

An interactive tool for analyzing Kronecker Graphs

Huy Nguyen¹, Alan Edelman²
^{1,2}MIT Computer Science and AI
Laboratories
²Department of Mathematics
{huy2n,edelman}@mit.edu

Introduction

Graph analytics are becoming increasingly important for a wide variety of applications. Tools for rapidly analyzing and visualizing such graphs represent an important new applications space for real time systems. This submission presents a tool for synthetically generating and visualizing graphs that are representative of a wide range of real world problems.

Kronecker graphs have recently attracted a lot of attention in the graph mining community. They not only possess many important properties of realistic graphs such as: the power law degree distribution, the power law scree plot, and small diameter; but also can be relatively easy to generate [1]. Leskovec and Faloutsos [2] showed that many mining applications can be built on top of this class of graphs including graph compression, extrapolation, sampling and anonymization. Moreover, they also showed a very elegant and efficient algorithm (linear time) to generate Kronecker graphs that match most global properties of any given real graph. However, we still understand very little about this class of graphs. Recently, Kepner [3] showed that even with simple combination of bipartite graphs and identity graphs as sub-structures, we can achieve a very rich class of Kronecker graphs that have important properties of a realistic graph. This suggests several natural questions: How rich is the set of general Kronecker graphs (with general sub-structures)? Are they any richer than the Kronecker graphs generated by bipartite graphs and identity graphs? If they are, is there a nice relation between the sub-structure and the Kronecker graph in general?

As an effort to attack these questions, we developed an interactive tool to assist computer scientists and algorithm designers to generate and analyze Kronecker graphs easily and efficiently.

Our graph generator algorithm is similar to the stochastic algorithm presented in [1]. However, we generalized the algorithm so that it can generate graphs with an arbitrary number of vertices N .

Our tool contains a set of techniques for analyzing Kronecker graphs including: graph permutation, graph clustering, graph growing and shrinking. The analytical tool also comes with a set of visualization tool to present Kronecker graphs.

Kronecker graph generator

We improved the stochastic algorithm which is shown in [1] to generate Kronecker graphs so that the number of vertices can be any integer (not only a power of n), where n is the size of the sub-structure. For a given N , our algorithm picks n^k - the smallest power of n that is larger than N . We generate a Kronecker graph of size $n^k \times n^k$, and then the identities of the vertices are randomly shuffled. The vertices are then sorted by degree (from high to low) and the N largest degree vertices are selected. This approach allows an arbitrary size Kronecker graph to be generated which preserves many of the global properties of the graph.

Analytical tools

– *Graph Permutation*: Although graph permutation (relabelling) does not change the topology of graph, it creates a different perspective of the graph for a human user. Looking at graphs from different perspectives is very helpful for users since it often helps them see more features of graphs. Currently, our tool provides four different ways to present the adjacency matrix of a Kronecker graph (see figure 1). On the top left is the view of the graph as generated; on the top right is the view of graph after applying partitioning permutations suggested by Kepner [3]; on the bottom left is the view of graph sorted by degree; finally, on the bottom right is the graph shuffled randomly.

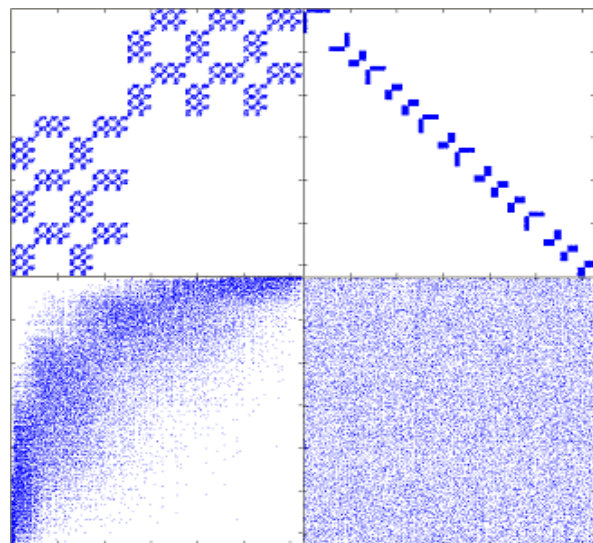


Figure 1: Four different views of the adjacency matrix of a Kronecker graph.

– *Graph partitioning*: When the sub-structure is a bipartite graph or close to bipartite, Kronecker graphs can be partitioned into disjoint (or almost disjoint) parts (as in the top right view of Figure 1) (by using permutations in [3]). This feature is useful when the analyzing process takes super-linear time. Partitioning the graph into small pieces and analyzing each of them separately saves a lot of computation cost as compared to running the analysis process on the whole graph.

– *Graph organic growth and shrinkage*: Our graph generator not only allows us to generate the Kronecker graph of any size but it in fact can also show how such Kronecker graph is growing (or shrinking). These growing (and shrinking) processes should be very useful for researchers who want to study the evolution of a computer network or social network.

– *Visualizing Kronecker graph*: Beside the interactive GUI that user can view the graph in MATLAB® *spy* style. We also created another visualization tool for Kronecker graphs that allows users to render Kronecker graphs growing on a sphere. This rendering is similar to the sphere rendering in [4]. However, thanks to the nice structure of Kronecker graph (clusterable into parts), the embedding from graph to sphere is much simpler and has nice visualization qualities.

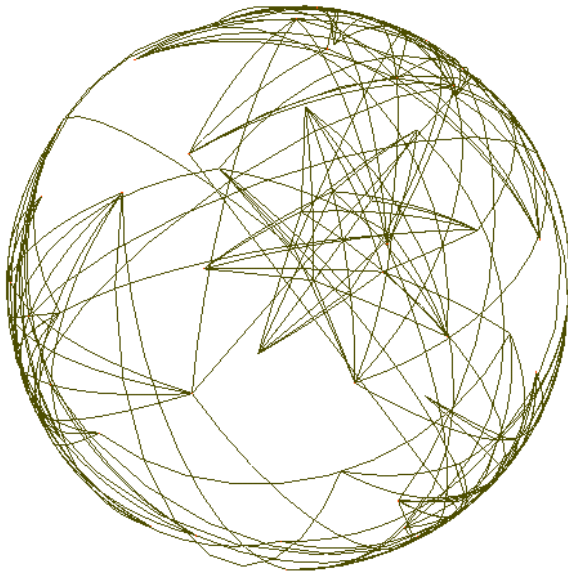


Figure 2: Kronecker graph growing on a sphere

The GUI

Figure 3 shows a screen shot of the GUI which includes 4 main parts:

– *Graph parameters* are tunable by users and specify how the Kronecker graph is generated. These parameters include the sub-structure graph (can be set to any linear combination of bipartite graph and identity graph), the desired edge/node ratio of the generated graph and the Kronecker power.

– *View options* include four different choice of view: Graph as generated, Graph in sorted order of degree, Graph

permuted by the partitioning permutation and Graph shuffled randomly.

– *Graph view* shows the graph in MATLAB® *spy* style.

– *Organic growth (and shrinkage) view* shows how the graph is growing and shrinking over time.

The use of this GUI is simple. Users first set the *graph parameters* as desired and then click on the “Generate” button to generate the graph. The generated graph will be shown on the *graph view* (on the top left chart of the GUI). Users can change the way graph is presented by changing *view options*. The *organic growth* of the graph can be seen on the top right chart of the GUI, a slider bar can be used to move back and forth in time.

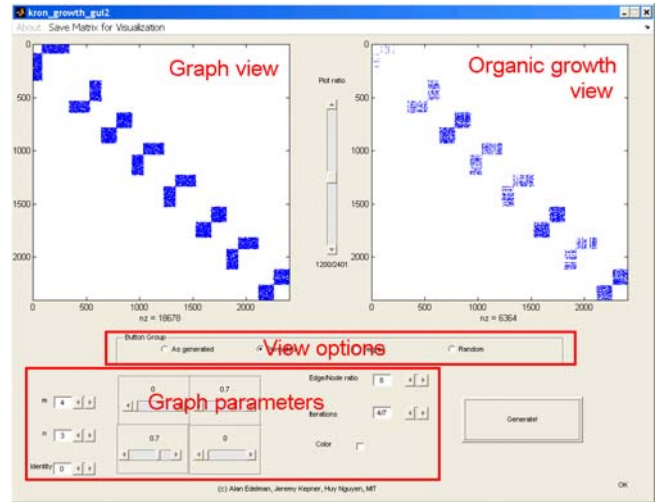


Figure 3: The GUI

The GUI Acknowledgments

We thank John Gilbert and Viral Shah for many enlightening conversations.

This work is supported by NSF Award CCF-0832997 and Akamai Fellowship.

References

- [1] Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos. Realistic, “Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication”, *European Conference on Principles and Practice of Knowledge Discovery in Databases*, 2005.
- [2] Jure Leskovec, Christos Faloutsos, “Scalable Modeling of Real Graphs using Kronecker Multiplication”, *International Conference on Machine Learning (ICML)*, 2007.
- [3] Jeremy Kepner, “Analytic Theory of Power Law Graphs”, *SIAM Parallel Processing 2008, Minisymposium on HPC on Large Graphs*, March 12-14, 2008, Atlanta, GA
- [4] John R. Gilbert, Steven Reinhardt, and Viral Shah. “An interactive environment to manipulate large graphs”. *In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages IV-1201-IV-1204, April 2007