An analysis of preferential attachment distributions in network modeling

Steven Perdomo
COT 5405, Design and Analysis of Algorithms, University of Central Florida

Date: 05/12/20

Abstract

This project expands on the work done by Narsingh and Aurel [1] in helping analyze the behavior of web-like networks. Their paper developed underlying rules, specifically preferential deletion, for simulating the development of a network and created numerical solutions. The goal was to be able to study the degree distribution of the simulation and determine if it was in agreement with previous works. In order to obtain different results from the former article, this project instead maintains a uniform deletion distribution while varying the preferential attachment distribution to determine what combination of simulated network behavior best represents a power law distribution.

1. Introduction

Analyzing the behavior of networks has been a continual effort for research scientists. It has been found that many systems, virtual or real life, draw upon similar behavior. Consequently, the ability to understand the underling properties of a generalized network could provide insight into many areas of interest. Different efforts have created different principles upon which to simulate and analyze these networks.

The article of interest for this paper [1] took into consideration previous works that used dynamic random graph models. A dynamic random graph model can be described as "a discrete-time process which starts out with a small fixed graph and in each subsequent time step a new node is added to the graph or an existing node is deleted from the graph [1]." At each step, the model is updated depending on predefined rules. Even though previous dynamic model efforts have used different rules, the degree distributions have been shown to follow a power-law distribution. Using a combination of preferential attachment and deletion rules [1], also demonstrated this behavior. This project aims to expand on this work by narrowing down a better combination of behavior to create a better network representation. There appeared to be an opportunity to tweak the preferential attachment distribution that was used in the main article. Instead of only using a linear distribution, this project also aims to explore a logarithmic and square distribution in combination with a uniform deletion distribution.

2. Methodology

Python was used to run both the model and numerical calculations and classes were used to represent both simulation and numerical operations. A full test cycle runs for a predefined amount of time steps t using a probability p for birth and q for death to dictate the result at each time step. This was repeated for the three different attachment distributions. Moreover, both classes record the number of nodes and edges throughout their life cycles and also compile their respective degree distributions as lists. These figures of data were then used for visual processing and validation. Numerical calculations for node count, edge count, and degree distribution were based off of the equations of the previous article.

Network Model

The network was modeled using an adjacency list representation with a python dictionary storing each list for the corresponding node. To begin with, a graph of one node with a self loop is created. For each time step, the probability values determine whether a node will be added or deleted from the graph.

In the event of a birth, a new node is added to the graph dictionary and an edge is created to an existing node based on the different probability distributions. Using the *NumPy* random choice function, the target node is randomly selected after calculating the probabilities for each node at the particular time step. In the event of death, a node is chosen to be deleted from the graph and its adjacency list is used to remove any edges from adjacent nodes. The deletion node is chosen based off a uniform probability distribution and similar to birth, *NumPy* is used to process the corresponding probabilities for each node.

The different attachment distributions that were explored are indicated with the following equations:

$$1 : P = \frac{log(d)}{2m}$$

$$2 : P = \frac{d}{2m}$$

$$3 : P = d * \frac{log(d)}{2m}$$

Where d = Number of nodes and m = Number of edges

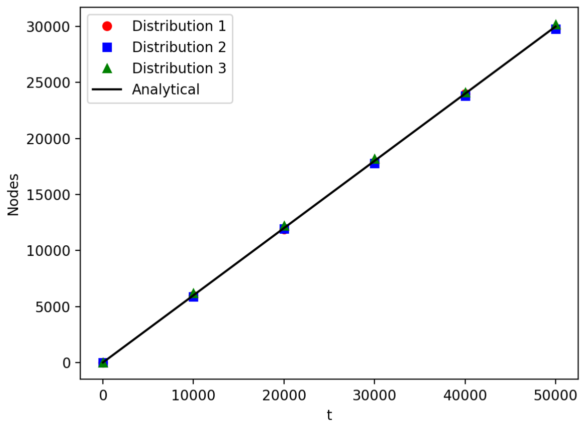

Fig. 1: Comparison of node quantities between distribution modeling and analytical data

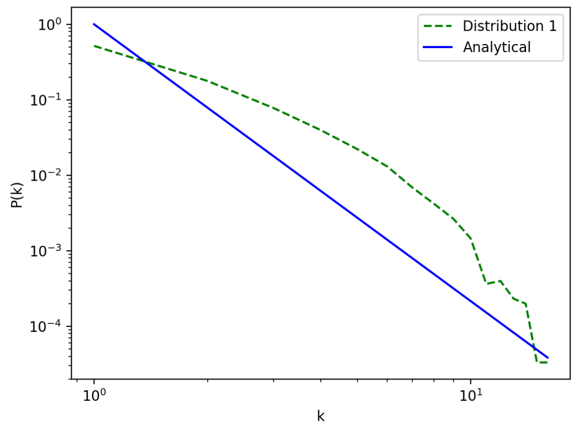

Fig. 3: Comparison of degree distribution data between modeling and analytical data for the first attachment distribution

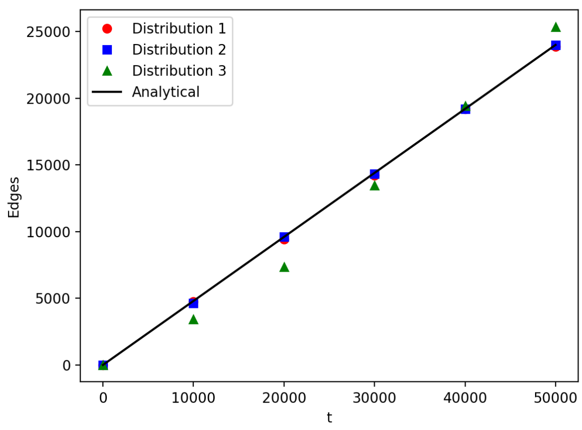

Fig. 2: Comparison of edge quantities between distribution modeling and analytical data

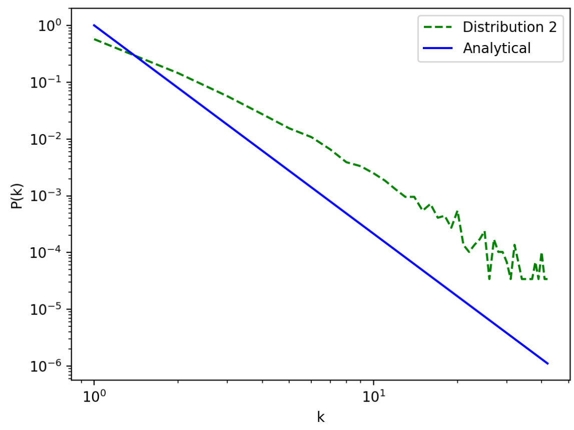

Fig. 4: Comparison of degree distribution data between modeling and analytical data for the third attachment distribution

3. Results and Discussion

In conjunction with the previous paper, the simulation was run with $t = 50000$ time steps but instead with a single p value = 0.8 and with three different preferential attachment distributions. Figures 1 and 2 show the node and edge quantities throughout the entire testing cycle for the three distributions along with the analytical results from the previous article.

These figures are used to show that the different distributions do not greatly impact the network behavior. In fact, the different distributions still yield very similar results when compared to each other and the analytical methods of the main article. Distribution 3 is the only outlier in the plots but only by a relatively negligible amount. This distribution seems to cause a small oscillation around the results of the other distributions and may indicate slight instability in terms of network edge development. Most likely, some nodes with large degrees were removed at certain points during the simulation.

Interestingly, the difference in network behavior is more noticeable in the degree distribution plots. The three distributions were again used in simulating the model this time with the intention of monitoring degree distributions. For the same probability $p = 0.8$, the degree distributions were recorded and compared with analytical results from the previous paper. Figures 3, 4, and 5 demonstrate the differences caused by using different attachment distributions.

The first two distributions created curves that did not exactly match that of a power distribution. This is apparent in the curved nature when plotted on a logarithmic plot. The first distribution matches work from the previous paper. On the other hand, the third distribution created a straighter line like that of a power law distribution and demonstrates the noise that is frequently encountered when plotting real networks on a logarithmic scale. This indicates that a linear probability distribution may not favor larger degree nodes for attachment as much as might occur in real networks. The third distribution might better represent attachment behavior in networks. It also seems to be indicated that the attachment distribution needs to be at least linear to achieve power law-like behavior.

4. Conclusion

This project expanded on the work done by previous efforts to define new rules to better simulate real world net-

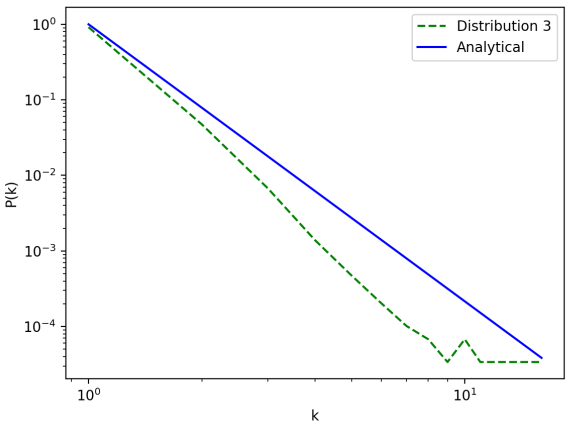

Fig. 5: Comparison of degree distribution data between modeling and analytical data for the second attachment distribution

works. The dynamic random graph model approach is open to different experimentation and combinations of underlying development rules. In comparison to the work in [1], this project tested different attachment distributions to better understand how this specific rule might affect the simulated network. Consequently, it was observed that a slightly larger attachment distribution than the one used in the former paper can yield results closer to that of the power law distribution and closer to real world networks.

Again, this data shows that there are even more properties that can exist in networks that should be investigated to fully predict network behavior. This set of rules helped create a more accurate representation of network behavior, but more investigation into attachment and deletion rules and their impact in combination are bound to lead to insights in the predictability of networks and by extension real world systems.

References

[1] Narsingh Deo and Aurel Cami. "Preferential deletion in dynamic models of web-like networks". Information Processing Letters, vol. 102, no. 4 (2007), pp. 156–162.