

# Force-Directed Edge Bundling for Graph Visualization

Danny Holten and Jarke J. van Wijk  
Eindhoven University of Technology

Binbin Liao  
Mar. 5th, 2010

## Background

Tuesday, March 9, 2010

Tuesday, March 9, 2010

### clutter in graph visualization

- visualizing node-link graphs
- visual clutter: graph with large number of nodes or edges
- clutter reduction methods
  - appearance based
  - spatial distortion
  - temporal

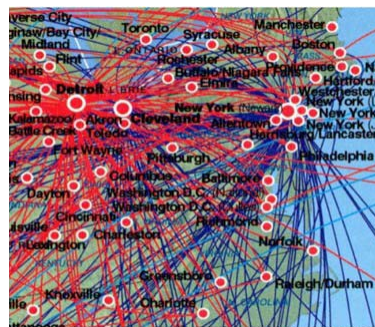


Figure 1. Airline routes from NorthWest Airlines, "World Traveler", November, 2001.

### clutter reduction method taxonomy

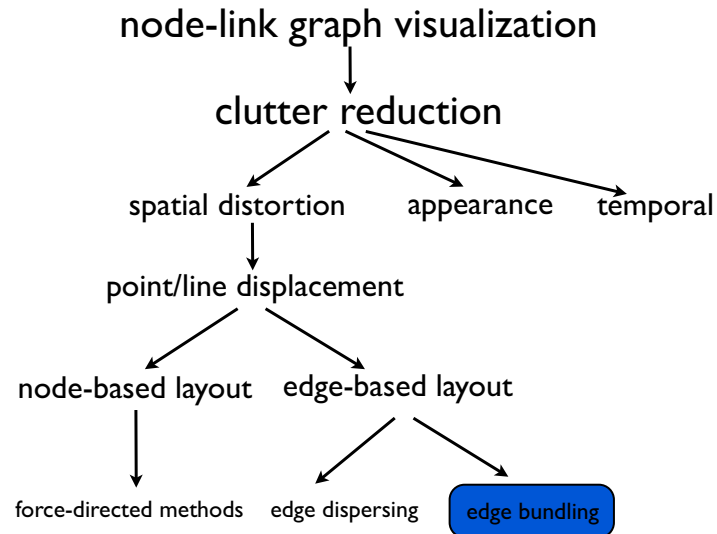
criteria	methods	
appearance	sampling	
	filtering	
	clustering	
	opacity, point size, resolution	
spatial distortion	point/line displacement (graph layout)	node-based
		edge-based
	topological distortion	
	space filling	
	pixel plotting	
temporal	dimensional reordering	
	animation	

extracted from (Ellis, TVCG07)

Tuesday, March 9, 2010

Tuesday, March 9, 2010

## overview



Tuesday, March 9, 2010

## Edge-based clutter reduction

- edge dispersing: *local, interactive*
  - EdgeLens (Wong, Infovis2003)
  - Edge plucking (Wong, Infovis2005)
- edge bundling: *global, static*
  - flow map (Phan, Infovis2005)
  - hierarchical edge bundling (HEB) (Holten, TVCG06)
  - geometry-based edge bundling (GBEB) (Cui, TVCG08)
  - force-directed edge bundling (FDEB)



Tuesday, March 9, 2010

## Edge dispersing - EdgeLens

- displace edges in a local area with a high degree of edge overlap to reveal hidden information without changing node position

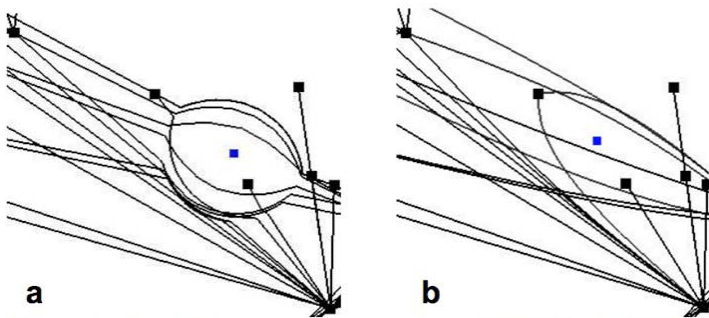


Figure 5. Two EdgeLens approaches. a) Bubble; b) Spline.

Tuesday, March 9, 2010

## Edge dispersing - Edge plucking

- temporarily *pulls* edges apart to clarify underlying node-edge relationships, instead of *push* in graphlens

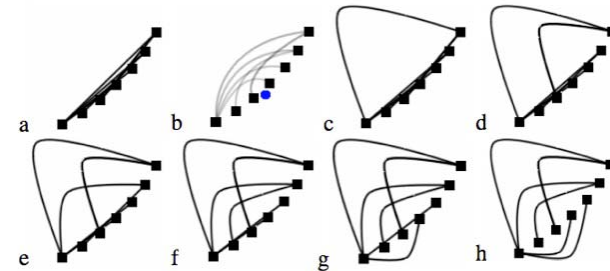
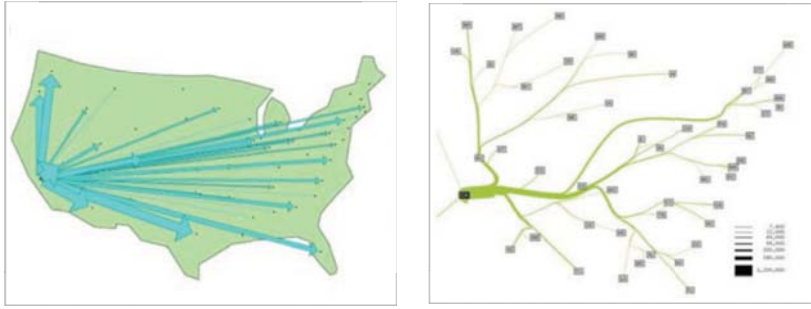


Figure 6. Exploring a cluster of nodes in a graph: a) the original graph; b) EdgeLens is applied; c) through h) Edge Plucking is used separate the edges within the cluster of nodes

Tuesday, March 9, 2010

## Edge bundling - flow map

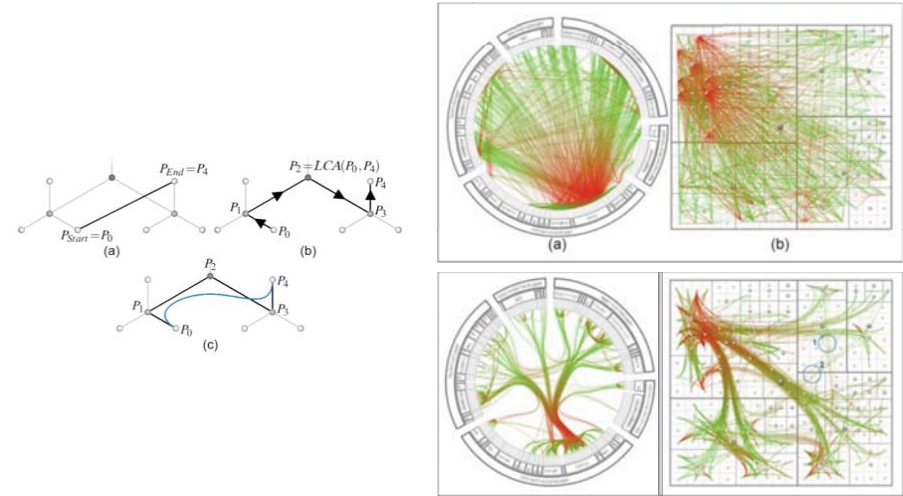
- generate a flow map using hierarchical clustering given the node positions and flow data



Tuesday, March 9, 2010

## Edge bundling - hierarchical edge bundling

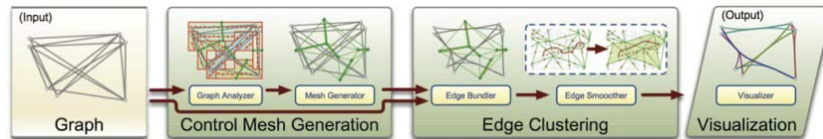
- draw edges along their paths within the hierarchical tree, rely on existing tree layout algorithms



Tuesday, March 9, 2010

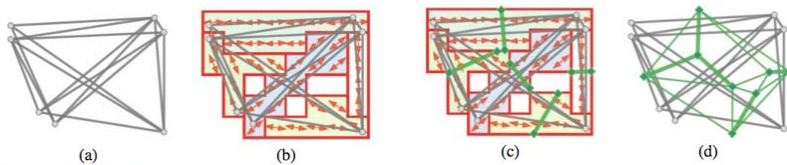
## Edge bundling - GBEB

- framework



- mesh generation

- primary local edge direction (cluster)
- and make mesh edges pierce perpendicularly through merged local edge clusters



Tuesday, March 9, 2010

## Edge bundling - GBEB pipeline

- edge bundling

- control points: center of the intersection of graph edges and mesh edge
- each edge assigned to a set of control points
- force edge curves passing through local control points

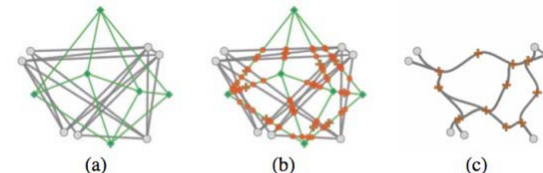
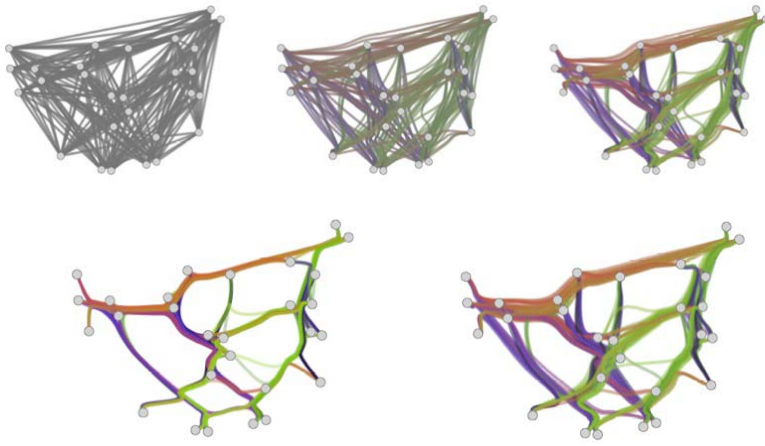


Fig. 6. Edge clustering by control points: (a) a graph with a control mesh; (b) the intersections and the control points; (c) the merged graph.

Tuesday, March 9, 2010

## Edge bundling - GBEB



Tuesday, March 9, 2010

## Background review

- node-link visualization
- clutter reduction
- Edge-based methods for clutter reduction
  - edge dispersing: *local, interactive*
    - EdgeLens
    - Edge plucking
  - edge bundling: *global, static*
    - flow map
    - hierarchical edge bundling (HEB)
    - geometry-based edge bundling (GBEB)
    - force-directed edge bundling (FDEB)

Tuesday, March 9, 2010

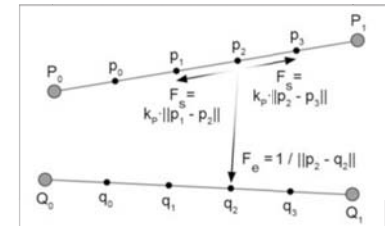
## Force-directed edge bundling

### force-directed edge bundling

- Force-directed method (*Fruchterman, 1991; Kamada, 1989*)
- FDEB
  - input: straight-line node-link graph
  - edge subdivision
  - spring attraction
  - electrostatic force

$$\mathbf{F}_{p_i} = k_p \cdot (\|p_{i-1} - p_i\| + \|p_i - p_{i+1}\|) + \sum_{Q \in E} \frac{1}{\|p_i - q_i\|},$$

$k_p = K/|P|$ : spring constant for each segment  
 E: set of all all interacting edges

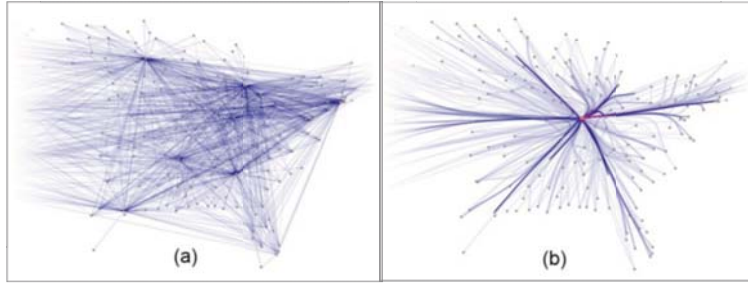


Tuesday, March 9, 2010

Tuesday, March 9, 2010

## Edge compatibility

- immediate result



- edge compatibility
  - angle compatibility
  - scale compatibility
  - position compatibility
  - visibility compatibility

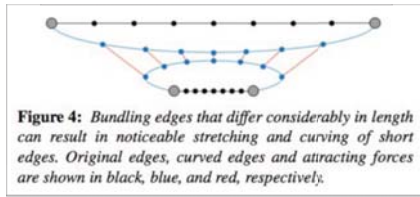


Figure 4: Bundling edges that differ considerably in length can result in noticeable stretching and curving of short edges. Original edges, curved edges and attracting forces are shown in black, blue, and red, respectively.

Tuesday, March 9, 2010

## Edge compatibility

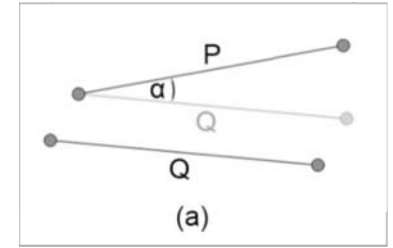
- angle compatibility

$$C_a(P, Q) = |\cos(\alpha)|,$$

with

$$\alpha : \arccos\left(\frac{P \cdot Q}{|P||Q|}\right).$$

- $C_a(P, Q) : [0, 1]$



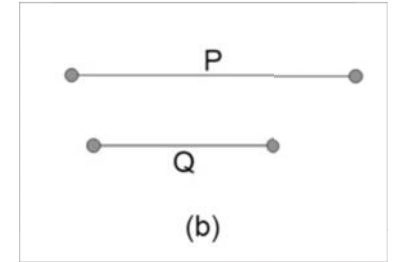
- scale compatibility

$$C_s(P, Q) = \frac{2}{l_{avg} / \min(|P|, |Q|) + \max(|P|, |Q|) / l_{avg}},$$

with

$$l_{avg} : \frac{|P| + |Q|}{2}.$$

- $C_s(P, Q) = 1$  if  $|P| = |Q|$ ,
- $C_s(P, Q) = 0$  if ratio  $\rightarrow \inf$



Tuesday, March 9, 2010

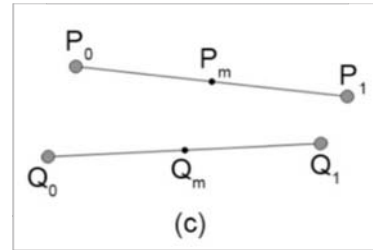
## Edge compatibility

- position compatibility

$$C_p(P, Q) = l_{avg} / (l_{avg} + \|P_m - Q_m\|),$$

with  
 $P_m$  and  $Q_m$  : midpoints of edges  $P$  and  $Q$ .

- $C_p(P, Q) = 1$  if  $P_m = Q_m$ ,
- $C_p(P, Q) = 0$  if  $\|P_m - Q_m\| \rightarrow \inf$



- visibility compatibility

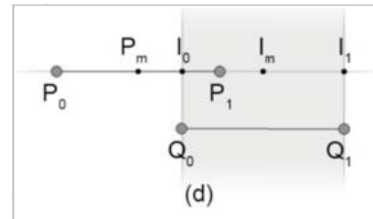
$$C_v(P, Q) = \min(V(P, Q), V(Q, P)),$$

with

$$V(P, Q) : \max\left(1 - \frac{2\|P_m - I_m\|}{\|I_0 - I_1\|}, 0\right),$$

$I_m$  : midpoint of  $I_0$  and  $I_1$ .

- $C_v(P, Q) = 1$  if  $P_m = I_m$ ,
- $C_v(P, Q) = 0$  if fall outside



Tuesday, March 9, 2010

## Revised model

- edge compatibility

$$C_e(P, Q) = C_a(P, Q) \cdot C_s(P, Q) \cdot C_p(P, Q) \cdot C_v(P, Q).$$

- revised model

$$\mathbf{F}_{p_i} = k_P \cdot (\|p_{i-1} - p_i\| + \|p_i - p_{i+1}\|) + \sum_{Q \in E} \frac{C_e(P, Q)}{\|p_i - q_i\|}.$$

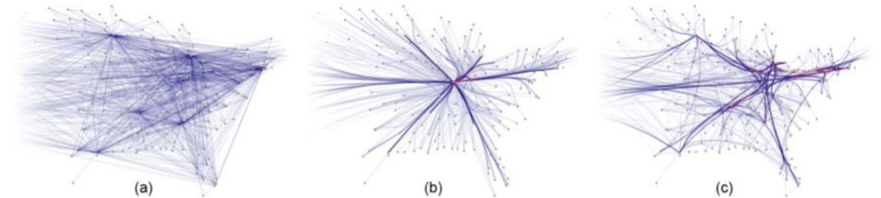


Figure 2: Part of (a) a straight-line graph that is bundled (b) without and (c) with edge compatibility measures. These measures reduce the amount of bundling between incompatible edges while retaining it in parts of the graph where this is desirable.

Tuesday, March 9, 2010

Tuesday, March 9, 2010

## Procedure

1. Parameters: cycle -  $C$ , iteration -  $I$ , segmentation -  $P$ , stepsize -  $S$
2. Init:  $S = S_0, P = P_0, I = I_0$
3. for cycle  $c < C$
4.   for iteration  $i < I$
5.     for each subdivision point  $p$
6.        $p += d \cdot S$
7.      $S /= 2; I = 2 \cdot I / 3; P += 2;$

$$O(I_{\text{sum}} N^2 P)$$

- $C = 5, S_0 = 0.04, P_0 = 1, I_0 = 50$

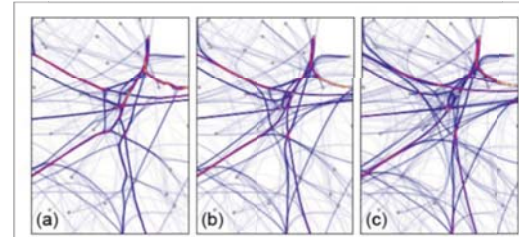
cycle	0	1	2	3	4	5
$P$	1	2	4	8	16	32
$S$	.04	.02	.01	.005	.0025	.00125
$I$	50	33	22	15	9	7

Tuesday, March 9, 2010

## smoothing

- filter out edges with small edge compatibility value
  - significantly reduce computation
  - large amount of smoothing gives less jagged edge appearance

$$C_e(P, Q) = \begin{cases} C_a(P, Q) \cdot C_s(P, Q) \cdot C_p(P, Q) \cdot C_v(P, Q) & \text{if } C_e(P, Q) > \eta \\ 0 & \text{otherwise} \end{cases}$$



**Figure 5:** Part of a bundled graph at (a) 0%, (b) 25%, and (c) 50% smoothing. A small amount of smoothing makes edges less jagged resulting in bundles that are easy to follow.

Tuesday, March 9, 2010

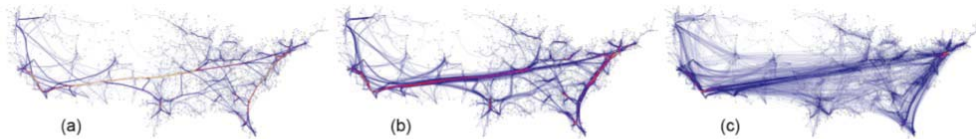
## Bundling straightening

- Continuously change the bundling strength without changing the spring constant  $K$

$$p'_i = (1 - s)p_i + s(P_0 + \frac{i+1}{N+1}(P_1 - P_0)),$$

with

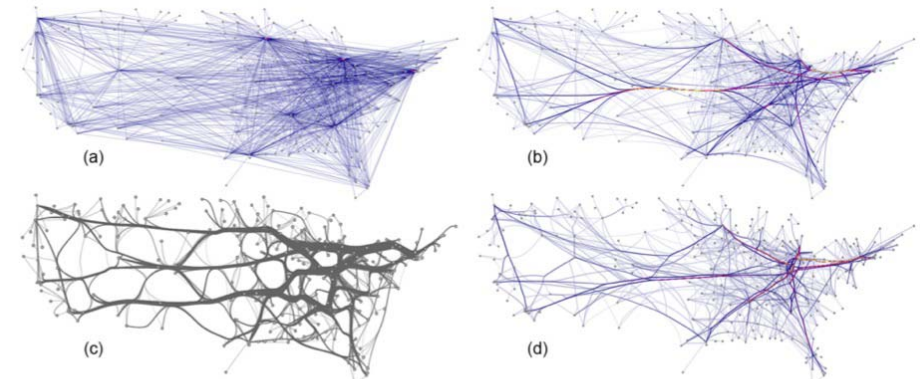
- $N$  : number of subdivision points,  
 $s$  : amount of straightening,  $s \in [0, 1]$ .



**Figure 9:** A low amount of straightening provides an indication of the number of edges comprising a bundle by widening the bundle. (a)  $s = 0$ , (b)  $s = 10$ , and (c)  $s = 40$ . If  $s$  is 0, color more clearly indicates the number of edges comprising a bundle.

Tuesday, March 9, 2010

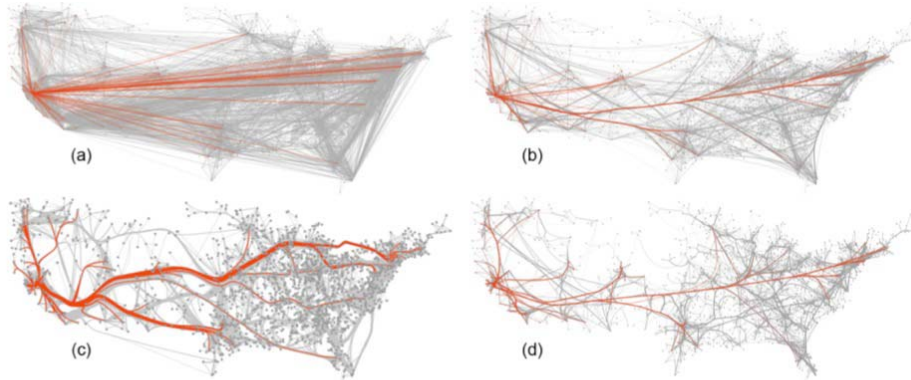
## Comparison to GBEB



**Figure 7:** US airlines graph (235 nodes, 2101 edges) (a) not bundled and bundled using (b) FDEB with inverse-linear model, (c) GBEB, and (d) FDEB with inverse-quadratic model.

Tuesday, March 9, 2010

## Comparison to GBEB



**Figure 8:** US migration graph (1715 nodes, 9780 edges) (a) not bundled and bundled using (b) FDEB with inverse-linear model, (c) GBEB, and (d) FDEB with inverse-quadratic model. The same migration flow is highlighted in each graph.

Tuesday, March 9, 2010

## Comparison to GBEB cont'd

- simplicity
- performance

	airline graph (235x2101)	migration graph (1715x9780)
FDEB	2.5s	12.9s
GBEB	18.8s	79.6s

Tuesday, March 9, 2010

## Conclusions

- intuitive
- simple to implement
- self-organizing approach to general graphs
- quality
  - moderate webbing
  - low curvature-variation
  - show high-level edge patterns

Tuesday, March 9, 2010

## Future work

- computational complexity

$$O(I N^2 P)$$

- solution
  - Hierarchical force calculation (Barnes, *Nature* 1986)
  - stress majorization (Gansner, *Graph Drawing* 2004)
  - GPU-based parallelization

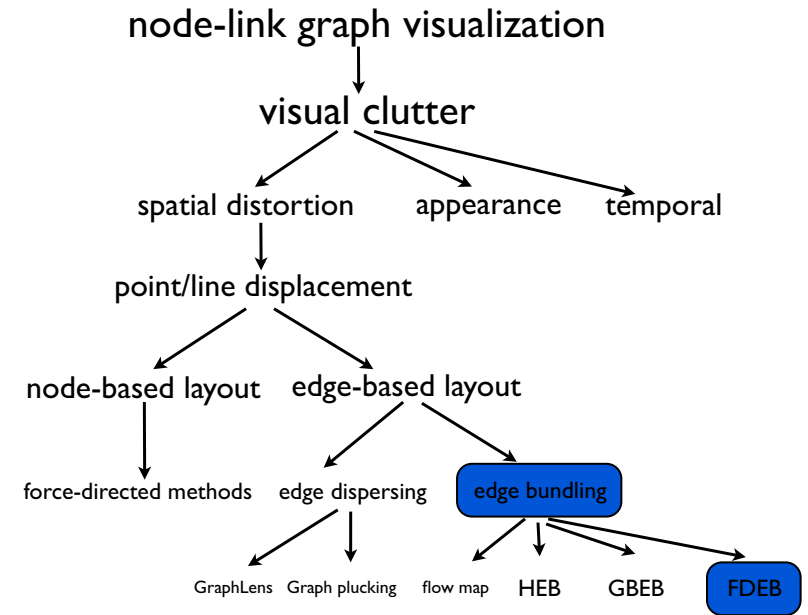
Tuesday, March 9, 2010

## Future work

- Quantify the quality of edge bundling visualization
- webbing/bundling: measure the amount of per-pixel overdraw and empty space for FDEB, GBEB and straight-line graphs
- smoothness: edge curvature variation and length

Tuesday, March 9, 2010

## review



Tuesday, March 9, 2010

Thanks!

Tuesday, March 9, 2010