



22nd International Conference on Knowledge-Based and
Intelligent Information & Engineering Systems

NetSim – The framework for complex network generator

Akanda Wahid -UI- Ashraf^{a,*}, Marcin Budka^a, Katarzyna Musial^b

^aDepartment of Computing and Informatics, Bournemouth University, Fern Barrow, Poole BH12 5BB, UK

^bAdvanced Analytics Institute, School of Software, Faculty of Engineering and IT, University of Technology Sydney, Australia

Abstract

Networks are everywhere and their many types, including social networks, the Internet, food webs etc., have been studied for the last few decades. However, in real-world networks, it's hard to find examples that can be easily comparable, i.e. have the same density or even number of nodes and edges. We propose a flexible and extensible NetSim framework to understand how properties in different types of networks change with varying number of edges and vertices. Our approach enables to simulate three classical network models (random, small-world and scale-free) with easily adjustable model parameters and network size. To be able to compare different networks, for a single experimental setup we kept the number of edges and vertices fixed across the models. To understand how they change depending on the number of nodes and edges we ran over 30,000 simulations and analysed different network characteristics that cannot be derived analytically. Two of the main findings from the analysis are that the average shortest path does not change with the density of the scale-free network but changes for small-world and random networks; the apparent difference in mean betweenness centrality of the scale-free network compared with random and small-world networks.

© 2018 The Authors. Published by Elsevier Ltd.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Selection and peer-review under responsibility of KES International.

Keywords: complex networks;simulation;random graphs;scale-free networks;small-world networks

1. Introduction

Network properties such as closeness centrality, betweenness centrality, degree distribution, clustering coefficient, and average geodesic path vary depending on the network's type and size. These properties contain essential information about the network's structure and dynamics. For instance, a high clustering coefficient with low average geodesic path might indicate a social network, i.e. a specific type of network that is formed through social interactions [18]. Degree distribution could indicate a particular growth mechanism unique to a specific kind of network – power law in the degree distribution indicates that the growth mechanism is likely to be based on preferential attachment and the network has been formed based on the 'rich get richer' rule [1]. To understand how different properties of a network

* Corresponding author

E-mail address: aashraf@bournemouth.ac.uk

behave with regards to the type of the network, and how properties change with network's size, we have designed and performed a number of simulations. In these simulations we have compared properties of three network models: 1) Barabási-Albert model for scale-free network 2) Watts-Strogatz model for the small-world network, and 3) Erdős-Rényi model for the random network. To compare properties across three different network models of varying size and density, all these models for a particular size need the same number of vertices and edges. Due to the different underlying mechanisms of these three network models, achieving this is not trivial. The difficulty is encountered mainly with scale-free and small-world network models, and we discuss it further in Section 3.2. To obtain comparable results we have developed a set of mathematical formulas that allow generating networks following these three models with a particular number of vertices and edges. This, together with a network simulator that can be easily extended to include other network models and properties, is the primary contribution of our work. The developed simulator, together with generated networks have been open-sourced on GitHub as an R package¹. The entire simulation workflow has been automated, so the user only needs to define input parameters to get a comprehensive set of results in the shape of multiple plots and tables. The second and equally important contribution of this research is the comprehensive analysis of the results of over 30,000 simulations, run to generate and investigate networks following different network models.

We have measured three types of network properties: 1) centralities (betweenness and closeness), 2) average shortest path, and 3) global clustering coefficient. For a given network model, we varied the number of vertices and edges, yet keeping them fixed across all three network types to make the results comparable.

In Section 2 we present a brief overview of existing network generators. Next, the proposed **NetSim framework** together with implemented network models are introduced in Section 3. In Section 4, we describe the design of the experiment that enables comparison of properties of implemented models, followed by results analysis in Section 5. Finally, conclusions are provided in Section 6.

2. Brief Overview of Network Generators Landscape

To the best of our knowledge, there does not exist a framework which generates comparable simulated networks in a comprehensive way for the random graph, scale-free, and small-world networks with a wide range of parameters but keeping the number of edges and vertices equal for all networks.

The *igraph* package [2] is a collection of network analysis tools available in R, Python and C. In our developed framework we have used *igraph* library in R for generating individual models and calculating network properties. However, our NetSim framework allows to generate all three models, with constant number of edges and vertices for a range of different parameters and to save all the calculated properties as serialised objects which are then used to present the results in a transparent manner with generated plots. The whole process is automated and the user does not need to worry about keeping the number of edges and vertices the same, only vertices are taken as inputs.

Brain Connectivity Toolbox [14] is a MATLAB toolbox for complex network analysis of structural and functional brain connectivity datasets. This toolbox has several models, including random and scale-free networks, and includes tools for network comparison which however focus mainly on different community structures, not network properties.

MatlabBGL[6] is another MATLAB package for working with graphs. It is based on the Boost Graph Library and enables generation of simple models like random graphs and cycle graph [15].

Stanford Network Analysis Platform (SNAP) [7] is a library written in C++ for high-performance large-scale complex network analysis. It allows to generate regular and random graphs.

BRITE [8] is another network topology generator. The main focus of this generator is to simulate network that mimics the characteristics of the actual Internet network. However, some important features are not considered in BRITE generator as the main goal is to reflect internet topology. One such not implemented feature is the small-world network model which the authors have pointed out in the documentation [8].

There exist a plethora of other software packages that allow generating networks. While those presented above are just a few examples, it is important to emphasise that, to the best of our knowledge, none of them enables to generate, in an automated way, networks that are easily comparable. Existing tools enable to generate networks of different models, but each model is used in isolation. If the user wants to create networks that can be compared, i.e. having

¹ <https://github.com/AkandaAshraf/netsim>

the same number of nodes and edges regardless of their structure, they have to work out all parameters by themselves which, as presented in the next section, is not a trivial task.

3. Proposed NetSim Framework

In this section, we introduce the network models implemented in the NetSim framework as well as our approach to generate networks with a consistent number of nodes/edges regardless of their structure. We have mathematically derived and empirically demonstrated our framework to be able to simulate and support analysis with varying parameters of three different network models. Our proposed framework, which has been turned into an open-source R package, NetSim, is extremely user-friendly. We can simulate large-scale networks once, calculate their properties and then save those network properties as serialised objects which can then later be fetched from hard-drive to generate plots and perform comparative analysis. This reduces the complexity of the process as generating a large scale network is computationally expensive. Our R package is based on *igraph* [2] and *ggplot2* [19], where the former is used for the core graph calculations and the latter for generating high resolution plots.

3.1. Network models

In our study, we have simulated three widely used network models: (1) Barabási-Albert model for the scale-free network [1], (2) Watts-Strogatz small-world model for the small-world network [18], and (3) Erdős-Rényi model for the random graph network [17, 3, 4, 5]. A brief introduction of these models is given below.

Random Graph: In the random graph network model, one creates a network with some properties of interest (specific degree distribution) and otherwise random. Although random graph model was first studied by [17] this model is mainly associated with Paul Erdős and Alfréd Rényi [3, 4, 5]. The random graph model could also be thought of as choosing uniformly a network from the set of all possible networks with exactly n vertices and m edges. This model is often referred to as $G(n, m)$ model for networks where n is the number of nodes and m being the number of edges. A slightly different definition is $G(n, p)$ model where p defines the probability of edges appearing between all possible pairs of vertices.

Scale-free Model: The scale-free model shows power law node degree distribution $P(k) \sim k^{-\alpha}$ (where k – node degree and α – parameter whose value is typically in the range $2 < \alpha < 3$) which usually indicates some interesting underlying process [9, 11]. This kind of distribution was first discussed by Price [13]. Price, in turn, was inspired by Herbert Simon, who discusses power law in a variety of non-network economic data [16]. Simon showed mathematically the fact that ‘rich get richer’ effect results in power law distribution. He called this mechanism ‘cumulative advantage’, but it is more often known as ‘preferential attachment’ as coined by Barabási and Albert [1].

Small-world Model: Transitivity measured by the network clustering coefficient is one of the least understood properties in network analysis [10]. Another important property we see in real networks is the small-world effect – all nodes are connected with each other by relatively short paths. To model these two properties Watts and Strogatz introduced a small-world network model [18]. Classical small-world model rewires edges in a simple circle model to random positions. In a slightly modified version of the model, the whole process starts from a circle and no edges are removed, but new edges are inserted between randomly chosen vertices [12]. There is an important parameter in this model p – the probability of rewiring/adding edges.

3.2. Network Generation Approach

To compare properties of the three network models, two parameters have been chosen to be fixed. These are the number of edges m and the number of vertices n . For the random graph model, it is simple as we use the $G(n, m)$ model and generate the ensemble directly.

In case of a scale-free network, because this is a growing network model, the final number of edges in the network depends on how many edges are attached in each step S as well as on the total number of vertices n .

The total number of edges in a scale-free network with n nodes and S edges added in each growth step is:

$$m = n \cdot S - \frac{S \cdot (S + 1)}{2}. \quad (1)$$

For a small-world network, the total number of edges for a network with n vertices and Nei number of neighbourhood within which the vertices of the lattice will be connected are:

$$m = n \cdot Nei. \quad (2)$$

To keep the number of edges equal to m for both the small-world network and the scale-free network (i.e. same as in Equation 1) we randomly delete a fixed number of edges x from the small-world network in Equation 2. As a result, the total number of edges for a small-world network is:

$$m = n \cdot Nei - x. \quad (3)$$

If we now solve both Equations 1 and 3 for x :

$$x = n \cdot (Nei - S) + \frac{S \cdot (S + 1)}{2} \quad (4)$$

From Equation 4, we can see that number of deleted edges does not depend on the number of vertices when Nei and S are equal as the first part of the equation on the right-hand side becomes 0. Due to this, in our comparison, we have considered S and Nei parameters to be equal. When it is 0, $\frac{S(S+1)}{2}$ might not have an impact on the type of the model for a large number of vertices as long as S is comparatively small. In our comparisons, we have considered S up to 16. The value is chosen as power of 2, where the power is from 1 to 4 i.e. $S \in \{2, 4, 8, 16\}$.

We have generated networks for scale-free and random graphs with n vertices and m edges. The number of edges is calculated from Equation 1, which is the number of edges we get in scale-free networks for a different number of vertices n . For small-world networks, first, a lattice with n vertices and Nei number of neighbourhood within which the vertices of the lattice will be connected is generated. From the generated lattice, x edges (Equation 4) are randomly removed. After that, the edges are rewired with probability p to obtain small-world network.

Following the process described above, we obtained networks of the three discussed models that are comparable as they have the same number of nodes and edges which, in turn, enables comparative analysis of different networks' properties including clustering coefficient, different types of centralities and average shortest paths.

4. Design of the Experiment

We have varied a number of parameters to generate different networks within three models. For all three network types we have $n \in \{100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000\}$ vertices. For scale-free networks $\alpha \in \{1.5, 1.75, 2, 2.25, 2.5, 2.75, 3, 3.25, 3.5\}$ and for small-world networks $p \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ are used.

The number of edges for scale-free and small-world networks is calculated using Equations 1 and 3 respectively. For the random graph, we have used $G(n, m)$ model where the number of edges is directly defined. Note, that number of edges will be the same for all networks for a given set of S and Nei values. Table 1 shows the number of edges which correspond to each S and Nei parameters for each vertex count defined from set n .

Table 1: Number of edges for different values of S and Nei

S	Nei	Number of edges in a network
2	2	197, 397, 597, 797, 997, 1197, 1397, 1597, 1797, 1997, 3997, 5997, 7997, 9997, 11997, 13997, 15997, 17997, 19997
4	4	390, 790, 1190, 1590, 1990, 2390, 2790, 3190, 3590, 3990, 7990, 11990, 15990, 19990, 23990, 27990, 31990, 35990, 39990
8	8	764, 1564, 2364, 3164, 3964, 4764, 5564, 6364, 7164, 7964, 15964, 23964, 31964, 39964, 47964, 55964, 63964, 71964, 79964
16	16	1464, 3064, 4664, 6264, 7864, 9464, 11064, 12664, 14264, 15864, 31864, 47864, 63864, 79864, 95864, 111864, 127864, 143864, 159864

Table 2: Number of simulated networks.

Network Model	$ n $	$ \alpha $	$ p $	$ S $ or $ Nei $	Sampling
Scale-free	19	$19 \times 9 = 171$		$171 \times 4 = 684$	$684 \times 30 = 20520$
Small-world	19		$19 \times 5 = 95$	$95 \times 4 = 380$	$380 \times 30 = 11400$
Random Graph	19	19		19	$19 \times 30 = 570$
Total				1083	32490

In our simulation, we have shown empirically that our method keeps the number of edges and vertices the same for all types of networks and is consistent with Equations 1 and 3.

For each combination of α and p , both S and Nei are increased as a power of 2. $S, Nei \in \{2, 4, 8, 16\}$. The reason behind keeping S and Nei the same is described in Section 3.2. Each of the networks was generated 30 times, and the mean for each analysed property was calculated over those 30 samples. Table 2 summarises the total number of generated networks. Without sampling, we have in total 1,083 different types of networks with different parameters.

5. Results and analysis

In this section, we present results that were obtained using the NetSim framework for five different properties and three different network models with varying size and density.

5.1. Number of Edges and Vertices

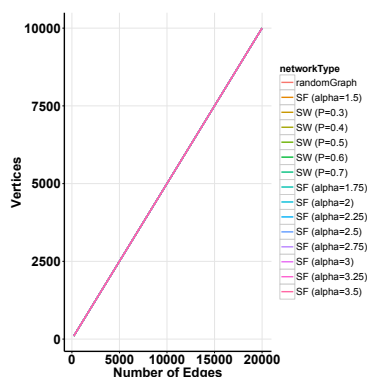
Fig. 1: Edge and Vertices plot for random graph networks, small-world networks, and scale-free networks; $S = 2 Nei = 2$.

Figure 1 shows the number of edges in relation to the number of vertices for different networks. Plots for all networks overlap and form a straight line. This is because in each of the networks we have the same number of

vertices and edges which shows empirically that our proposed method described in Section 3.2 is viable. In Figure 1 the number of edges in relation to the number of vertices is presented for $Nei = 2$ and $S = 2$ and the same trend holds for all other experimental settings from Table 1.

5.2. Closeness Centrality

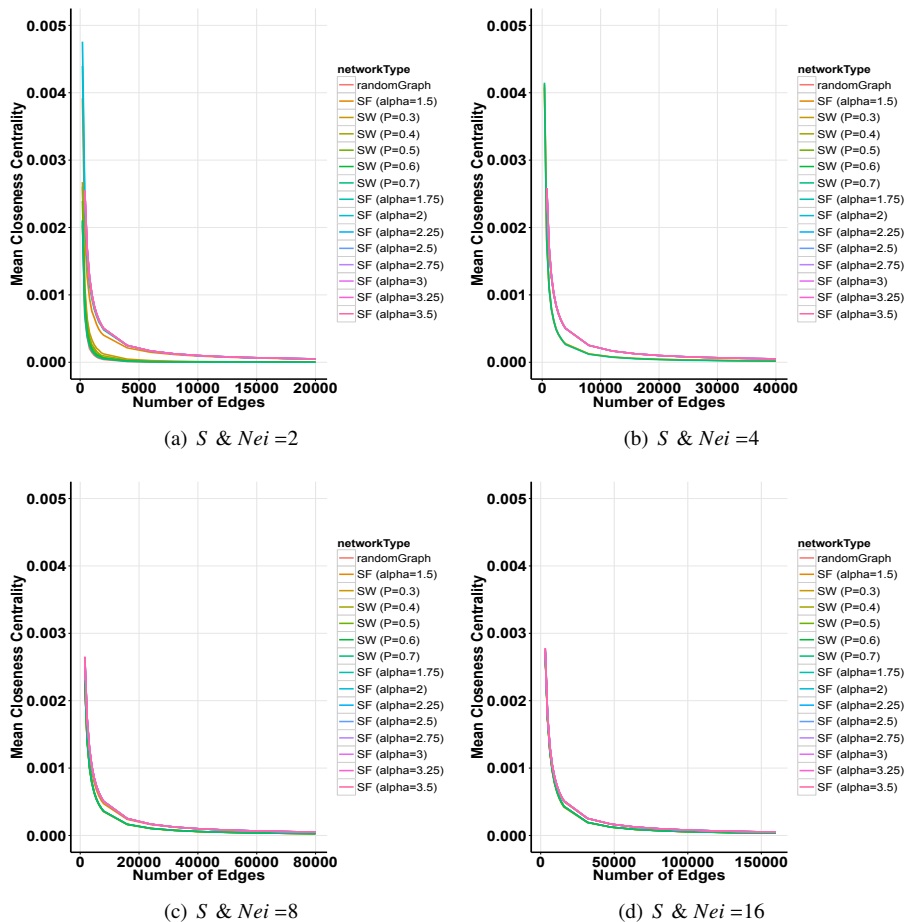


Fig. 2: Closeness Centrality in relation to edges for random graph, small-world, and scale-free networks with different values of S and Nei .

Figure 2 shows mean closeness centrality in relation to the number of edges. Closeness centrality is expected to have higher values in social networks due to the small-world effect, which implies that every vertex should be connected with other vertices via a short path even if the network is large.

As network size is increased closeness centrality decreases for all types of networks. This is because all generated networks become sparser as they grow in terms of the number of vertices (i.e. moving from left to right of a single plot). Although we have increased the number of vertices and edges together, the number of edges is not increased sufficiently with respect to vertices to maintain a dense network. However, increasing the value of S and Nei implies more edges, thus a denser network. For the value of 16 of both Nei and S , the simulation results in networks with the biggest number of edges among all generated networks. For $Nei = 16$ and $S = 16$ mean closeness centrality is similar for all generated networks. This implies that as the network density grows, different types of networks might exhibit similar behaviour in terms of closeness centrality.

When comparing different network types, one can see that random graph has the lowest value for mean closeness centrality. This is something expected as a purely randomly grown network does not have the small-world effect.

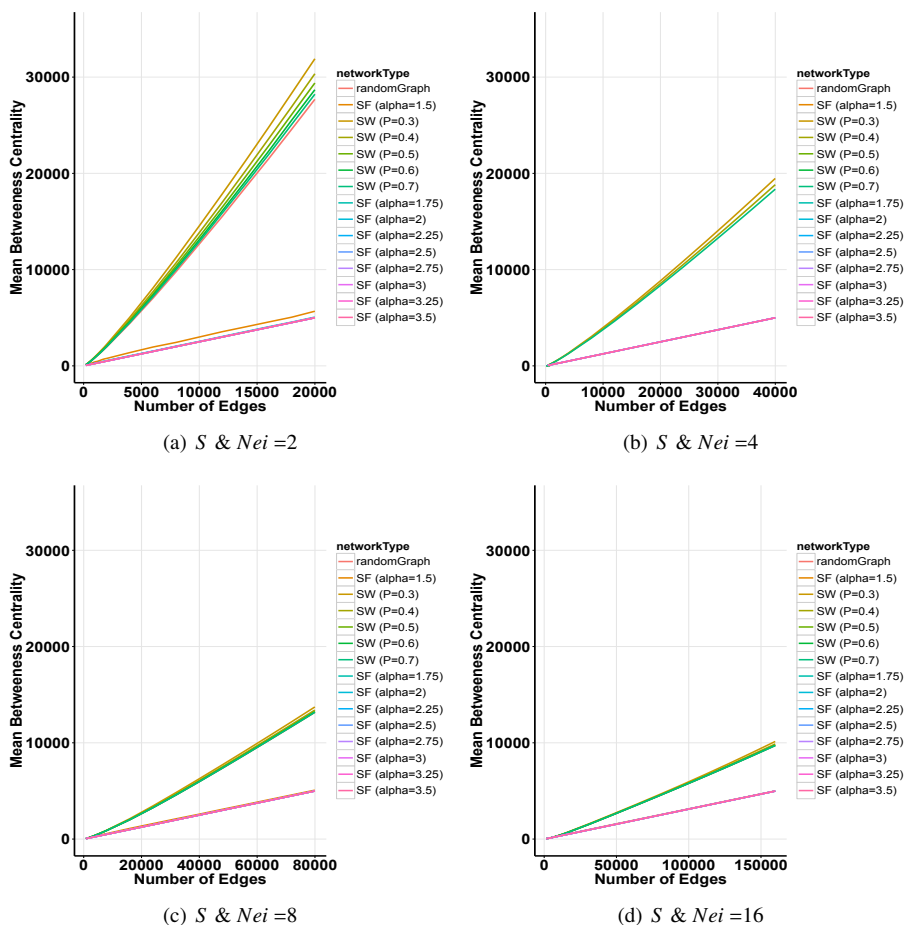


Fig. 3: Betweenness Centrality in relation to edges for random graph, small-world, and scale-free networks with different values of S and Nei .

Scale-free networks with higher values of α seem to have a higher value of mean closeness centrality. This might, in turn, imply that the ‘rich get richer’ effect has a positive impact on the mean closeness centrality of the entire network.

5.3. Betweenness Centrality

Betweenness centrality is a measure of how much influence a vertex has on the information flow within a network. We see a linear increase in the mean betweenness centrality values for all the networks with increasing size of the network (Figure 4). This is because as a network becomes bigger regarding the number of vertices, each vertex lies between more vertices in general.

One important observation here is that there are two clusters of networks in each of the plots. There is a clear difference between scale-free networks and small-world/random networks. As we make networks denser by increasing values for S and Nei , networks in those two clusters remain separate but tend to behave more like each other within the same cluster. We also see that as the network gets denser, betweenness centrality decreases.

Scale-free networks tend to have lower values of betweenness centrality, and the value decreases with growing values of α . This can imply that the ‘rich get richer’ effect is responsible for the decrease in mean betweenness centrality values. Although for small-world and random networks this centrality measure is far higher than scale-free networks, randomness tends to play a negative role here. We see that with small-world networks, higher values of p tend to have lower mean betweenness centrality although they lie in the upper cluster.

5.4. Average Shortest Path

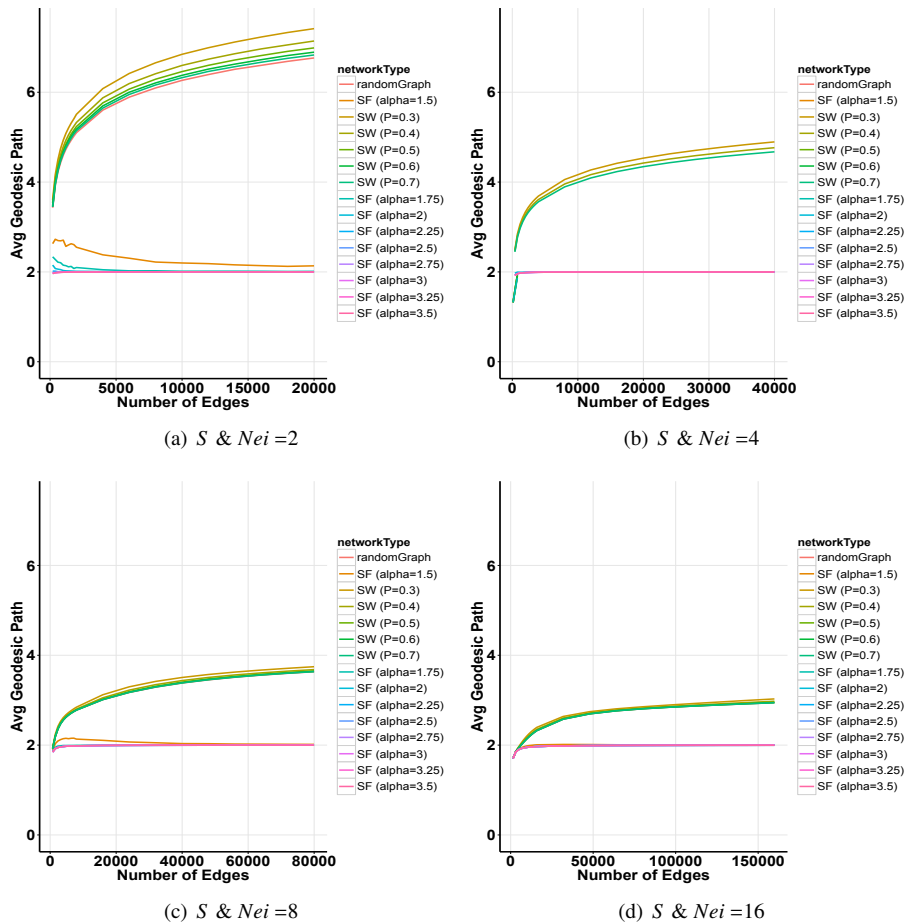


Fig. 4: Average Shortest Path in relation to edges for random graph, small-world, and scale-free networks with different values of S and Nei .

The average shortest path is calculated as the mean shortest path for all pairs of vertices. In our analysis, the average of shortest paths between all pairs of vertices is plotted against the number of edges in Figure 4.

It is interesting to see that when S and Nei are increased, resulting in a denser network, the shortest path is decreased for all three types of networks with different parameters. This implies that the small-world phenomena are more apparent in a denser network and this is true for every small-world, random graph, and scale-free network.

Small-world networks with lower values of p tend to have longer average shortest paths, whereas networks with higher values of p have shorter average shortest path. This is intuitive as having lower values of p means fewer connections are rewired in our networks from the initial lattice which has a high clustering coefficient but lower intensity of the small-world phenomenon i.e. longer shortest path. When the value of p is increased, it gets closer to a random network, which has a shorter average geodesic path. We can also see this phenomenon for the simulated random networks, which have a very short average geodesic path.

Interestingly, size of the network does not seem to influence the value of average geodesic/shortest path of scale-free networks, and they stay almost constant. Whereas, for the cases of small-world and random graph networks, as the number of edges and vertices increase, the average shortest path also increases. In fact, for some of the scale-free networks, increasing the size of the network tends to result in smaller values of the average geodesic paths. This

is expected, as new edges tend to be attached to vertices where we already have edges due to the ‘rich get richer’ phenomena. This results in a stable value of average geodesic path with regards to the size of the network.

One important note for all the plots in Figure 4 is that as we go right on the x -axis, we have more edges and vertices as detailed in Table 1. However, with a higher number of vertices, the network tends to become sparser as we are not increasing the number of edges at a rate which will keep the density constant. On a single plot, when we move from left to right, we have sparser and larger networks while in four plots with different parameters of S and Nei , increasing the number of S and Nei results in denser networks.

5.5. Global Clustering Coefficient

The global clustering coefficient (a.k.a. transitivity) is measured as the ratio of the triangles (for three vertices it includes three closed triplets) to the connected triples in networks.

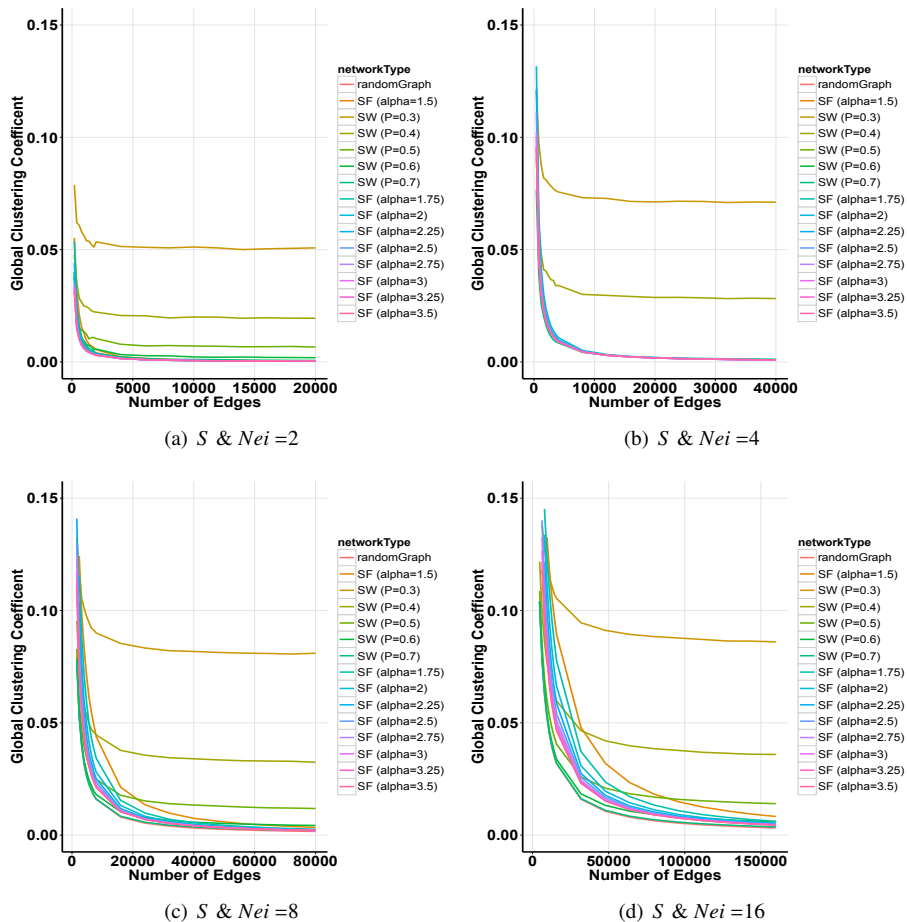


Fig. 5: Global Clustering Coefficient in relation to edges for random graph, small-world, and scale-free networks with different S and Nei .

We observe larger values for transitivity for small-world networks with lower values of p , which is expected (Figure 5). The lower the values of p , the smaller the number of connections that are rewired, and the network is closer to the initial lattice which has a higher number of connected triples. Also, scale-free network tends to have lower values of transitivity and it decreases with increasing α (Figure 5). This is also expected as a scale-free network is biased towards already existing links, and they are not evenly spread, resulting in a lower number of connected triples. Another important observation is that transitivity seems to be constant with respect to increasing number of edges and vertices but not density. With denser networks (increasing S and Nei) transitivity tends to decrease until it

reaches a stable point. This is more visible with small-world networks that have a smaller number of edges rewired. Other networks seem not to be significantly influenced by the changes in density.

6. Conclusions

Our NetSim software allowed us to conduct a simulation study that resulted in comparative analysis of three main network models: random, small-world and scale-free. For all generated networks, we analysed closeness and betweenness centrality as well as average shortest path and global clustering coefficient.

Our experiments allowed to compare selected properties of different network models. The conducted study revealed some interesting insights into how different network structures influence the properties in question.

Results for clustering coefficient and average shortest path confirmed the analytical results that are known for the three considered models [10]. Additionally, it is apparent that for scale-free networks the average shortest path does not depend on network density, whereas for small-world and random networks the average shortest path decreases as the density increases. The clustering coefficient achieves the highest value, regardless of the density, for the small-world network with the smallest probability of rewiring $p = 0.3$. This confirms the theoretical consideration as this network in its structure, is closest to regular lattice which has very high clustering.

For closeness centrality, especially for denser networks, there is not much difference across the network models. For sparse networks, closeness centrality tends to be smaller for small-world and random networks than scale-free networks. For scale-free networks, density does not influence closeness centrality, and it becomes stable. Whereas, for small-world and random networks the closeness centrality increases as the network gets denser. For betweenness centrality, there is a clear difference between scale-free networks and other models. Betweenness for scale-free networks is consistently smaller compared with other models. We attribute that to the ‘rich get richer’ phenomenon that causes a concentration of edges in hub nodes. This study offers a generic network simulator and a set of analyses which have revealed some interesting, and previously unknown characteristics of the networks. Our results indicate that only by looking at a wide spectrum of generated networks we can identify extraordinary phenomena (e.g. mean betweenness centrality differentiates a scale-free network from a random graph or small-world network) that can otherwise be overlooked. Such information is useful not only to identify the type of real-world networks but also to calculate properties that can not be derived analytically.

References

- [1] Barabási, A.L., Albert, R., 1999. Emergence of scaling in random networks. *science* 286, 509–512.
- [2] Csardi, G., Nepusz, T., 2006. The igraph software package for complex network research. *InterJournal, Complex Systems* 1695, 1–9.
- [3] Erdős, P., Rényi, A., 1959. On random graphs i. *Publicationes Mathematicae (Debrecen)* 6, 290–297.
- [4] Erdős, P., Rényi, A., 1960. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci* 5, 43.
- [5] Erdős, P., Rényi, A., 1961. On the strength of connectedness of a random graph. *Acta Mathematica Hungarica* 12, 261–267.
- [6] Gleich, D., 2007. *Matlab bgl v2*. .
- [7] Leskovec, J., Sosič, R., 2016. Snap: A general-purpose network analysis and graph-mining library. *ACM Transactions on Intelligent Systems and Technology (TIST)* 8, 1.
- [8] Medina, A., Lakhina, A., Matta, I., Byers, J., 2001. Brite: An approach to universal topology generation, in: *Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2001. Proceedings. Ninth International Symposium on, IEEE. pp. 346–353.
- [9] Mitzenmacher, M., 2004. A brief history of generative models for power law and lognormal distributions. *Internet mathematics* 1, 226–251.
- [10] Newman, M., . *Networks: an introduction*. 2010. United States: Oxford University Press Inc., New York , 1–2.
- [11] Newman, M.E., 2005. Power laws, pareto distributions and zipf’s law. *Contemporary physics* 46, 323–351.
- [12] Newman, M.E., Watts, D.J., 1999. Scaling and percolation in the small-world network model. *Physical Review E* 60, 7332.
- [13] Price, D.d.S., 1976. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American society for Information science* 27, 292–306.
- [14] Rubinov, M., Sporns, O., 2010. Complex network measures of brain connectivity: uses and interpretations. *Neuroimage* 52, 1059–1069.
- [15] Siek, J.G., Lee, L.Q., Lumsdaine, A., 2001. *The Boost Graph Library: User Guide and Reference Manual, Portable Documents*. Pearson Education.
- [16] Simon, H.A., 1955. On a class of skew distribution functions. *Biometrika* 42, 425–440.
- [17] Solomonoff, R., Rapoport, A., 1951. Connectivity of random nets. *The bulletin of mathematical biophysics* 13, 107–117.
- [18] Watts, D.J., Strogatz, S.H., 1998. Collective dynamics of small-worldnetworks. *nature* 393, 440–442.
- [19] Wickham, H., 2011. *ggplot2*. Wiley Interdisciplinary Reviews: Computational Statistics 3, 180–185.