Random Spatial Network Models for Core-Periphery Structure

Junteng Jia and Austin R. Benson

jj585@cornell.edu, arb@cs.cornell.edu

Cornell University

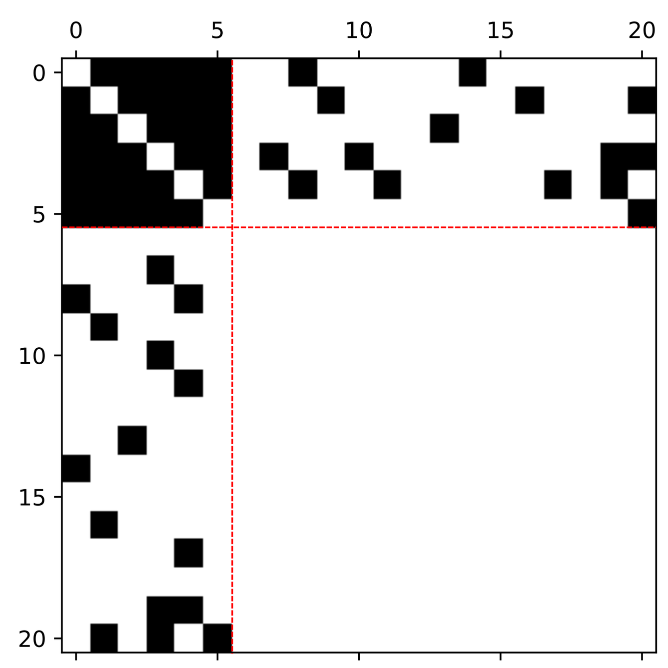
Quick overview

What is core-periphery structure?

Ideal core-periphery structure:

- tightly connected core vertices
- sparse connections between core & periphery vertices
- disconnected periphery vertices

adjacency matrix $\Longrightarrow$



Practically core-periphery structure is:

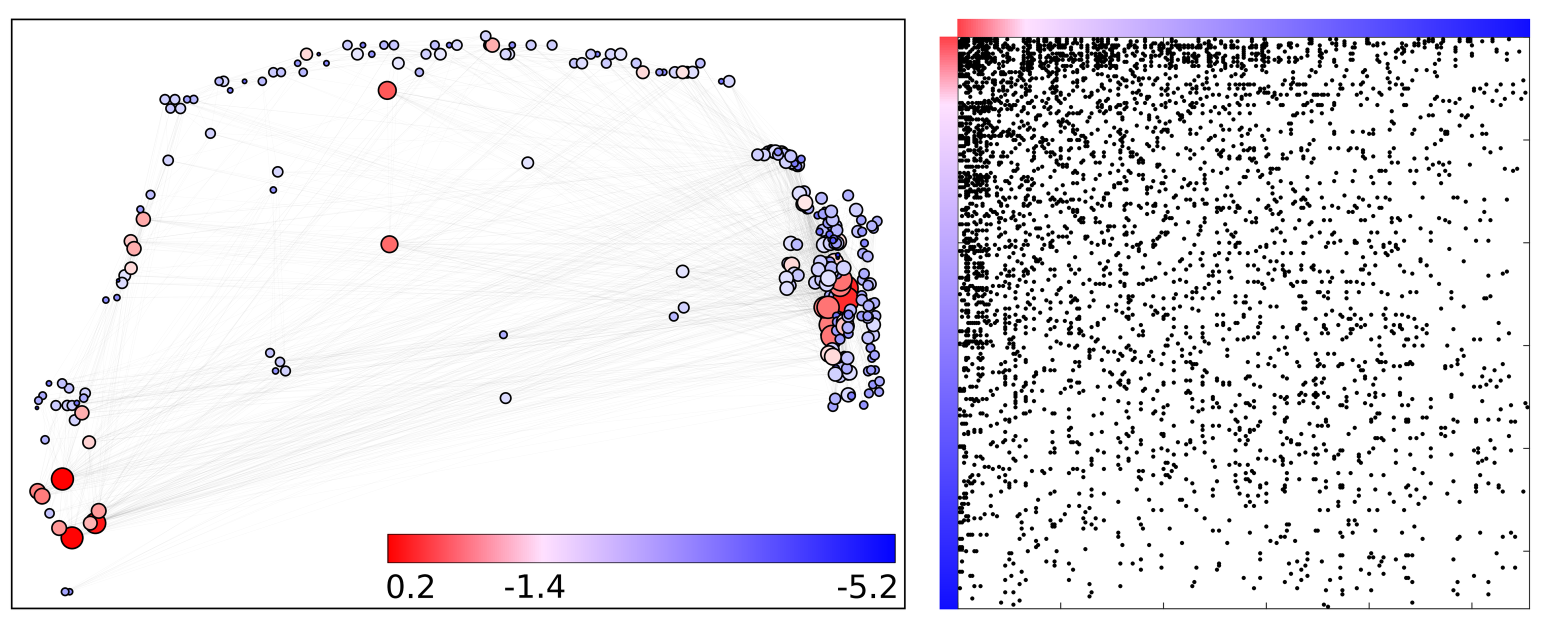
- too complicated to express as 2-block model
- commonly found in real-world spatial networks
- providing a notion of centrality

Our Model:

- uses core scores θ_u for each vertex u, larger value $\rightarrow$ more in core
- posits an intuitive random process for edge generation:
  - core vertices have higher probability ρ_{uv} to connect
  - edge probability decays with spatially distance (K_{uv})
  - a parameter ϵ specifies decay rate

$$\rho_{uv} = e^{\theta_u+\theta_v} \big/ (e^{\theta_u+\theta_v} + K_{uv}^{\epsilon})$$

- given an adjacency matrix A and spatial vertex coordinates, we infer vertex core scores via maximum-likelihood (inferred core scores in the *C.elegans* network are color-coded below)



The size of vertices in the network (left panel) is proportional to the square root of their degree. The vertices in the adjacency matrix (right panel) are ordered by decreasing core scores.

What does our model preserve?

The following log-likelihood objective function is maximized,

$$\Omega = \sum_{u<v} \left[A_{uv} \log \rho_{uv} + (1 - A_{uv}) \log(1 - \rho_{uv})\right]. \tag{1}$$

The stationary conditions with respective to θ and ϵ guarantees two important proprieties of our model.

Stationary Condition 1

$$\frac{\partial \Omega}{\partial \theta_w} = 0 \iff \sum_{u \neq w} A_{wu} = \sum_{u \neq w} \rho_{wu} \tag{2}$$

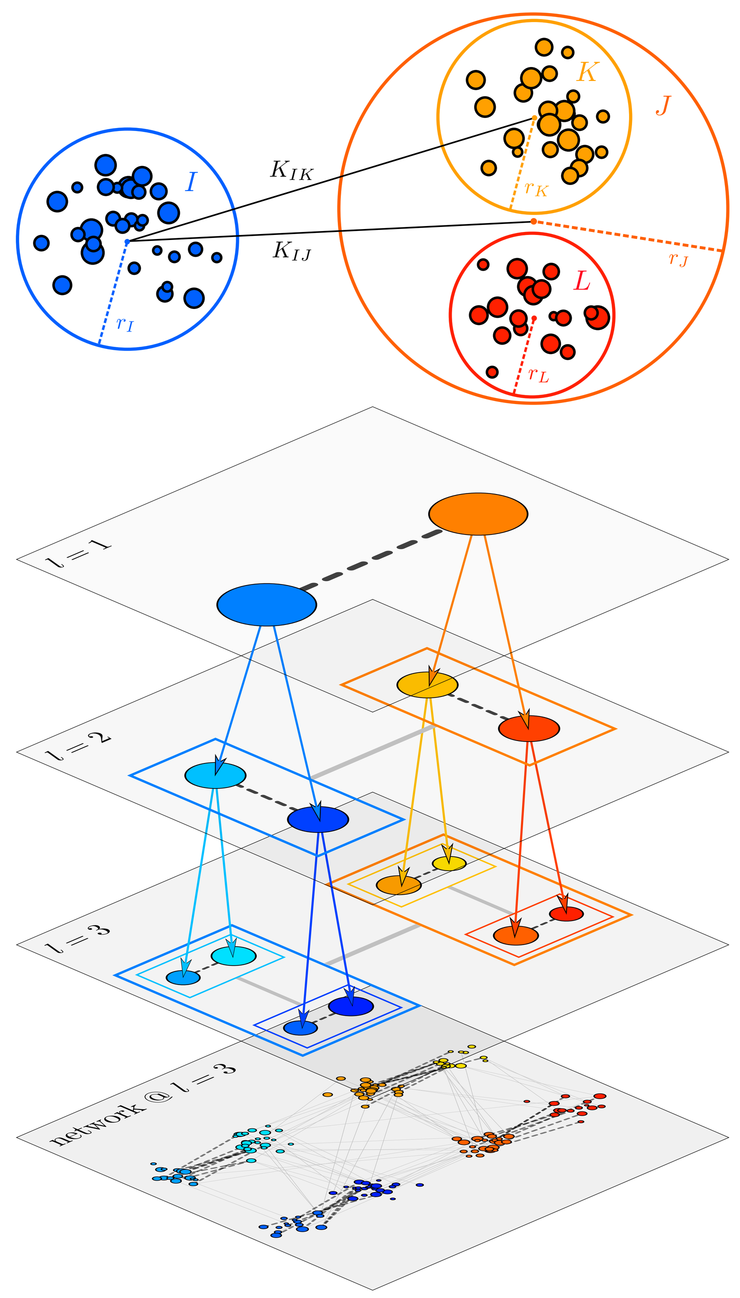
The expected degree of every vertex in the random network model equals its degree in the input network.

Stationary Condition 2

$$\frac{\partial \Omega}{\partial \epsilon} = 0 \iff \sum_{u<v} A_{uv} \log K_{uv} = \sum_{u<v} \rho_{uv} \log K_{uv} \tag{3}$$

The expected aggregated log-distance — which measures the overall edge lengths — of the random network model equals the aggregated log-distance of the input network.

Fast computation

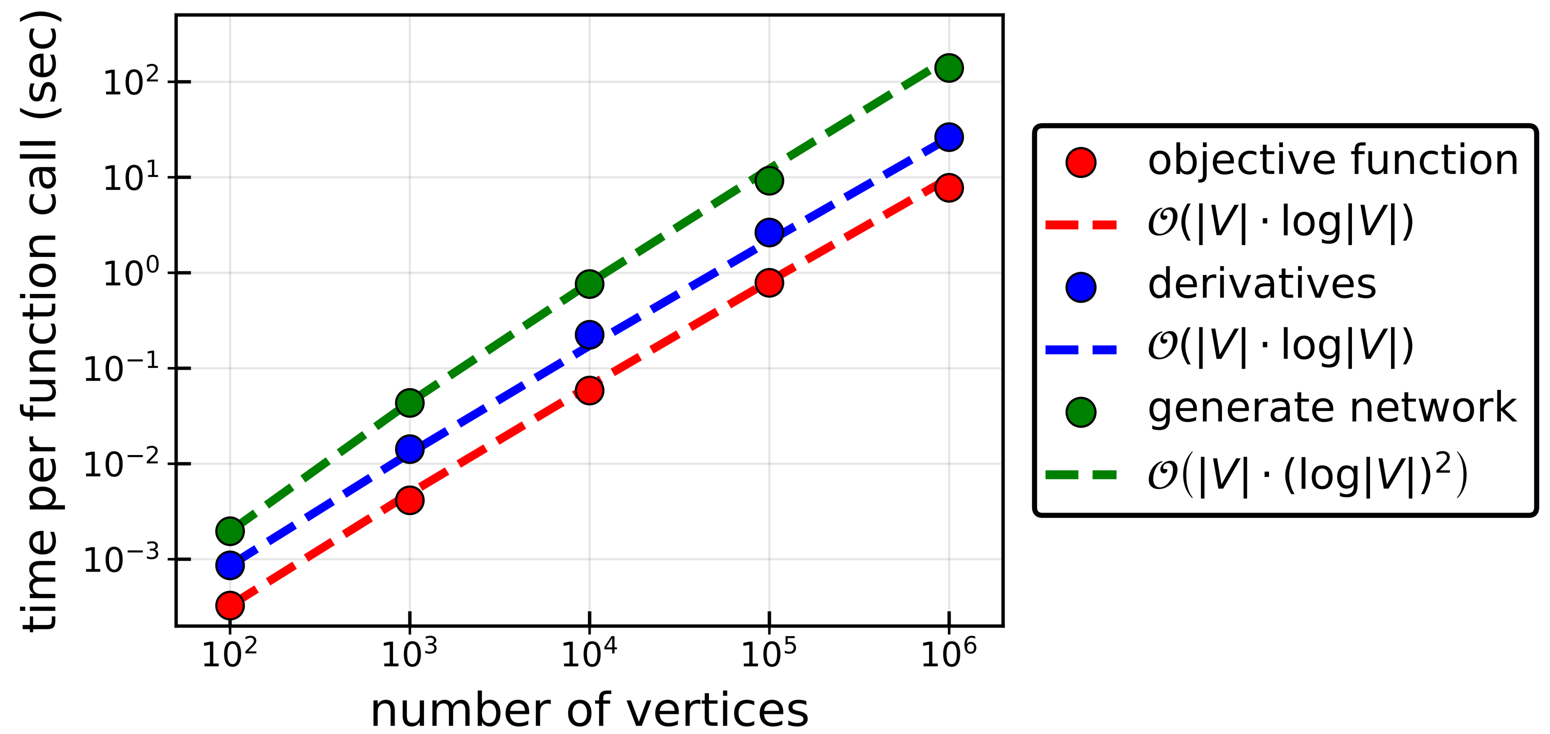


During inference, we maximize the log-likelihood objective with gradient-based method, notice:

- directly evaluating objective and gradients takes $\mathcal{O}(|V|^2)$ **!**
- Equations 1–3 closely resemble many-body simulation
- Faraway vertices contribute very little individually

Use fast multipole method idea:

- sum over the long-range vertex pairs in clusters
- algorithm complexity lowers to $\mathcal{O}(|V| \log |V|)$ **!**
- parameters introduced to trade-off accuracy and complexity



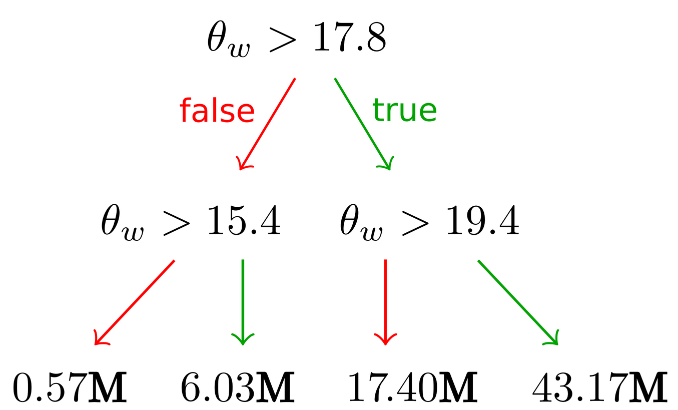
Scalability testing of our fast algorithms on a family of synthetic networks. The observed timings are scattered with circles, which agrees very well with the ideal efficiencies plotted in dashed lines.

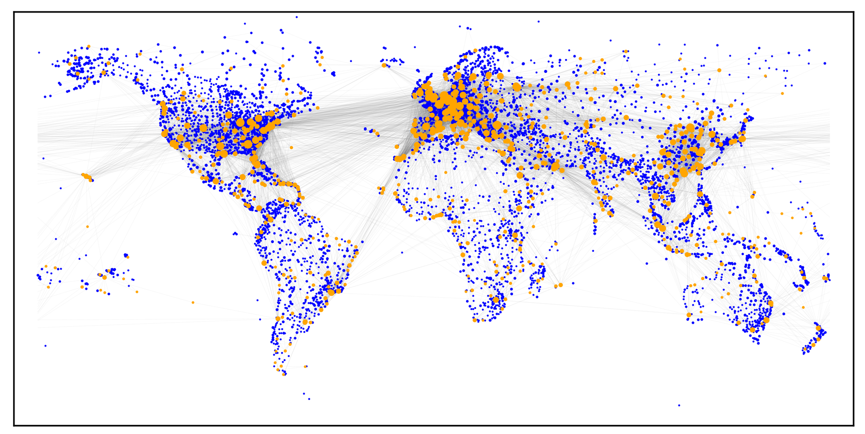
How can we further use our model?

There are two important aspects:

- our model captures a notion of vertex centrality
- comparing with other centrality measures, our model
  - accounts for spatial positioning of vertices
  - gives an explanation for generative core-periphery structure

We compare our core scores against other centrality measures in downstream data mining tasks, e.g., predicting airport enplanement.





We train a decision tree to correlate different centrality measures to airport enplanements. Core scores have the highest test accuracy.

	degree	BC	CC	EC	PR	core score
R^2	0.762	0.293	0.663	0.542	0.637	**0.846**

Other centrality measures are degree, betweenness centrality (BC), closeness centrality (CC), eigenvector centrality (EC), and pagerank (PR). More data-mining experiments are reported in the paper.

This research was supported by NSF award DMS-1830274 and ARO award 86798.