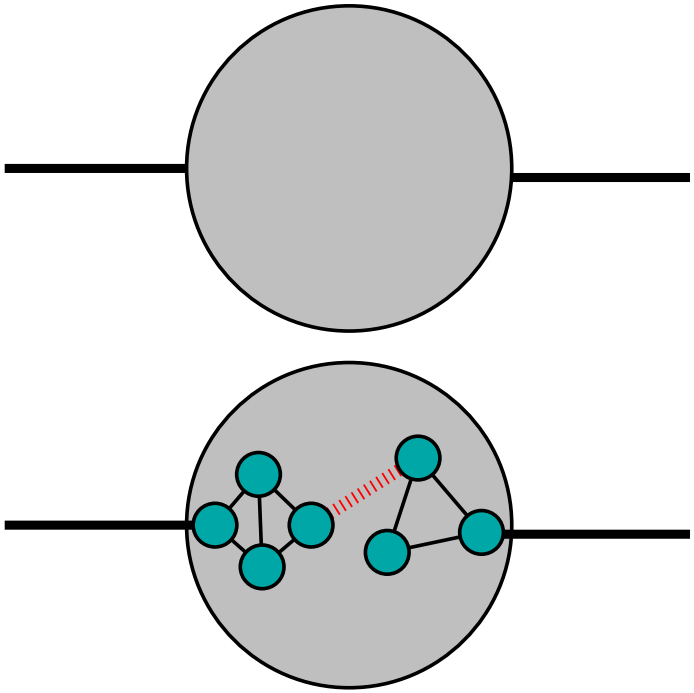
(b)

- Defined in GrouseFlocks work
- A path in the hierarchy means at least one path in the graph
- **Path-preserving hierarchies** respect this property

# Path-Preserving Hierarchy



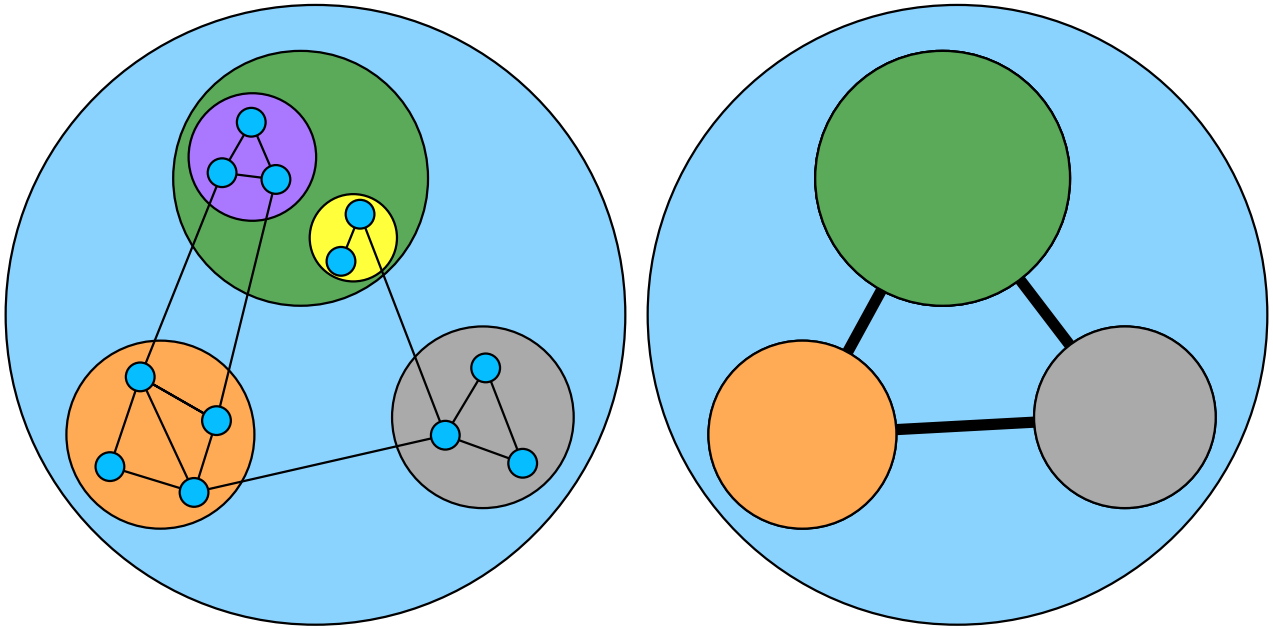
(a) Edge Conservation



(b) Connectivity Conservation

- Metaedge if and only if a pair of descendants connected
- Metanodes contain connected subgraphs
- If preserved, paths in cuts are also in underlying graph

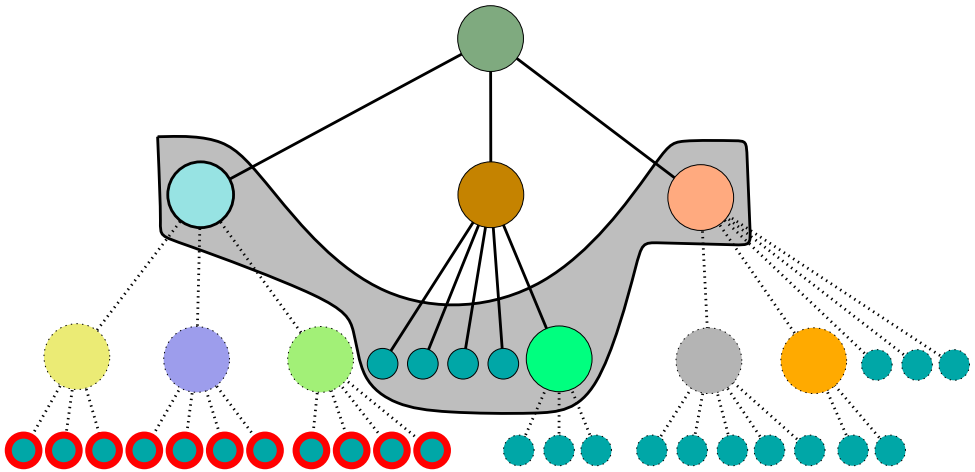
Hierarchies that are not path-preserving can be misleading



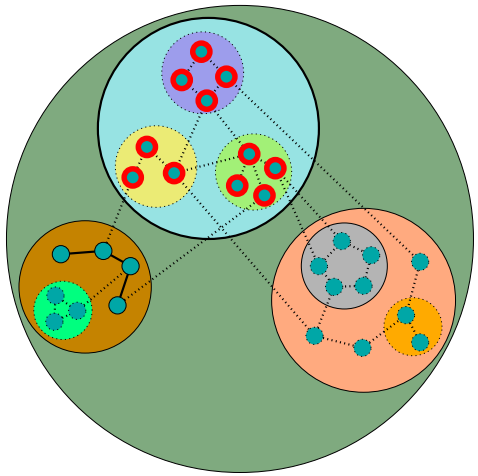
(a) Invalid Hierarchy (b) Hierarchy Cut

- Cycles can appear when there are not cycles present.

# Algorithm Step Selection



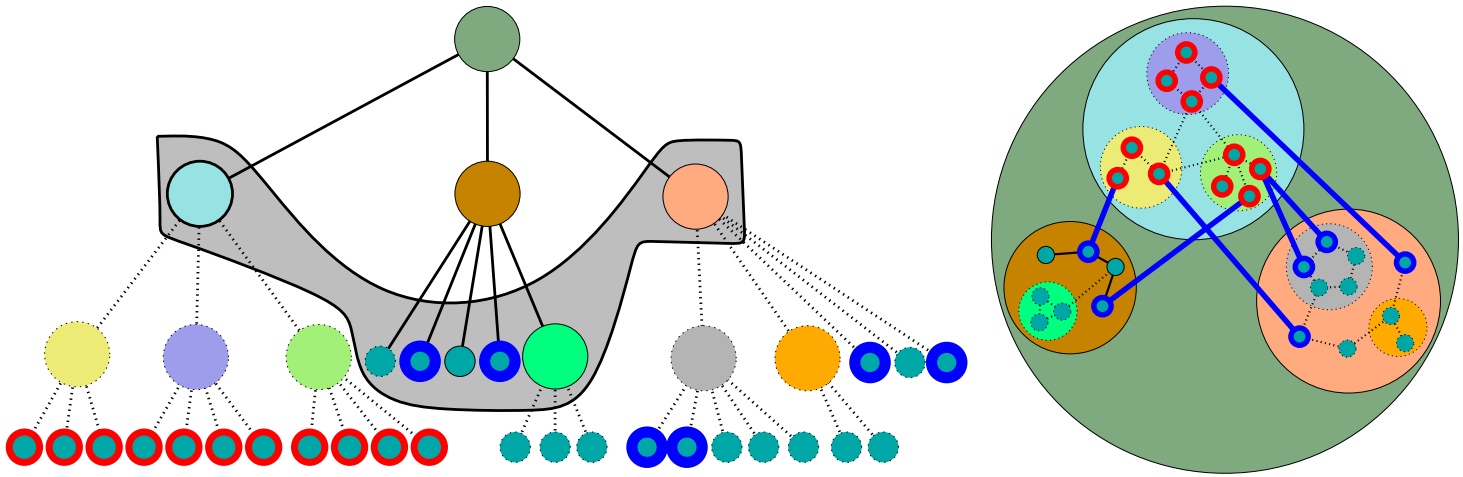
(a) Hierarchy



(b) Hierarchy

- User selects a node
- All nodes in input graph are selected

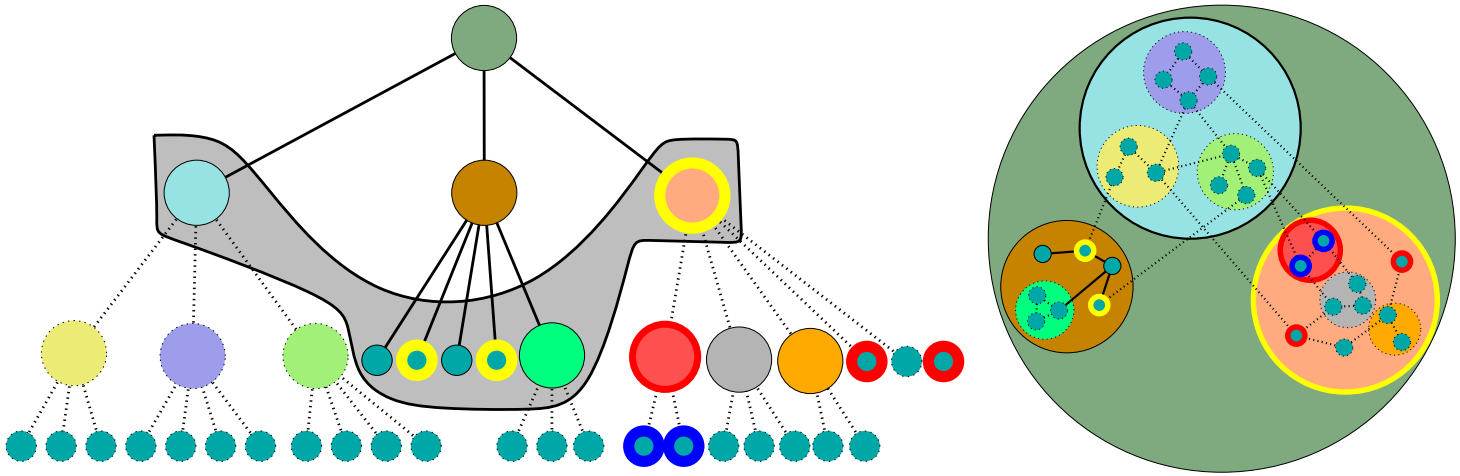
# Algorithm Step Select Adjacent



(a) Hierarchy (b) Graph

- Adjacent nodes to the selection are chosen

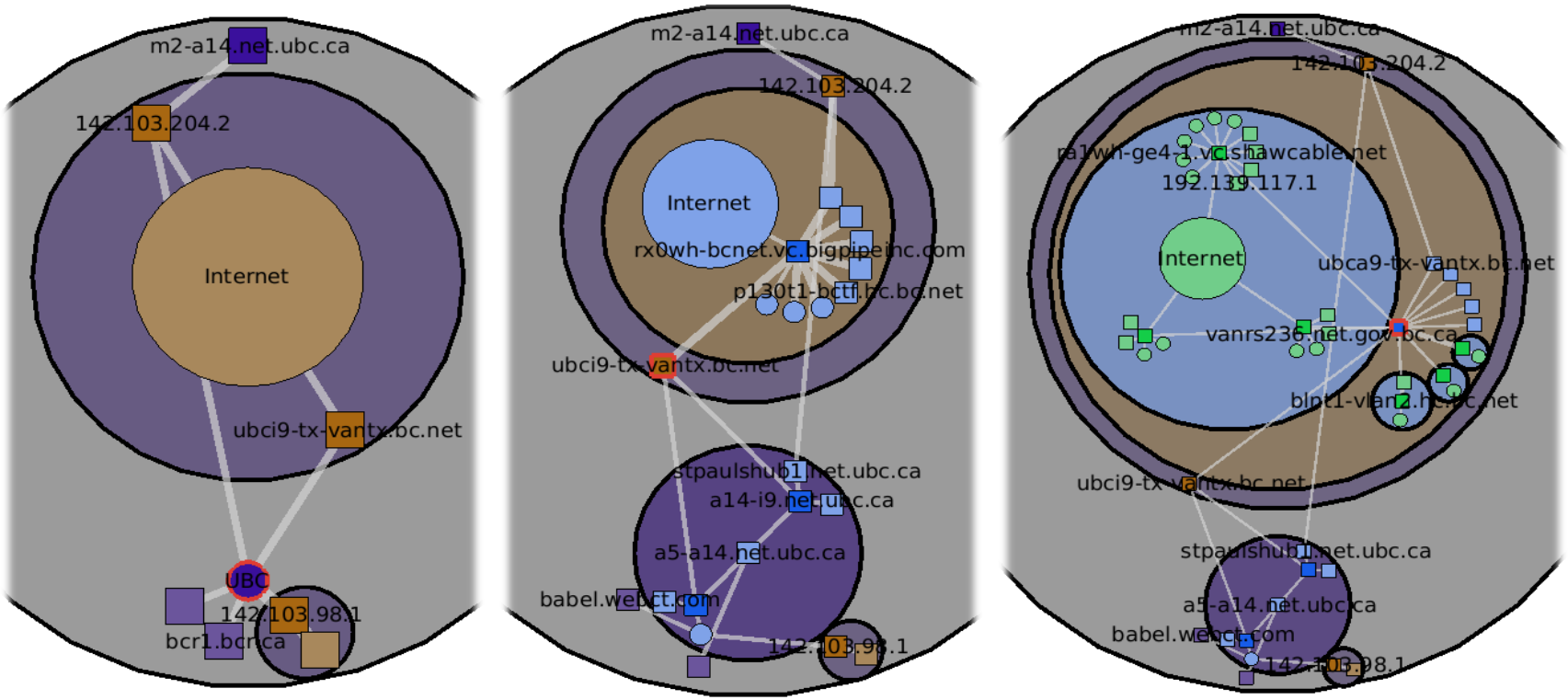
# Algorithm Step Decomposition



(a) Hierarchy (b) Graph

- Hierarchy reformed according to selection set
- Nodes distance one away are selected in red

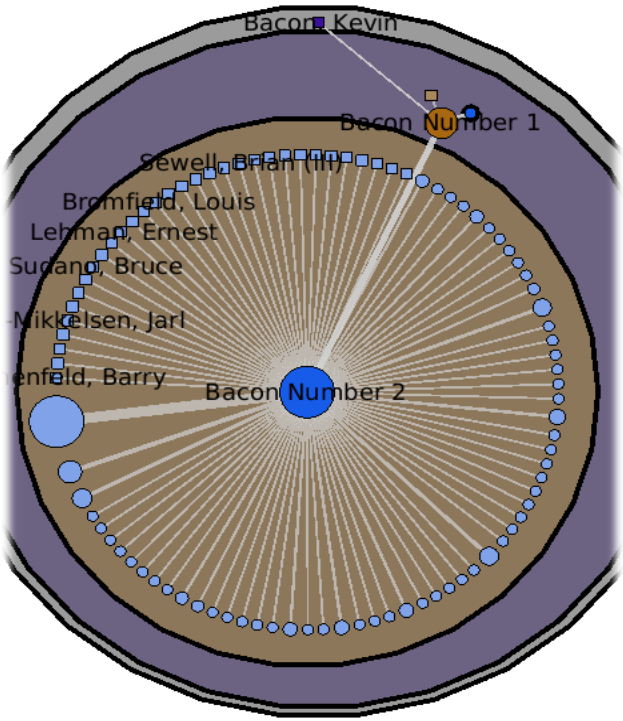
# Internet Backbone




(a) (b) (c)

- Principal Internet backbone routers
- Successive tugs reveal structure around UBC network
- **200,000** nodes and **400,000** edges

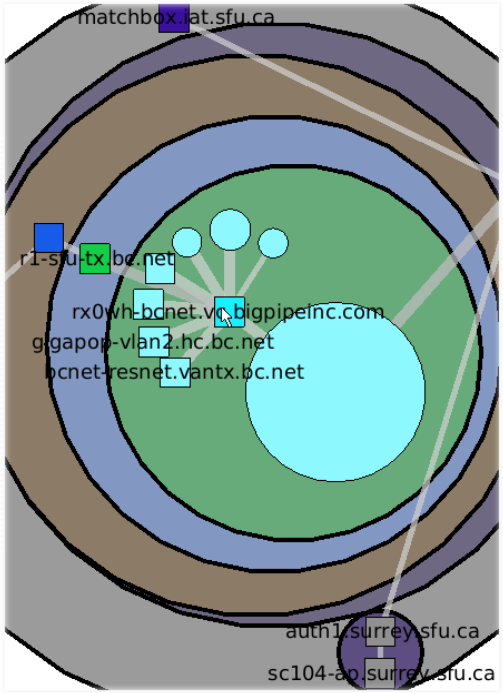
# Results: IMDb




(a)

- Bacon numbers 1 and 2 act together
- Trend not seen for Bacon number 3
- about **39,000** nodes and **2,000,000** edges

# Demo


matchbox.iat.sfu.ca
r1-sfu-tx.bc.net
rx0wh-bcnet.vc.bigpipeinc.com
gigapop-vlan2.hc.bc.net
bcnet-resnet.vantx.bc.net
auth1.surrey.sfu.ca
sc104-ap.surrey.sfu.ca

## Future Work

- Extend to weighted graphs and other notions of proximity
- Speed up technique to make tugs interactive
  - preserving hierarchy is costly
- Interactive techniques with large disconnected graphs

## Conclusion

- Presented a technique to tug out elements near a subgraph
- Executed in a path preserving way
- Provides fluid interaction with very large graphs in seconds

Software and Acknowledgements

- TugGraph available as Tulip perspective
  - released in Tulip shortly
  - `www.tulipsoftware.org`
- Thank you UBC and LaBRI for comments on the work
- Partially funded by the INRIA GRAVITÉ project
- Questions?