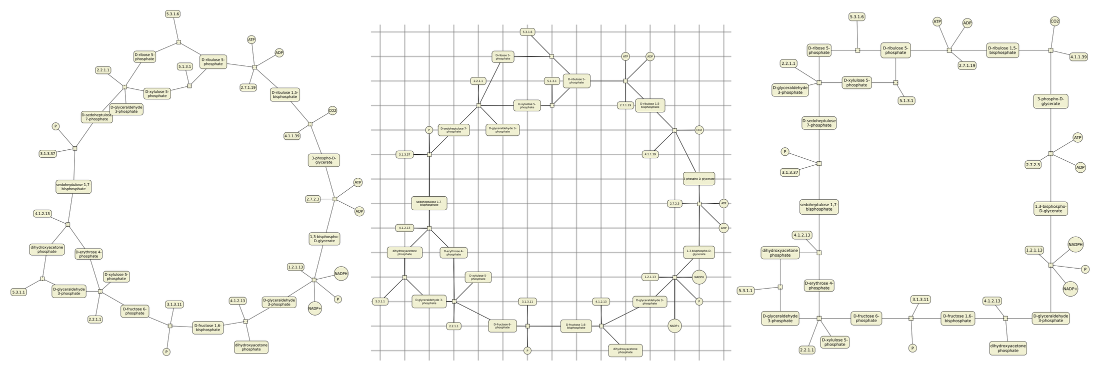


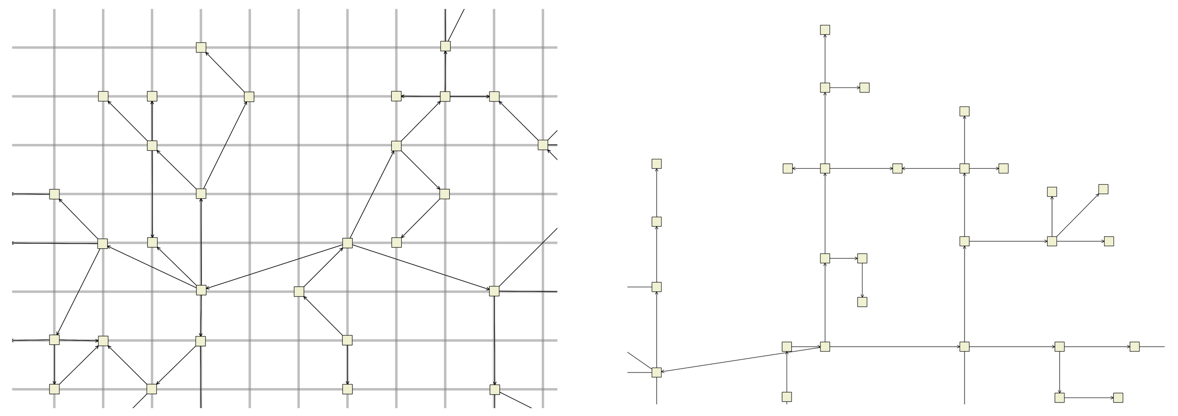
Adaptive Grid-like Layout

Steve Kieffer, Tim Dwyer, Kim Marriott, Michael Wybrow

sakie1@student.monash.edu,
{Tim.Dwyer, Kim.Marriott, Michael.Wybrow}@monash.edu

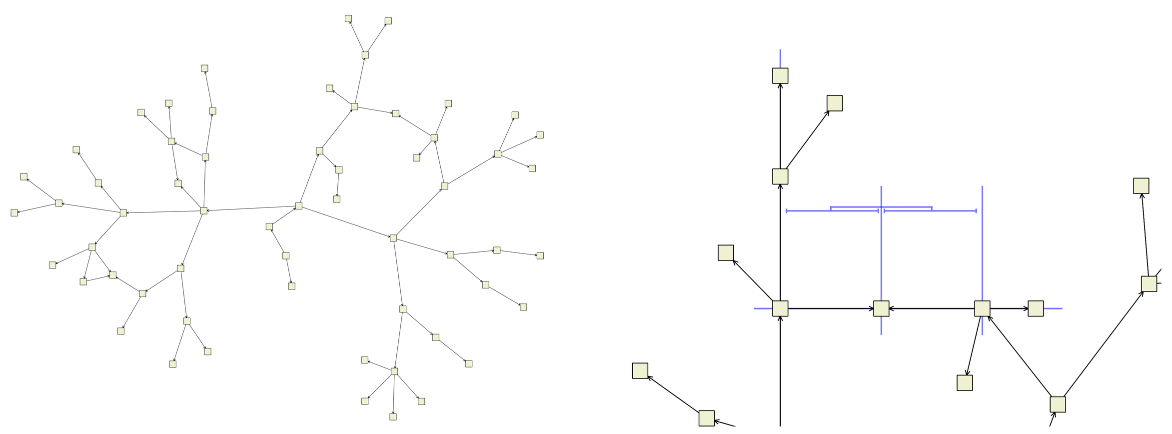
MArVL Monash Adaptive Visualisation Lab — Monash University — NICTA

The Goal: Grid-like Layout



Nodes are positioned on a fairly coarse grid.
AND/OR
Most or all edges are simple horizontal or vertical lines.

Basics



Force-directed layout untangles and distributes nodes by minimising a *stress* function. We use the *P-stress* function from [1]:

$$\sum_{u<v\in V} w_{uv}\left(\left(d_{uv}-\|(x_u,y_u)-(x_v,y_v)\|\right)^+\right)^2 + \sum_{(u,v)\in E} wp\left(\left(\|(x_u,y_u)-(x_v,y_v)\|-d_L\right)^+\right)^2$$

Solving subject to **alignment and distribution constraints** makes it a constrained optimisation problem.

The Problem

Stress minimization alone cannot achieve grid-like layout.

Adding alignment constraints by hand is labour-intensive.

References

[1] T. Dwyer, K. Marriott, M. Wybrow Topology preserving constrained graph layout In *GraphDrawing'09*, 2009.

[2] J. Stott *et al* Automatic metro map layout using multicriteria optimization In *TVCG'11*, 2011.

Our Contributions

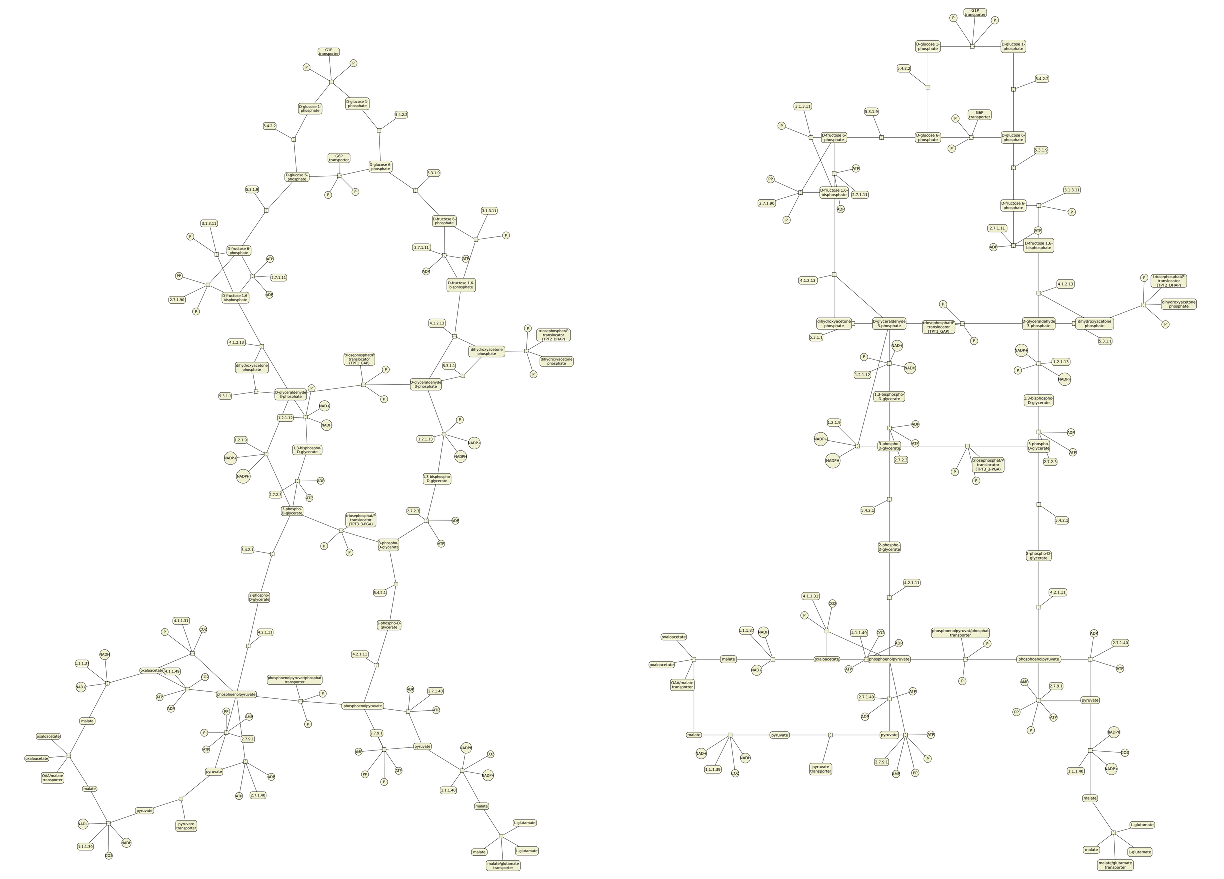
We designed three techniques: *Adaptive Constrained Alignment (ACA)* which automatically applies alignment constraints, and *Grid-Snap* and *Node-Snap* which add special terms to the stress function.

Adaptive Constrained Alignment (ACA)

ACA is a greedy algorithm that repeatedly chooses an edge and aligns it horizontally or vertically, halting when no good choices remain. Choices are weighted by a cost function designed to improve grid-like layout aesthetics.

Basic cost $K_{dS}(e,D)$ of aligning edge e in direction D estimates change in stress function caused by this alignment. Minimizing K_{dS} means respecting the overall shape given by the FD-layout.

An additional fixed cost of 1000 for any alignment that would make a degree-2 node into a "bend-point" promotes straight lines and perfect rectangles.



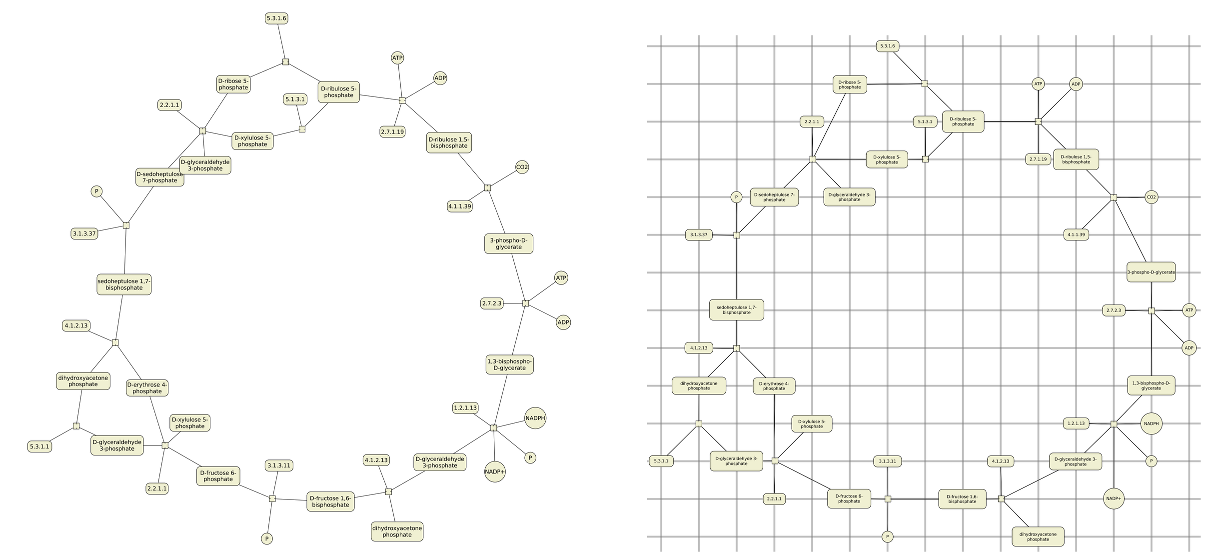
FD layout of a biological network (left) and the result of ACA (right) with costs tailored for conventions of SBGN (Systems Biology Graphical Notation).

Grid-Snap (GS)

To make the lines of a virtual grid exert an attractive force on nodes, we augmented the stress function with the term

$$\sum_{u\in V} q_\sigma(x_u-a_u)+q_\sigma(y_u-b_u)$$

where σ is the "snap distance", (a_u,b_u) is the closest grid point to the centre point (x_u,y_u) of node u, and $q_\sigma(z)=z^2/\sigma^2$ for $|z|<\sigma$ and 0 otherwise.



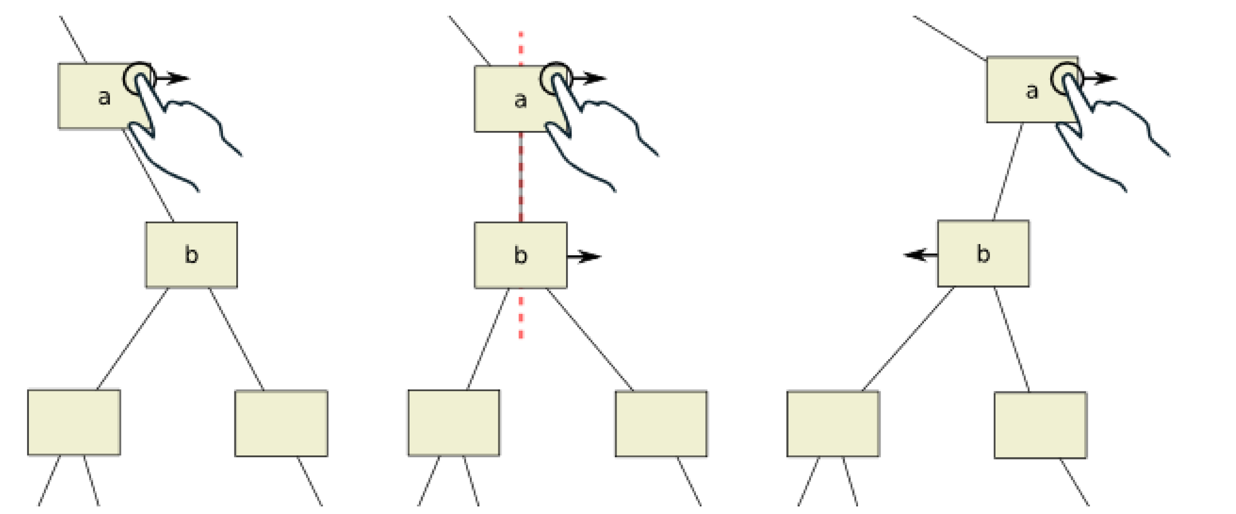
Grid-Snap (right) retains the roughly circular shape of the Force-Directed layout (left), but positions nodes regularly and neatly on points of a grid.

Node-Snap (NS)

We made nodes exert an attractive force on each other with the stress term

$$\sum_{(u,v)\in E} q_{\alpha(u,v)}(x_u-x_v)+q_{\beta(u,v)}(y_u-y_v)$$

where $\alpha(u,v)$ and $\beta(u,v)$ are the average width resp. height of nodes u,v, and $q_\sigma(z)$ is the same as for Grid-Snap (see above). This makes nodes snap into alignment as the user drags them interactively.



The user drags node a steadily to the right. When a and b are nearly aligned, the node-snap stress makes them pop into alignment. Now b follows a until attached edges pull it left, overcoming its attraction to a.

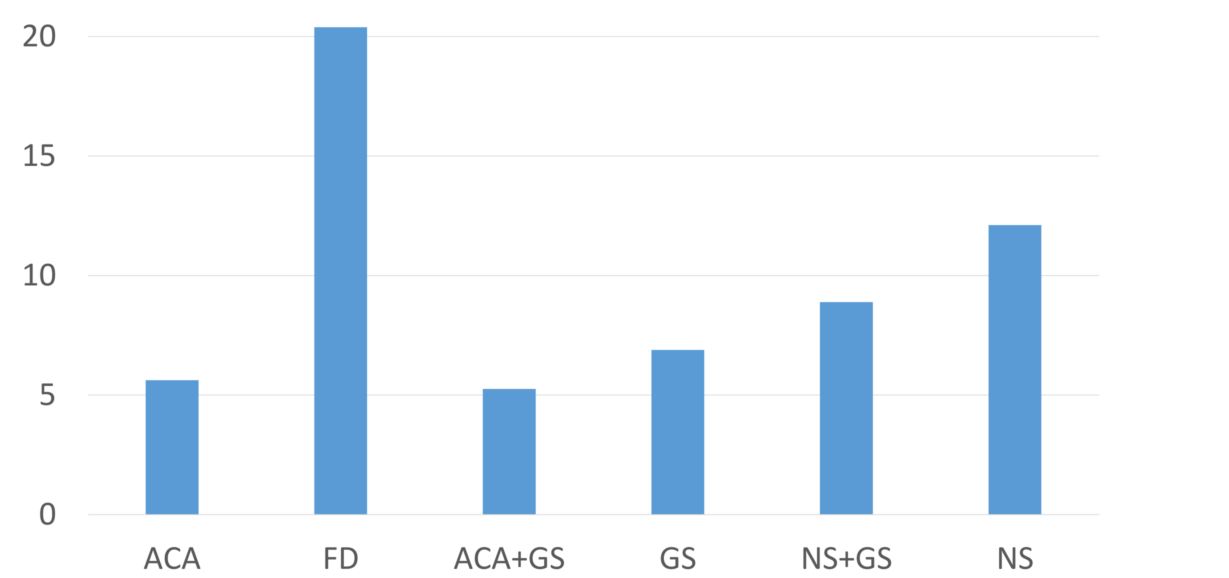
Aesthetic Criteria

Among natural aesthetic criteria for grid-like layout are:

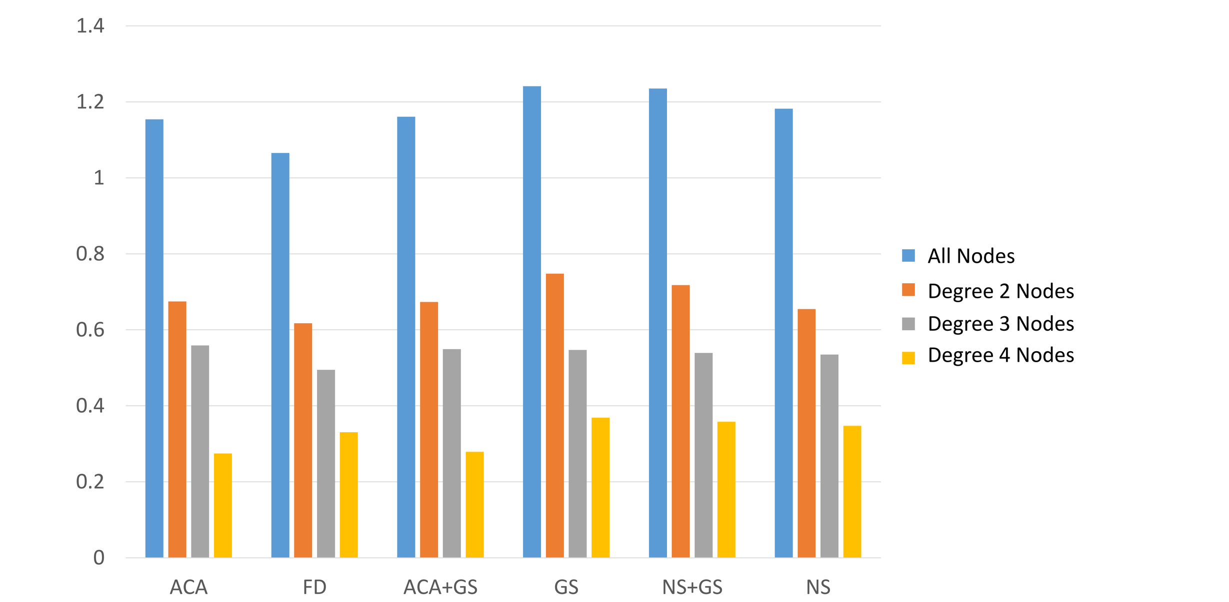
Angular resolution: edges incident on the same node should have a uniform anuglar separation. We use the metric of [2].

Edge obliqueness: we assign zero cost to horizontal and vertical edges, a small positive cost to edges at 45° angle, and a larger cost for angles closer to but different from 0 or 90°.

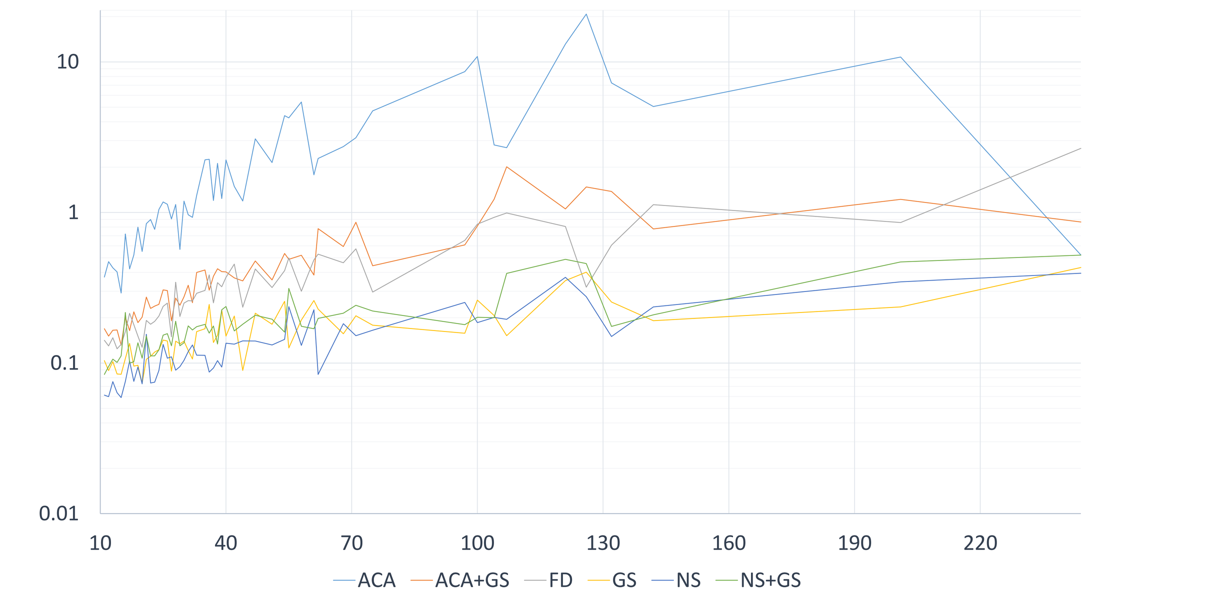
Evaluation



Edge obliqueness: ACA is best.



Angular resolution: ACA does almost as well as FD on degree-2 nodes, and better than FD for degree-4.



Running time in seconds against number of nodes for 252 graphs from the AT&T Graphs corpus. Times given do not include the other layout stages.