


Parallel Generation of Simple Null Graph Models

Jack Garbus Christopher Brissette George M. Slota

Rensselaer Polytechnic Institute

ParSocial 2020

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.

Community Detection

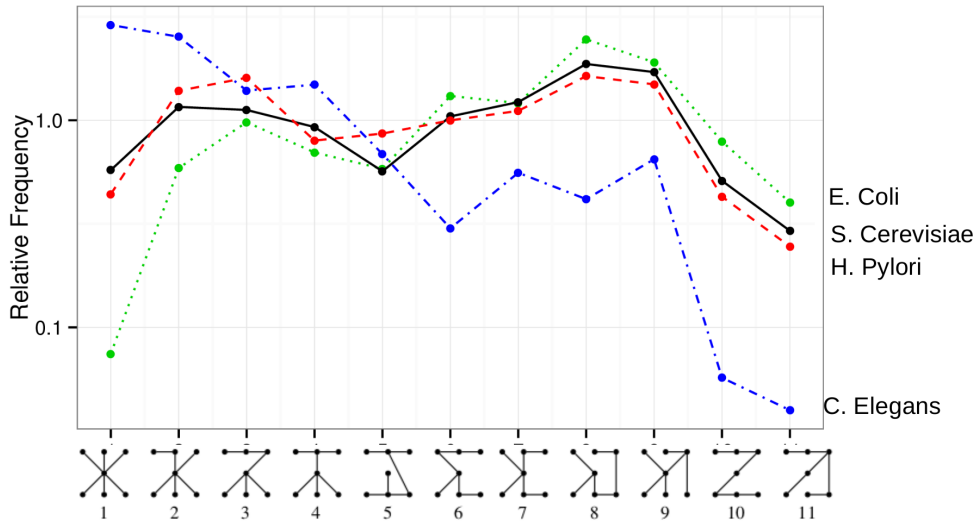
via modularity maximization

Communities: "Dense" subgraphs in a network.

Most common approach: modularity maximization.

$$\max_{v \in c_v, u \in c_u} Q = \frac{1}{2m} \sum_{(u,v) \in E(G)} \left(A_{u,v} - \frac{\mathbf{k_u k_v}}{\mathbf{2m}}\right) \delta(c_u, c_v)$$

Motifs: "Frequently" occurring subgraphs within a network.



Common Idea: Comparison to a Null Graph Model

Null Graph Model: uniformly random graph, usually matching some property such as a **degree distribution**.

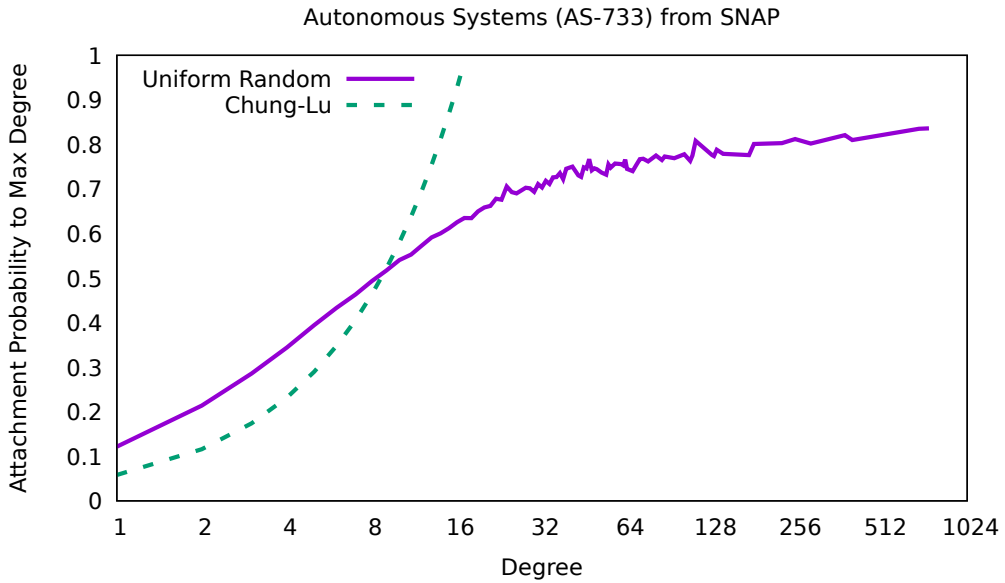
Motif Finding: "Frequent" subgraphs relative to similar networks or a *null graph model*.

Modularity Maximization: "Dense" subgraphs relative to pairwise attachment probabilities of a *null graph model*.

These applications often consider *simple graphs*, but **they commonly utilize an approximation of degree-wise attachment probabilities for non-simple graphs**.

Attachment Probabilities

$\frac{k_u k_v}{2m}$ is probability of edge between u, v in a uniformly random *loopy multi-graph* (e.g., configuration or Chung-Lu models).



- Issue 1: For skewed and/or dense *simple graphs*, **this approximation is very wrong**. Above: AS-773 empirical vs. approximate.
- Issue 2: Using $\frac{k_u k_v}{2m}$ to generate simple graphs can result in considerable error in the output degree distribution.

So what does this mean?

Simply put: using $\frac{k_u k_v}{2m}$ to generate graphs or compute network measurements on simple graphs is probably **bad practice**.

- **Motif Finding**: graph generation gives higher assortativity for large-degree vertices and results in combinatorial explosion for subgraph counts.
- **Modularity Maximization**: attachment probabilities bias towards anti-assortativity in pairwise community membership for large-degree vertices.
- See Fosdick et al. 2018 for a study of these considerations and consequences.

This current work focuses on parallel null graph model generation for motif finding and related applications.

Parallel Gen. of Uniformly Random Simple Graphs

We consider two distinct problems:

1. Generating a random simple graph matching the degree distribution from a given edge list.
2. Generating a random simple graph matching an input degree distribution.

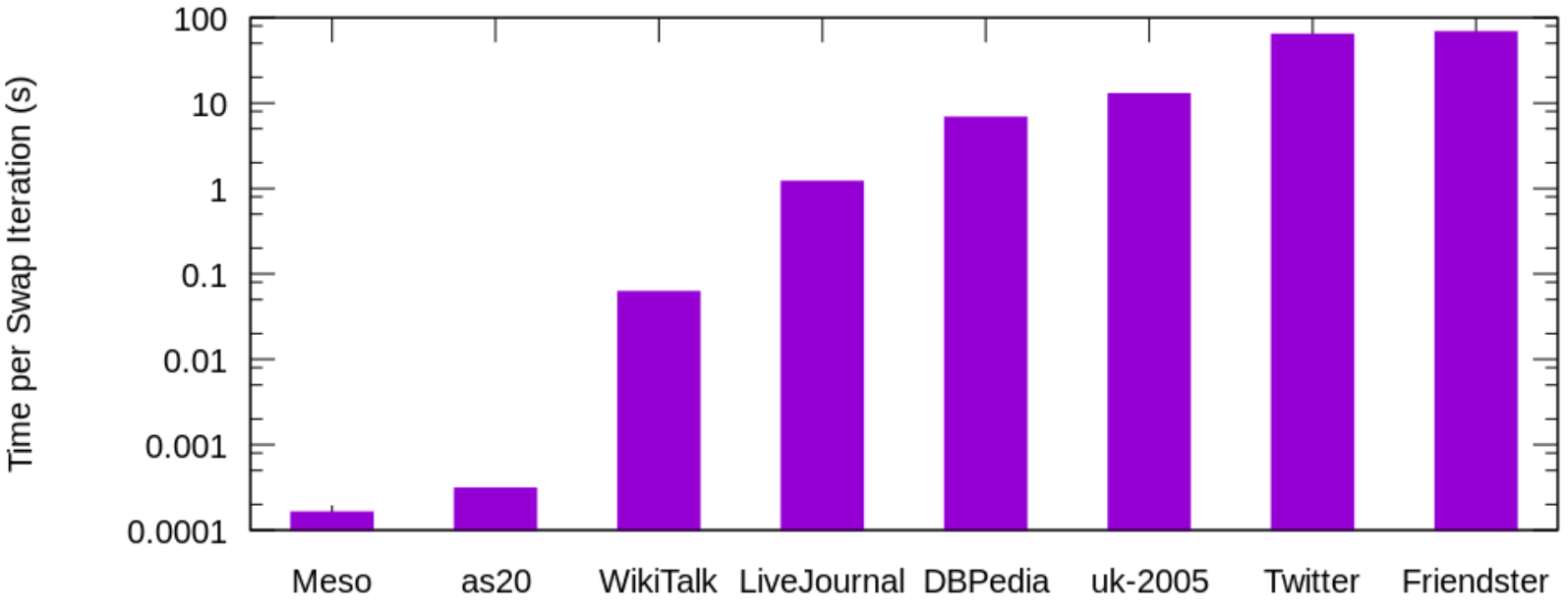
To address these, we implement the following methods:

- **Method 1**: Implement a scalable procedure for parallel *double-edge swaps* (Problems 1 and 2).
- **Method 2**: Implement a parallel way to generate a simple edge list matching in expectation an input degree sequence by calculating attachment probabilities (Problem 2).

Method 1: Scalability of edge-swapping

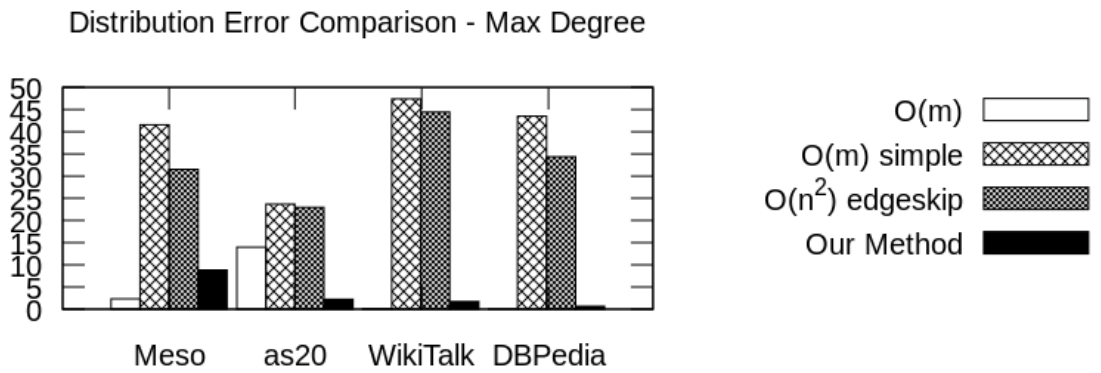
Our edge-swapping routine strong scales very well. Relative to prior work (Bhuiyan et al. 2017), we observe an order-of-magnitude speedup.

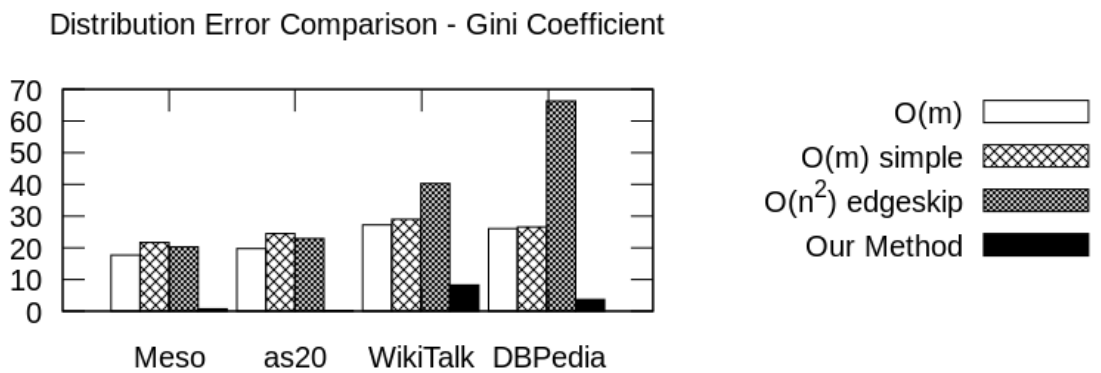
Time for a single iteration (attempting to swap every edge in the edge list) given below for several well-known test inputs:



Method 2: Edge-list generation

Our method for computing attachment probabilities for edge-skipping outputs an edge list that better matches the input maximum degree (top) and Gini coefficient (bottom).





Discussion and open questions

Open questions:

1. **How can we analytically determine uniformly random attachment probabilities for a simple graph?**
   – We have an approximation, but requires $O(n^2|D|^2)$ work and approximation error in practice is unknown.
2. **Can we directly sample from the simple graph space?**
   – Many approaches exist, but are somewhat restrictive (e.g., require $d_{max} < m^{\frac{1}{4}}$).
3. **How many iterations of edge-swapping is required to get a uniformly random graph sample?**
   – Open problem, but we observe about one successful swap per edge is a good approximation.

Conclusions and thanks!

Major takeaways:

- $\frac{k_u k_v}{2m}$ is often "inappropriate" for simple graph generation and applications that use null graph models.
- We develop better faster and better quality methods to quickly output uniformly random graphs, such as for use when comparing motif counts to a null graph model.
- There's still many open questions to consider.

Thank you! Contact below with any questions.

slotag@rpi.edu www.gmslota.com