Efficient Construction of 2K+ Graphs

Minas Gjoka†, Balint Tillman‡, Athina Markopoulou†, Rasmus Pagh ‡
† University of California, Irvine, ‡ ITU Copenhagen
(Preferences: Regular Paper, Poster Presentation)

I. Introduction

Researchers often need to generate synthetic graphs with key properties resembling those of networks of interest. This is for example the case when the full topology is unavailable or impractical to measure or use. In our work, we are interested in generating graphs that resemble online social networks in terms of their joint degree distribution (JDM) and their degree-dependent average clustering coefficient ($\bar{c}(k)$).

The above two graph properties have been chosen based on the systematic framework of dK-series [2], which characterizes the properties of a graph using a series of probability distributions specifying all degree correlations within d-sized subgraphs of a given graph G. Increasing values of d capture progressively more properties of G at the cost of more complex representation of the probability distribution. The joint degree distribution (JDM) and the degree-dependent average clustering coefficient ($\bar{c}(k)$) provide a sweet spot between accurate representation of online social networks and practical constraints in estimation and construction. One one hand, the joint degree distribution (2K- distribution) alone is insufficient for modeling online social networks, which exhibit high clustering. On the other hand, the 3K-distribution captures all information about subgraphs of three nodes, but there are currently no efficient algorithms for estimating 3K parameters or constructing graphs with a target 3K.

We design two algorithms for constructing graphs with (1) an exact 2K distribution and (2) an exact 2K and an approximate $\bar{c}(k)$. Our approach is the first that achieves exact 2K, high clustering, and speed, whereas previous methods could get only two out of three.

II. Exact 2K Construction Algorithm

Prior methods that generate 2K graphs were based either on the configuration model [2], which can produce multigraphs; or on a balanced degree invariant [3], which significantly constrains the edges considered for addition.

We design a new deterministic algorithm that receives as input a target JDM$^{\odot}(k,l)$ and creates a *simple* graph that has provably the *exact* JDM$^{\odot}(k,l)$. First, we create a set of nodes V and assign k_v stubs to every node $v \in V$ according to the target node degree distribution (fully defined by JDM$^{\odot}(k,l)$). Then we add one edge (v,w) at a time between a node v of degree k and a node w of degree l, until it reaches JDM$^{\odot}(k,l)$.

1: **for** $(k,l) \in JDM^{\odot}(k,l)$
2: **while** $JDM(k,l) < JDM^{\odot}(k,l)$
3: Pick disconnected nodes $v \in V_k$ and $w \in V_l$
4: **if** v does not have free stubs
5: neighbor switch for v (preserve 2K + free a stub)
6: if w does not have free stubs
7: neighbor switch for w (preserve 2K + free a stub)
8: add edge between (v, w)
9: $JDM(k,l)$ ++

The intelligence lies in the fact that it is always possible to perform a "neighbor switch" move, and add edge (v,w).

The running time is $O(|E| \cdot k_{max})$, since we add one edge at a time, and a neighbor switch for node v takes $O(k_v)$ time. A key advantage compared to prior approaches [2,3] is the flexibility to (i) visit any order of degree pairs (lines 1-2) (ii) any order of node pairs (v,w) even before completing a degree pair and also to (iii) start with a partially built graph. We exploit this flexibility in the next step, to impose additional network structure (in our case clustering) on top of the joint degree distribution.

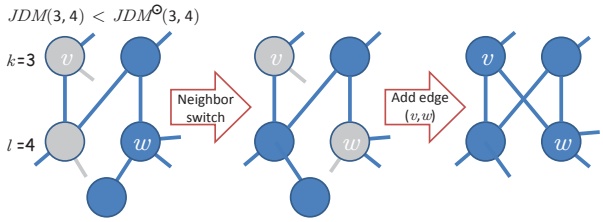

Fig. 1. Example of adding an edge between node v (of degree k) and node w (of degree l). One "neighbor switch" move is needed in this case.

III. Exact 2K with Approximate Clustering

Prior work [2] applied the configuration model to generate a graph with a JDM$^{\odot}(k,l)$ and performed $2K$-preserving double-edge swaps to target 3K. We found this approach to be slow in practice because the starting 2K-graph had very few triangles. We designed a new heuristic approach for generating simple graphs with an exact JDM$^{\odot}(k,l)$ and clustering close to $\bar{c}^{\odot}(k)$ that was orders of magnitude faster [1]. The details of the algorithm are omitted due to lack of space but it consists of two main steps.

First, we exploit the flexibility of our previous 2K construction to construct a graph with the exact 2K but with many triangles (*i.e.,* high clustering). To achieve this goal, we assign to every node v a coordinate r_v randomly selected from interval $(0,1)$. Then, we run the previous 2K exact algorithm, but we add edges in a particular order (in increasing distance on the interval) that ensures that the created edges tend to be local and thus leads to many triangles. At the end of this step, we have a graph with an exact 2K and many triangles. Then, we use an MCMC approach to rewire edges using double-edge swaps and target $\bar{c}(k)$, while preserving 2K. They key insight is that it is easier to destroy triangles, starting from a triangle-rich graph, than to create triangles starting from a triangle-poor graph, as in prior approaches [2]. As an additional optimization, we select the edges for swap, not randomly, but preferring edges with fewer triangles attached, so that after deleting these edges few triangles are destroyed.

Table I shows the results of our approach applied to construct synthetic graphs that resemble (in terms fo 2K+$\bar{c}(k)$) real networks taken from `snap.stanford.edu` and from the Facebook 100 dataset. You can see that the running time is significantly reduced, from weeks to minutes, with the most significant reduction in the case of constructing graphs with high clustrering, which was our goal.

Dataset	$\|V\|$	$\|E\|$	$\bar{c}$	our 2K + smart MCMC	prior 2K + MCMC
FB: UCSD	14 948	443 221	0.227	568 s	177 533 s
FB: Harvard	15 126	824 617	0.212	1 182 s	387 506 s
FB: New Orl.	63 392	816 884	0.222	1 463 s	381 397 s
soc-Epinions	75 877	405 737	0.138	888 s	8 958 s
email-Enron	36 692	183 831	0.497	4 279 s	196 202 s
CAIDA AS	26 475	53 377	0.208	121 s	168 s

TABLE I: Graph generation time in seconds.

References

[1] M. Gjoka, M. Kurant, and A. Markopoulou. 2.5K-Graphs: from Sampling to Generation. In *Proc. of IEEE INFOCOM '13*, Turin, Italy, April 2013.
[2] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat. Systematic topology analysis and generation using degree correlations. In *SIGCOMM*, 2006.
[3] I. Stanton and A. Pinar. Constructing and sampling graphs with a prescribed joint degree distribution using markov chains. *ACM Journal of Experimental Algorithmics*, 2011.