

Abstract

Event-based dynamic graph drawing can be used to visualise simulated disease spread in a population. Two-dimensional force-directed static graph-drawing algorithms can be extended to 2D + T to draw the temporal network in the space-time cube. In such a way, dynamic graph drawing lets us visualise, for example, a contact tracing simulation [1] of disease spread on such a temporal network. Building on the framework of the DynNoSlice algorithm [2, 3], we propose an extended version of the animation generating software that encodes location of infection spatially. The improvements include a number of functionalities allowing the user to highlight or filter out particular aspects of the disease transmission encoded with a dynamic graph. We test our improved algorithm on Covid-19 contact tracing simulations.

Cluster-specific forces

A cluster is an element type with:

- a *pole node*, positioned in the centre cluster's,
- a *circumference*
- *member nodes* (sharing a commonality we want to make prominent in the drawing)

An edge is inserted between the pole and a member nodes. Members are kept within the cluster's circumference.

A number of *forces* (following the definition in [2] and utilised in DynNoSlice) graphically differentiate the *members* from the rest (other clusters and un-assigned members).

- Node-to-node – attraction between the pole node and the cluster's members nodes.

- Circumference repulsion adapted version of the PrEd algorithm force [4] – the circumference is circumscribed over a polygon (32-degree graph face, nodes placed equidistantly). A member is repelled to the pole from the circumference from each of the face's **abstract** edges:

$$F(v, (a, b)) = \frac{(\gamma - \|p_v - v_e\|)^2}{\|p_v - v_e\|} (p_v - v_e)$$

$$v \in (a, b), a \neq v, b \neq v, \|p_v - v_e\| < \gamma$$

p_x is the position of node x , y is the desired distance from a node to an edge, and v_e is the position of node v on a line which is defined by the edge (a, b) .

- Pole-to-node repulsion – applied to every non-member node (all nodes that are unassigned to a cluster):

$$F_u^r(v, u) \leftarrow \left(\frac{\gamma}{\|p_v - p_u\|} \right)^8 \hat{v}u$$

p_u and p_v are the positions of the member node and the pole node respectively, and $\hat{v}u$ is the unit vector pointing in the direction from p_v to p_u .

Figure 1. Cluster with pole p and m_i member nodes

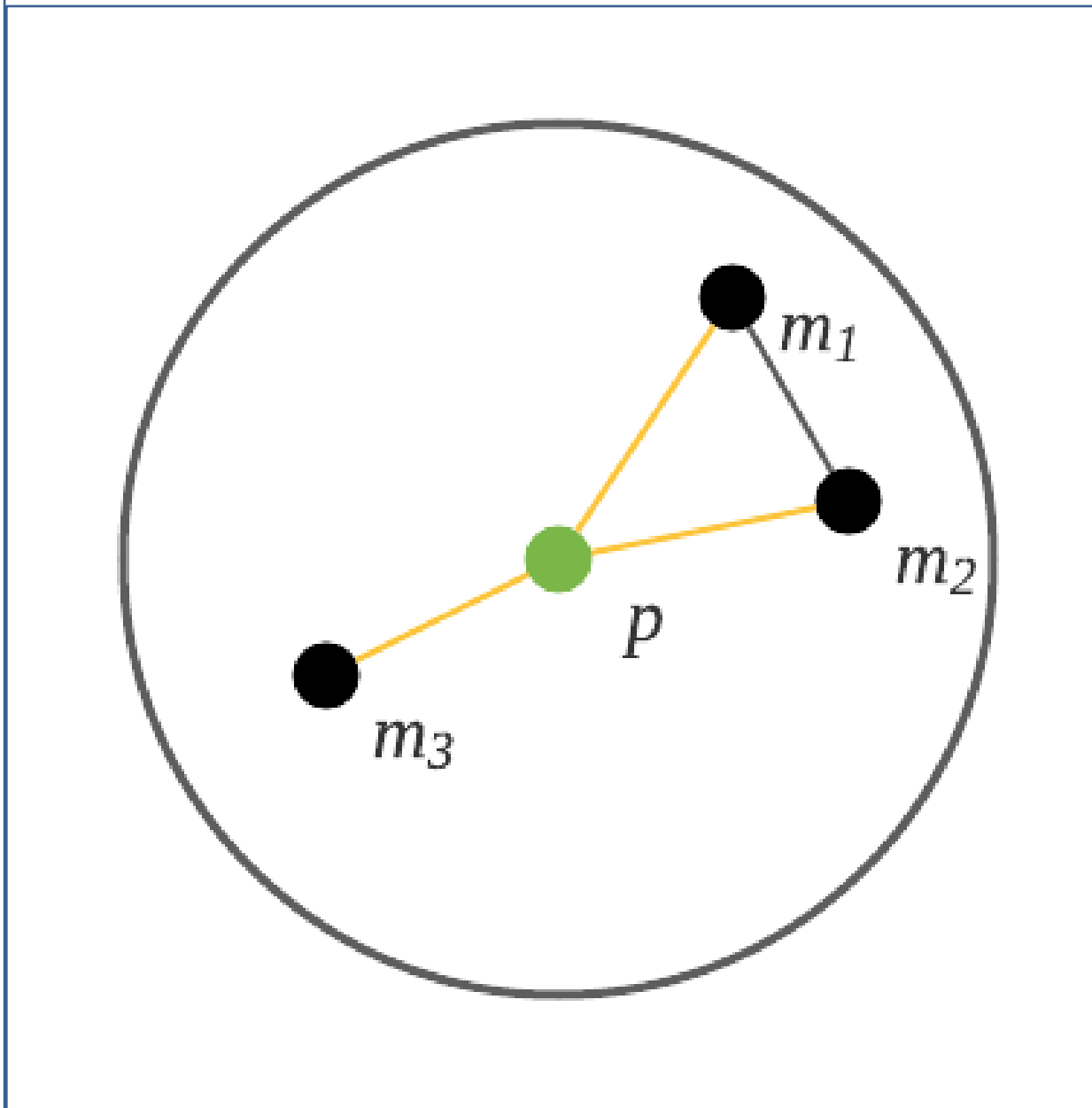


Figure 2. Pole-to-node repulsion

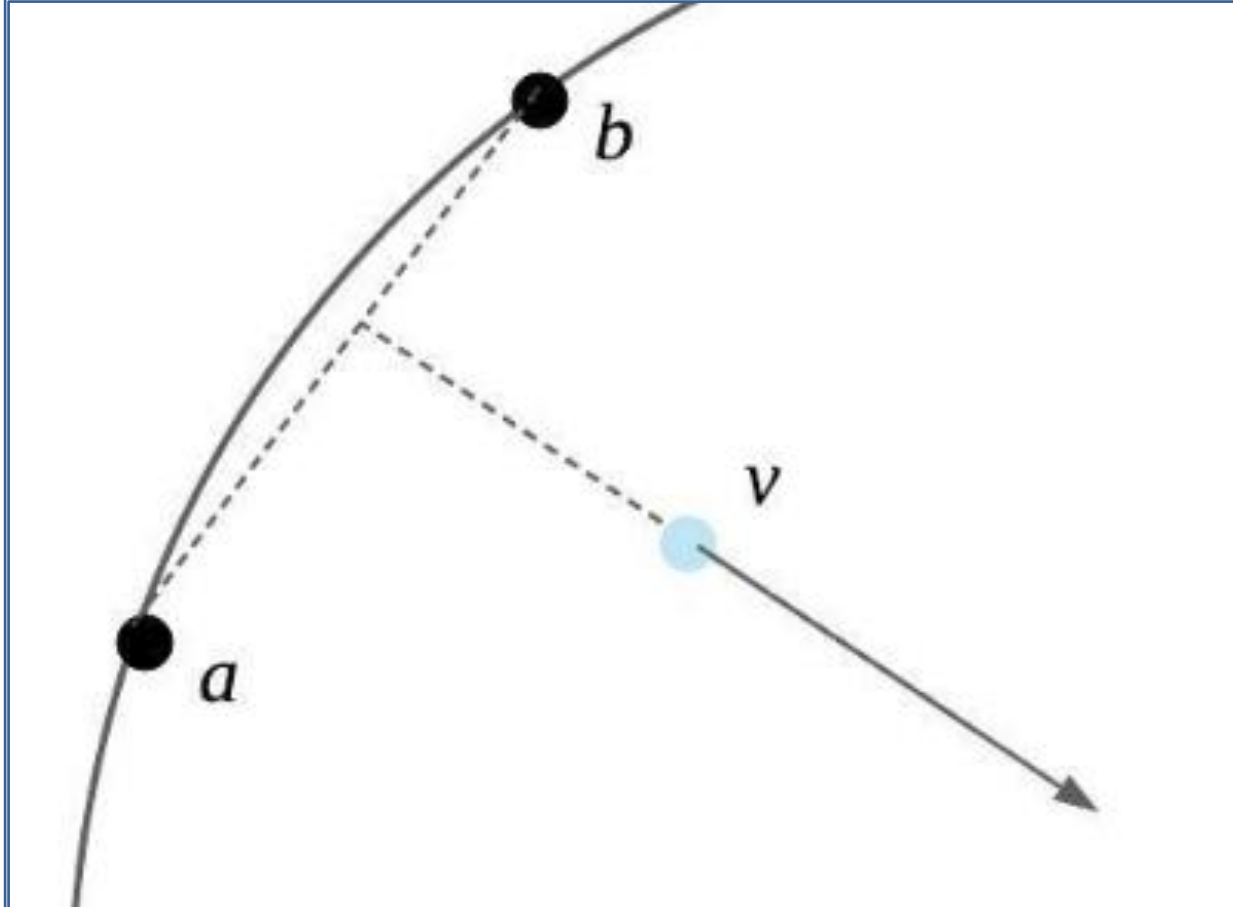
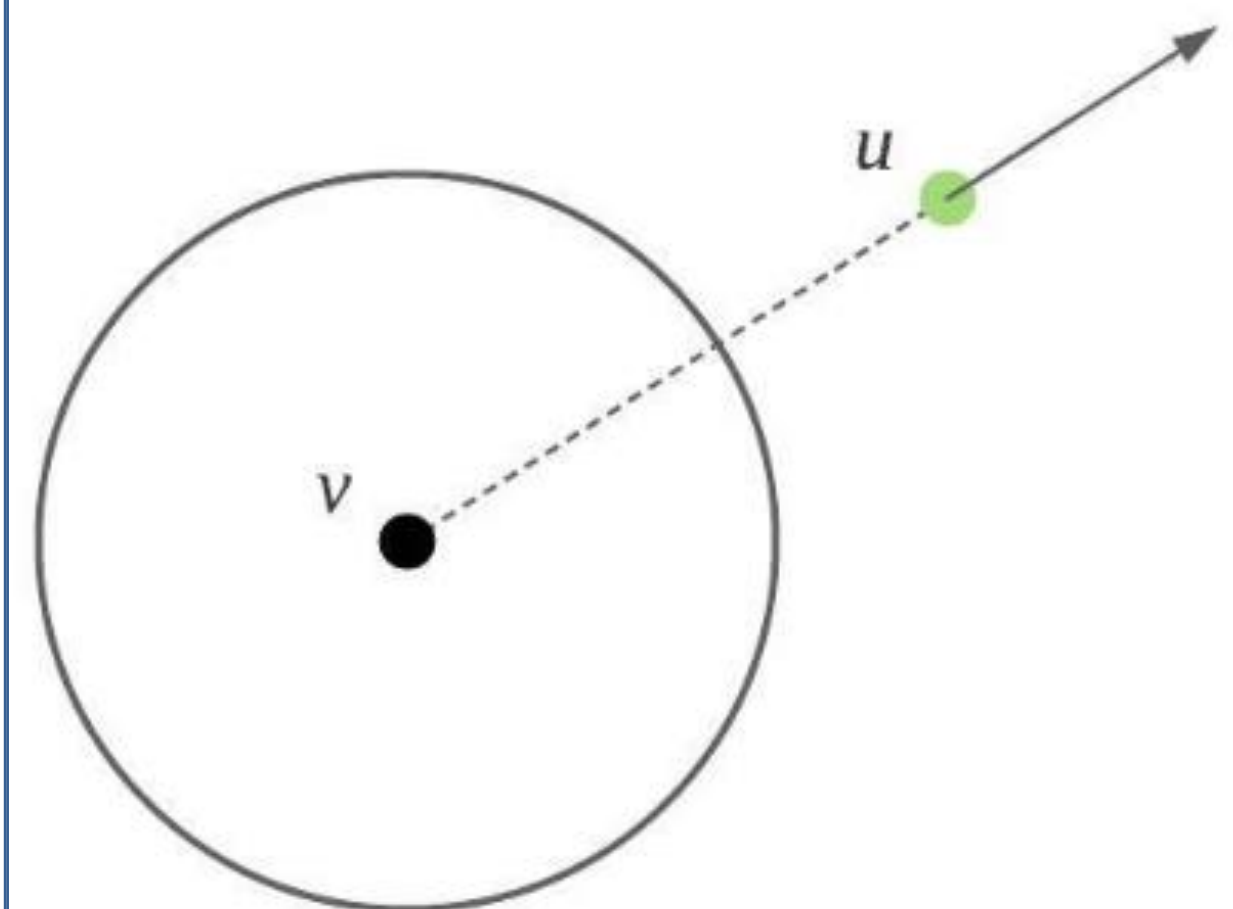


Figure 3. Circumference repulsion



PrEd adaptation

A force described by this node-to-pole repulsion formula, rather than the one included in the PrEd algorithm [4], provides a more stable layout with graphs of composed of above ca. 30 nodes.

New functionalities

Figure 4. Space-time cube view

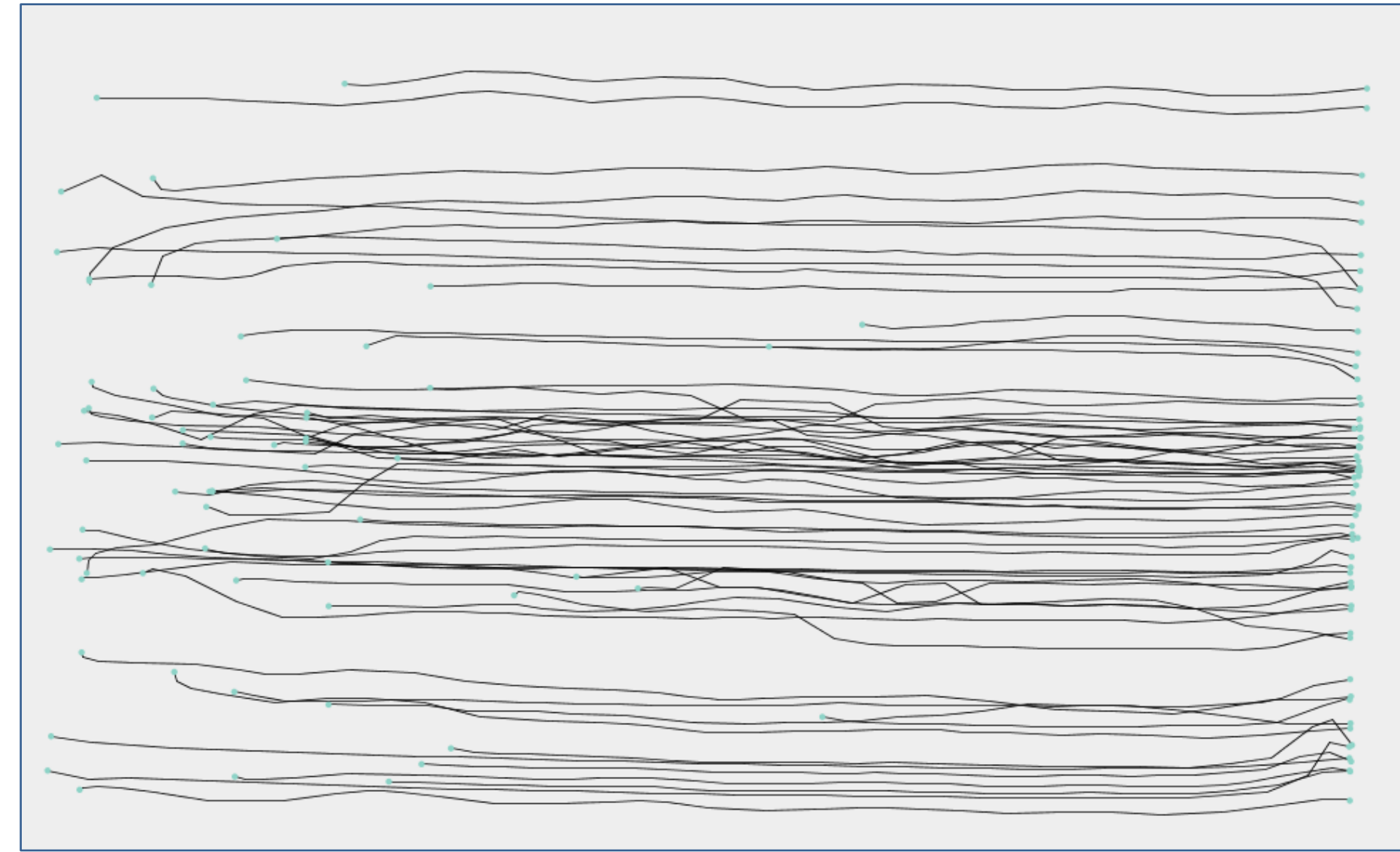


Figure 5. The *Continuous with Multiple Locations Attraction* functionality

- Cluster pole nodes pinned
- Pick multiple locations, each represented as a cluster (pole labelled as a location name)
- Nodes infected in *location* will be encouraged to stay within *cluster*, circumference.

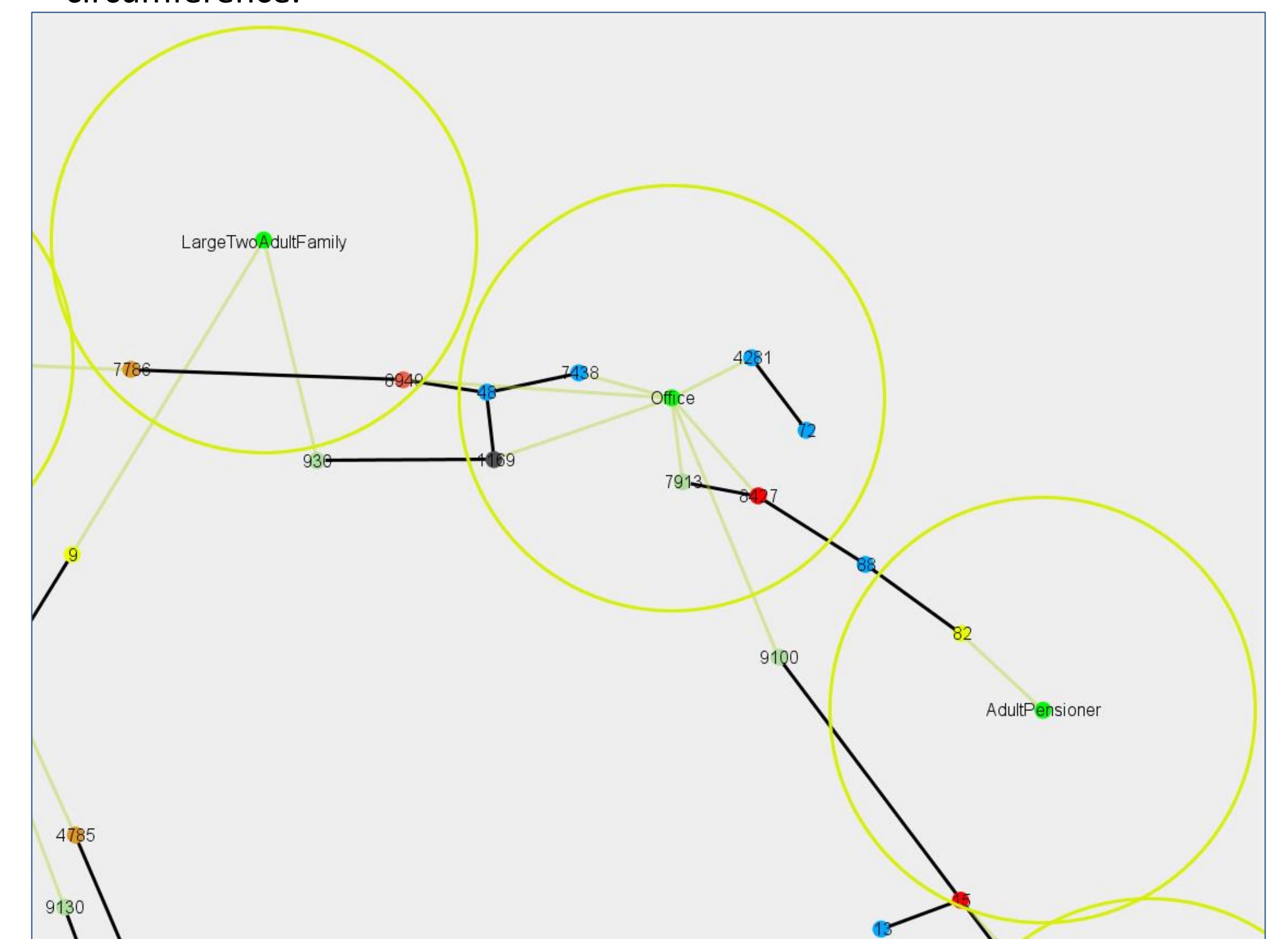


Figure 6. Cluster pole nodes floating

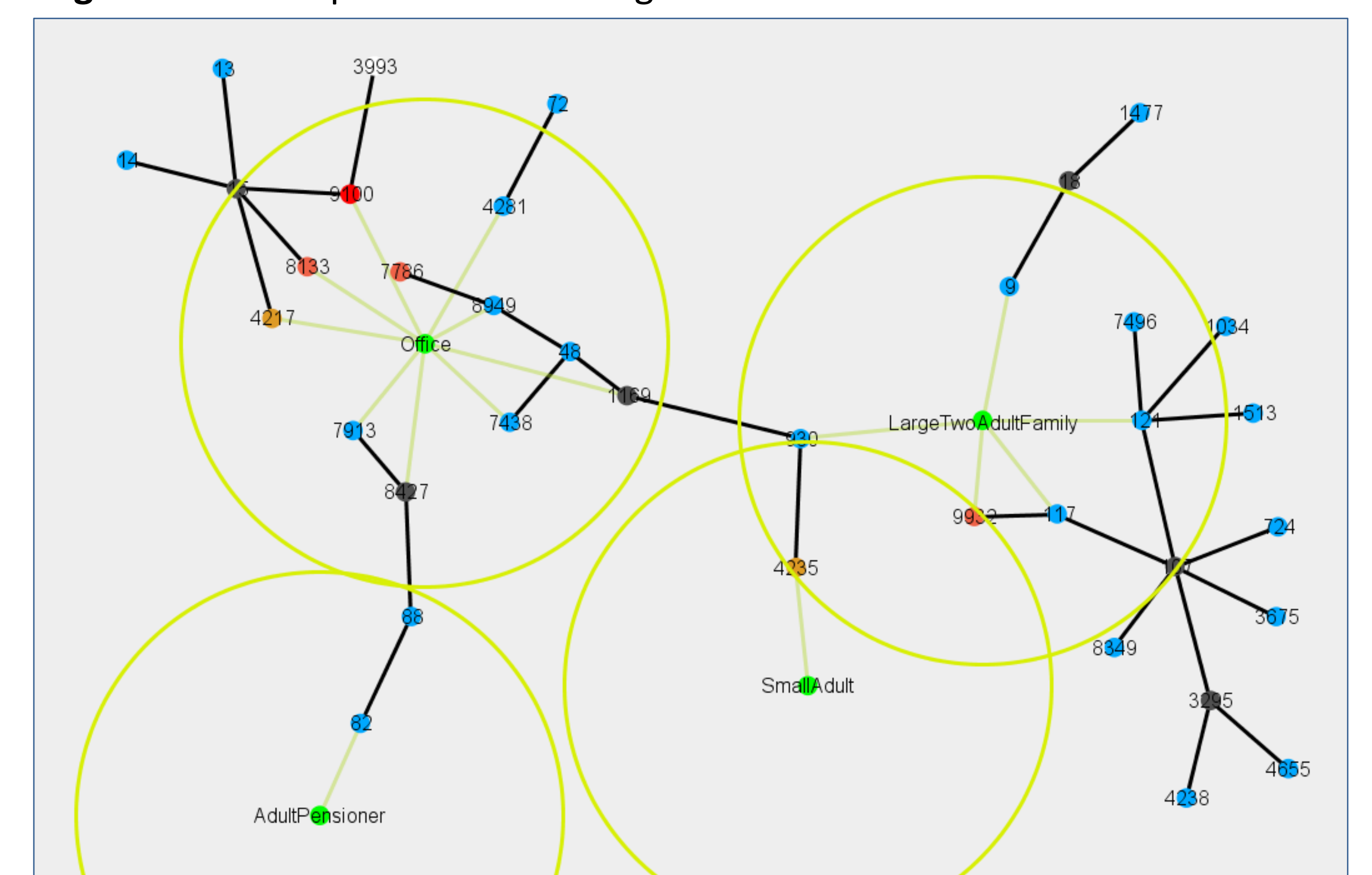


Figure 7. Node colour-encoding scheme

Infection status	Colour encoding
Exposed	Light Green
Asymptomatic	Yellow
Presymptomatic	Orange
Symptomatic	Red
Severely symptomatic	Dark Red
Recovered	Blue
Dead	Black

Contact

Kasper Krawczyk
Swansea University, United Kingdom
Email: 2035886@swansea.ac.uk

Acknowledgements

Special thanks to Max Sondag for his support, sharing his expertise and insights.

References

1. Mohr, S., Matthews, L., Brett, S., Mano, V., Nonweiler, J., Takahashi, R., Townsend, E.: Contact Tracing Model. <https://github.com/ScottishCovidResponse/Contact-Tracing-Model>, accessed: 09-18-2020
2. Simonetto, P., Archambault, D., Kobourov, S.: Drawing dynamic graphs without timeslices. In: International Symposium on Graph Drawing and Network Visualization. pp. 394–409. Springer (2017)
3. Simonetto, P., Archambault, D., Kobourov, S.: Event-based dynamic graph visualisation. *IEEE Transactions on Visualization and Computer Graphics* 26(7), 2373–2386 (2020)
4. Bertault, F.: A force-directed algorithm that preserves edge-crossing properties. *Information Processing Letters* 74(1-2), 7–13 (2000)