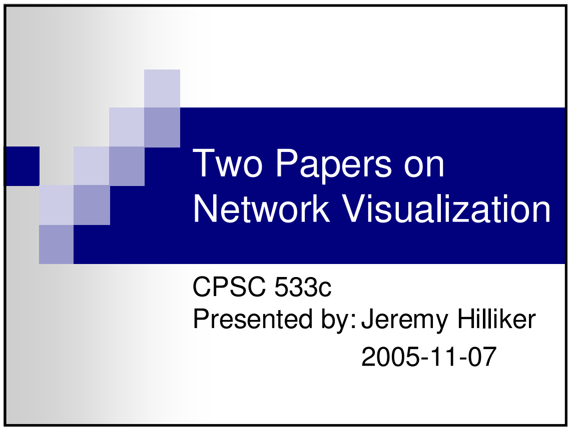
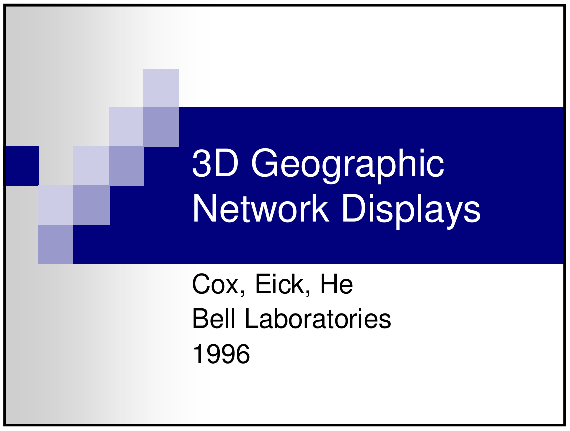


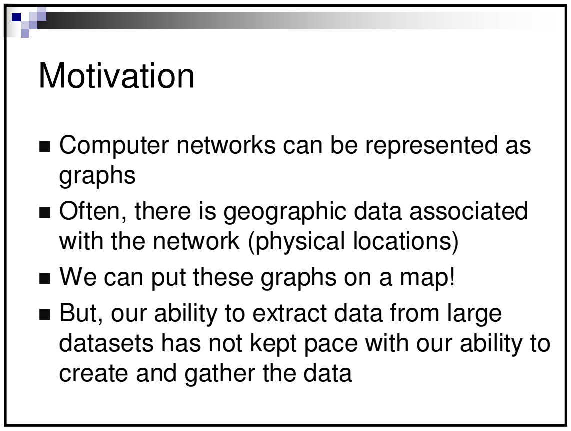
Two Papers on Network Visualization

CPSC 533c
Presented by: Jeremy Hilliker
2005-11-07



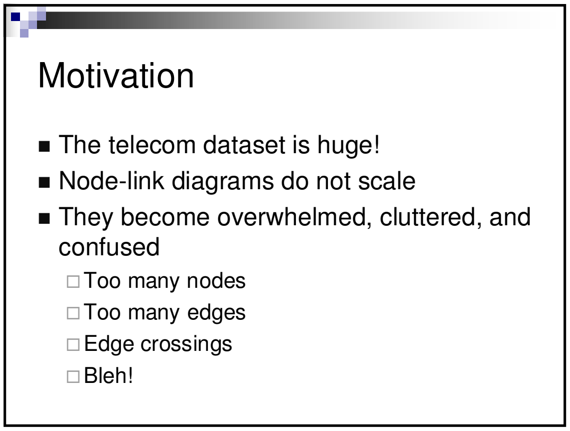
3D Geographic Network Displays

Cox, Eick, He
Bell Laboratories
1996



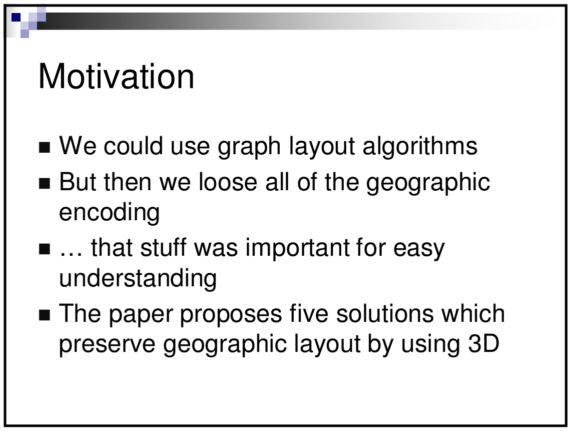
Motivation

- Computer networks can be represented as graphs
- Often, there is geographic data associated with the network (physical locations)
- We can put these graphs on a map!
- But, our ability to extract data from large datasets has not kept pace with our ability to create and gather the data



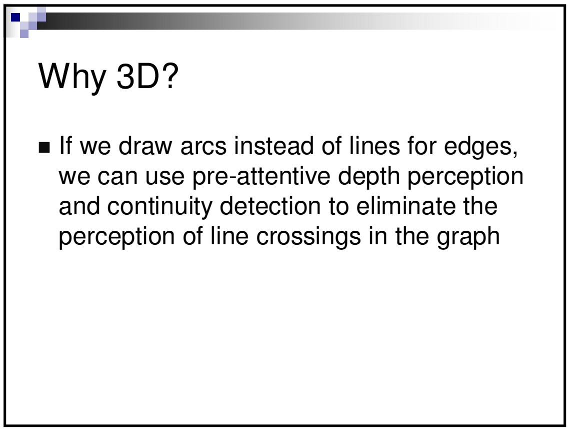
Motivation

- The telecom dataset is huge!
- Node-link diagrams do not scale
- They become overwhelmed, cluttered, and confused
  - □ Too many nodes
  - □ Too many edges
  - □ Edge crossings
  - □ Bleh!



Motivation

- We could use graph layout algorithms
- But then we loose all of the geographic encoding
- … that stuff was important for easy understanding
- The paper proposes five solutions which preserve geographic layout by using 3D

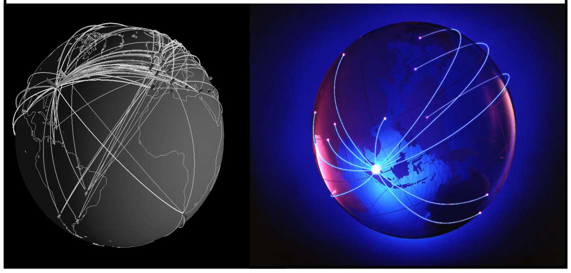


Why 3D?

- If we draw arcs instead of lines for edges, we can use pre-attentive depth perception and continuity detection to eliminate the perception of line crossings in the graph

Global Networks

- Position nodes geographically on a globe
- Draw edges as arcs between them



Global Networks

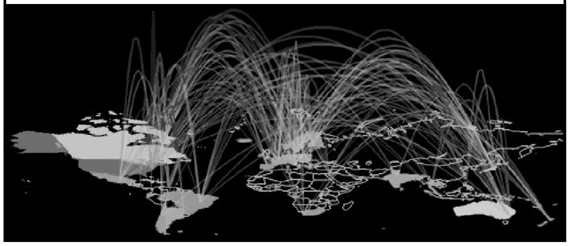
- Retains spatial information
- “Eliminates crossings” … doubtful
- Nodes represented by glyph which can use all of that glyph encoding stuff
- Arcs encoded by colour for extra info
- Illuminated by a fixed light which can indicate passage of time… not convinced
  - What happens at “night”?
- User restricted to rotations, so can’t get lost

Global Networks

- We can’t see around the globe, so we need a translucency control to see through it
  - But it’s still confusing if there is edge occlusion
- That edge clutter is still there
  - We can filter, losing context
  - Or we can select how to re-rout an edge
    - Perhaps underground?
- … it gave great geographic context, but still had all of the 2D layout problems… but worse… I don’t think it helped much

Arc Maps

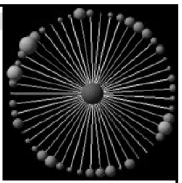
- Idea: embed a 2D map in 3D space, run edges as arcs in 3D



Arc Map

- Not restricted to a global view, can be of a small region
  - Leads to “drill-down” = details on demand
- Arcs in 3D reduce edge clutter
  - Really get continuation
  - Can rotate and zoom to get depth perception
- Arc height can give another encoding of info
- Can make arcs translucent to reduce occlusion

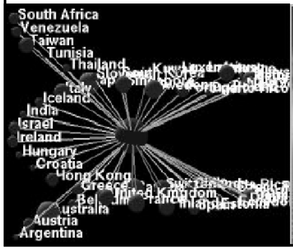
Spoke View

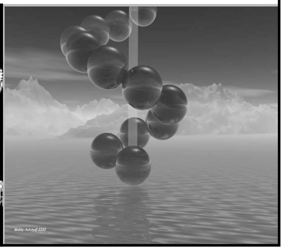


- Colour code spokes for edge data
- Colour and size code nodes
- Nodes can be placed in geographic position if we put the root at the centre of a polar projection
  - This would make it a filtered 2D global view…
- But this won’t scale
- All lines become same length wasting screen space
- Statement: we can rebuild it using 3D!

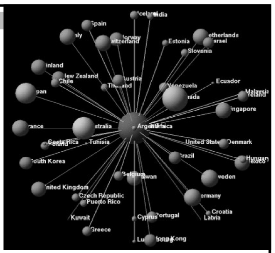
Helix View

- What if the spoke view was a top down view of a helix structure?
- We could rotate it to see everything





Pincushion View



- Arrange nodes on surface of a sphere
- Lines maintain the same spatial length (radius), but different screen length
- But nodes are evenly spread out
- Still need to rotate it to see everything

Discussion

Visualizing Large-Scale Telecom Nets and Services

Koutsofios et al.
AT&T Labs
1999

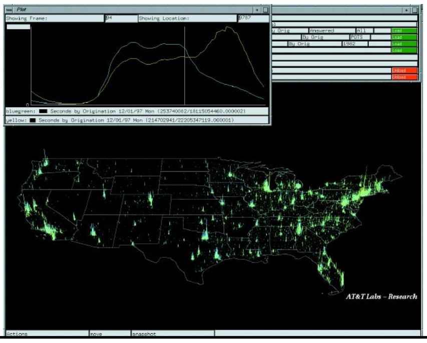
Motivation

- Still have lots of data … lots of small data
- Old databases don't handle lots of real-time, small, inter-related data well
- Understanding full scale of data is needed to manage effectively
- Goals:
  - □ Go from data to business decisions quickly
  - □ Raise level of abstraction… lines, not devices
  - □ Real-time responsiveness
- Main contribution: stream based, not query/response

Visualization stuff

- Linked 2D and 3D views (detail + overview)
- Automated context-preserving pan + zoom
- Different overlays for different data
- Semantic zoom (value per state vs. county)
- Animation over time
- Can browse and drill-down
- … seems pretty okay

Visualization Stuff



Showing Frame: 84 Showing Location: 8787

bluegreen: ■ Seconds by Origination 12/01/97 Mon (253740082/18115054460.000002)

yellow: ■ Seconds by Origination 12/01/97 Mon (214702941/22205347119.000001)

Architecture

- 3 modules
  - □ Data collector (and storage)
  - □ Aggregator (data processing and pre-proc.)
  - □ Visualization (not the important part here)
- Communication over self-describing data-independent formats
  - □ Sounds like a bad idea… (was 1 year after XML)
  - □ North-American telecom is a dinosaur
- Uses advanced systems stuff for fast communication

Data Collector

- Data is converted to the native format
- Some data has to be aggregated and joined over diverse and content-dependent sources
  - □ Mostly because telecom data is a mess
- Data that is in the right format just needs a schema attached
  - □ Doesn't sound convincing

Data Processing

- Based on pipeline model
- Concurrent processes are piped together
- Pipes can:
  - □ "Tee"
  - □ Filter
  - □ Count
  - □ Split
- Pipelines are parallelizable, modular, and simple… fast efficient, and maintainable
- Pipeline modules are compiled and dynamically linked

Data Visualization

- Interaction pattern:
  - □ View data
  - □ Focus on something interesting
  - □ Query for more details
  - □ Re-aggregate and view results
- Does this by maintaining a link between raw data, aggregate data, and visualized representation

Architecture for Performance

- Does systems stuff to make things fast
  - □ Pipelines
  - □ Random access files with version stamps
  - □ Direct I/O
  - □ Memory mapping
  - □ Dynamic linking of runtime generated code

Discussion

Sources

- Images taken from original papers or found through goolge image search
- 3D Geographic Network Displays Kenneth C. Cox, Stephen G. Eick, Taosong He. ACM SIGMOD Record Volume 25, Number 4, pp 50-54, 1996
- Visualizing Large-Scale Telecommunication Networks and Services Eleftherios Koutsofios, Stephen C. North, Russell Truscott, Daniel A. Keim. Proc IEEE Visualization 1999, pp 457-461.