Comparison of Graph Generation Models focusing on Accuracy and Variation

Mei Fukuda
Tokyo Institute of Technology

Kazuki Nakajima
Tokyo Institute of Technology

Kazuyuki Shudo
Tokyo Institute of Technology

ABSTRACT

Generation models of graphs have been used to compare and analyze the properties of graph structures and to produce graphs that resemble real-world networks. When using a generation model to mimic a real-world network, it is desirable for the error in the properties between the target graph and the generated graph and the variation of the errors between generated graphs are small. However, since many existing generation models generate graphs by adding edges at random, the extent of the error and its variation for each generated graph is unclear.

This paper studies the error and the variation of properties of graphs generated using the dK-series framework, which has been proposed to analyze the topology of a network based on the degree of nodes. In addition, we propose a new graph generation model that takes the degree distribution and degree-dependent clustering coefficient as inputs. We show that the proposed model is able to reduce the error to a greater extent than other generation models.

KEYWORDS

network analysis, graph generation models, estimation, sampling, social networks

ACM Reference Format:
Mei Fukuda, Kazuki Nakajima, and Kazuyuki Shudo. 2020. Comparison of Graph Generation Models focusing on Accuracy and Variation. In *Proceedings of ACM Conference (MLG '20)*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Graph generation models have historically been used to compare and analyze the properties of graph structures and to generate graphs that resemble real-world networks. For example, by generating and comparing random graphs with the same number of nodes and edges as the target graph, we can observe the properties of the graphs, such as high-clustered and small-world properties [33]. Additionally, sufficient real-world graph data are difficult to obtain, though researchers want these data, which is a motivation for generating many graphs that resemble real-world networks. Furthermore, while many attempts have been made to use sampling to estimate the overall properties for networks where the overall topology is unknown, such as large graph structures of user relationships in online social networking services (OSNs) [1, 12, 18, 19], the properties that can be estimated by sampling are limited to those that can be computed from local information such as the number of nodes, degree distributions, and clustering coefficients. Therefore, graph generation models have been used to generate graphs that mimic even properties that are difficult to estimate.

Priya et al. proposed a framework for random graph generation models called dK-series, which characterizes the properties of a graph using a series of probability distributions specifying all the degree correlations within d-sized subgraphs of a given graph [23]. Consider the following scenario. Suppose you want to understand the structure of a real network by comparing it to a graph generated by fixing a certain property. In this case, the feature of the generated graph when you hierarchically add properties to be fixed is also of great interest. Additionally, there is another motivation for generating graphs with properties that we know or estimate via sampling when a particular property of graphs is available or the overall data are not available. The dK-series provides a systematic description for such various graph generation models we use depending on a purpose. Gjoka et al. proposed a generation model for a 2.5K graph based on dK-series [15]. The model generates a graph with a fixed joint degree distribution and degree-dependent clustering coefficients. Moreover, the graphs generated by this model can accurately mimic real-world networks.

Graphs generated by a generation model based on dK-series can estimate globally defined properties with high accuracy, but due to the random addition of the edges, the noninput properties are different for each generation, even if the input is the same. Whether the generated graphs vary greatly from one generation to the next or whether they always fall within a certain range is unclear. If we want to generate a graph that resembles a certain graph, the fact that the generated graph is "similar" to the target graph has to be reliable, so the error in the properties between the target graph and the generated graph, as well as its variation, should be small. However, so far, the error and variation of the properties of the generated graphs for the same input have not been thoroughly studied. Furthermore, while dK-series can mimic almost all properties at 3K graphs, except those fixed at the input, the generation of 3K graphs entails enormous computational complexity: 2.5K has been able to

MLG '20, August, 2020, San Diego, CA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

reduce the complexity to a practically usable level. However, even at 2.5K, there are concerns about scalability, and the generation of large-scale graphs with hundreds of millions of nodes and edges remains challenging.

In this study, we generate a large number of graphs using generation models based on dK-series for a real network dataset and compare the errors, variation, and execution time of each property of the generated graphs. We also propose a new generation model (1K+) that overcomes the scalability of graph generation while maintaining the accuracy. The structure of the rest of this paper is as follows. Chapter 2 provides the notation and the definitions of the properties to be discussed. Chapter 3 presents related work. Chapter 4 discusses the errors and the variations of generation models based on dK-series. Chapter 5 proposes the new model and discusses its accuracy and variation. Finally, Chapter 6 summarizes this paper.

2 NOTATION

We consider an undirected, unweighted and static graph $G = (V, E)$, with $|V| = n$ nodes and $|E| = m$ edges. We call a node $v \in V_k$ a "degree k node", where V_k is the set of all nodes of degree k. Let the set of nodes $V = \{v_1 \cdots v_n\}$; the degree of v_i is denoted as d_i. $N(v) \subset V$ is the set of neighbors of v.

2.1 Properties of Interest

The degree distribution is represented as a probability distribution $P(k) = P(d_i = k)$ for $i \in \{1, \cdots, n\}$. In this paper, we use another property of degree, called the joint degree distribution (JDD) $P(k_1, k_2) = P(d_i = k_1, d_j = k_2)$ $(i \neq j)$, which represents the distribution of a pair of degrees connected as edges.

The clustering coefficient c_i of a node v_i captures the extent to which the neighbors of a node form triangles and is typically defined as the ratio of the number of edges between neighbors divided by the maximum number of such edges. Thus, the clustering coefficient c_i is represented as $c_i = \frac{2T_i}{d_i(di-1)}$, where T_i is the number of triangles including v_i. The degree-dependent clustering coefficient $\bar{c}(k)$ is defined as $\bar{c}(k) = \frac{1}{|V_k|}\sum_{v_i \in V_k} c_i$. Additionally, the average clustering coefficient $\bar{c}$, c_i averaged over all nodes in G, is defined as $\bar{c} = \frac{1}{n}\sum_{v_i} c_i$, and $\bar{c}(k)$ determines $\bar{c}$ because $\bar{c} = \frac{1}{n}\sum_k |V_k| \cdot \bar{c}(k)$.

The shortest path distribution is defined as the probability of a distance between arbitrary pairs of nodes, where the distance between nodes v_i and v_j, $dist(v_i, v_j)$, in unweighted graphs is denoted as the minimum number of edges of paths connecting nodes v_i and v_j. The maximal clique distribution is defined as the frequency of the size of maximal cliques, where a clique is a complete subgraph and its size is the number of nodes in the subgraph. Furthermore, the cycles distribution is defined as the frequency of the cycle length for a minimal cycle basis, in which a cycle cannot be reconstructed by the union of cycles in the base. The spectrum is the eigenvalues of a graph: we use the distribution of the 20 largest eigenvalues. The closeness centrality of a node v_i is defined as the multiplicative inverse of $\sum_{v_j \in N(v_i)} dist(v_i, v_j)$, the sum of distances of the node to all other nodes. The greater the closeness centrality of a node is, the closer it is to all other nodes.

3 RELATED WORK

The Erdős-Rényi model is a random graph generation model, also known as the ER model, that was introduced in the late 1950s [11]. There are two types of ER models, the $G(n, m)$ model and the $G(n, p)$ model. The $G(n, m)$ model involves uniform random selection among a set of graphs with n nodes and m edges. In contrast, the $G(n, p)$ model independently constructs an edge for each pair of vertices with probability p; that is, the expected number of edges in the $G(n, p)$ model is $\binom{n}{2}p$. The main motivation for generating random graphs with the ER model is to verify what properties a graph with the same number of nodes n and edges m generally satisfies, which is not a sufficient model to mimic a real network. Since the $G(n, m)$ model does not add edges with probability independent from that of other edges, the $G(n, p)$ model, which usually considers p as a function of n and considers the structure in $n \to \infty$, is commonly used.

The model for generating a random graph with a fixed degree distribution is called the configuration model. The configuration model has been studied since the 1970s and has been used in a number of research projects [3–7, 21, 22, 24, 25]. Here, we apply the model using the approach by Newman, Strogatz, and Watts [9, 10]. Suppose that the degree sequence $d_i \cdots d_n$ is independent and satisfies $P(d_i = k) = \frac{|V_k|}{n}$. This model takes the number of nodes n and the degree distribution $P(d_i = k)$ as inputs and, assuming that there are d_i half-edges from node v_i, randomly selects two half-edges to construct a connecting edge. This method may generate multiple edges and self-loops. An algorithm to avoid multiple edges and self-loops has also been proposed [17].

Priya et al. proposed dK-series [23], a framework for systematically characterizing a graph using a probability distribution $P(k_1, k_2, \cdots, k_d)$ that identifies all degree correlations in a connected subgraph consisting of d nodes. The 0K graph fixes only the average degree $\bar{k}$ of the graph G; that is, the $G(n, m)$ model in the Erdős-Rényi model described above is equivalent. The 1K graph, given the degree distribution $P(k)$, can be generated by the configuration model described above. The 2K graph is generated by identifying and fixing the joint degree distribution (JDD) $P(k_1, k_2)$, which can be accomplished by extending the configuration model [23]. A model for generating 2K graphs that avoids multiple edges and self-loops has been proposed by Isabelle et al. [29, 30].

The 3K graph fixes the distributions $P_\triangle(k_1, k_2, k_3)$ and $P_\wedge(k_1, k_2, k_3)$ of 3-node subgraphs formed as triangles and wedges. As d increases, the size of the subgraphs increases. An nK graph defined using the number of nodes n is identical to graph G of the original network.

Priya et al. use a method called dK-randomizing rewiring to randomly swap edges while preserving the existing form of the dK distribution and compare the result to the properties in the original graph. It has been shown that $d = 2$ reproduces most of the properties, and $d = 3$ reproduces all the properties almost faithfully, while the computational complexity of the dK graph generation increases rapidly as d increases [23]. Moreover, generating a graph that satisfies the input probability distribution is difficult at 3K and above [23].

Gjoka et al. proposed an idea to reproduce properties that are difficult to estimate by generating a graph with fixed properties that can be estimated with high accuracy by sampling a large-scale graph structure, such as a social network, where the overall topology is unknown.

Following the dK-series, Gjoka et al. presented a model to generate 2.25K graphs with a fixed average clustering coefficient $\bar{c}$ in addition to JDD and 2.5K graphs with a fixed degree-dependent clustering coefficient $\bar{c}(k)$ and JDD and compared the results with 2K graphs with respect to accuracy. They also proposed a method to estimate the JDD and degree-dependent clustering coefficients with high accuracy [15, 31].

A 2.5K graph is generated by two steps: 1) assigning a number of positions to all nodes in the algorithm to generate the 2K graph and sorting all pairs of nodes by their distance to produce a graph with many triangles, and 2) reducing the clustering coefficients by swapping two nodes with the same degree while still satisfying the JDD.

The 2.5K graph is similar to the target graph for a wide range of major properties other than the maximal clique distribution. However, due to its nature as a random graph, different graphs are generated for each generation, and there is no mention of how much variation exists and how reliable the graphs are once generated.

Orsini et al. studied the randomness of graphs generated by dK-series models [27]. They showed that many important properties of real networks are closely reproduced by dk-random graphs whose degree distributions and clustering coefficients are the same as in the corresponding real network. They considered some, but limited, real networks. Specifically, social network datasets should also be considered. Additionally, various other properties must be verified, and more graphs must be produced to verify the accuracy.

Furthermore, many studies on graph generation models that fix degree-dependent clustering coefficients have been reported. Serrano et al. [28] proposed a model for generating random graphs satisfying the degree-dependent clustering coefficient $\bar{c}(k)$ and degree distribution $P(d_i = k)$, but this method is not practical for graphs that have large clustering coefficients. According to Wang [32], graph generation models satisfying a given clustering coefficient has also been proposed by Bansal et al [2], Newman [26], and Gleeson [16]. Bansal et al. proposed a model that satisfies the average clustering coefficient $\bar{c}$, and Newman proposed a model that fixes c_i for each node unless one edge is used to generate multiple triangles. Gleeson proposed an improved model that focuses on the constraints of Newman's generation model. Except for Bansal et al.'s generation model, these approaches cannot satisfy the degree distribution at the same time. Therefore, limited graph generation models are available for fixing degree-dependent clustering coefficients and degree distributions.

Table 1: Dataset: Each number is of the largest connected component of each graph.

Dataset	$\lvert V\rvert$	$\lvert E\rvert$	$\bar{d}$	$\bar{c}$
Caltech [14]	769	16 656	21.65	0.409
Rice [14]	4 087	184 828	45.22	0.294
wiki-Vote [20]	7 115	100 762	14.16	0.141

Table 2: Environment.

OS	macOS Mojave 10.14.5
Processor	2.7 GHz Intel® Core™ i7
Memory	16 GB 2133 MHz LPDDR3
Language	Python 2.7.10 (for 2K & 2.5K graphs) Python 3.6.3 (for others)

4 ERROR AND ITS VARIATION FOR GENERATED GRAPHS

We generate a number of random graphs with generation models based on dK-series using properties computed from real network datasets as input and compare the various properties of the generated graphs with the real values in the datasets.

4.1 Preparation

In this experiment, we use three datasets with the various topologies shown in Table 1. The entire dataset is treated as an undirected graph, and its largest connected component (LCC) is used in the experiment.

The 0K graphs are generated with `gnm_random_graph` [8] in NetworkX, a Python library for network graph analysis. The 1K graphs are generated with the configuration model, and 2K graphs and 2.5K graphs are generated with `construct_simple_2K`, `construct_triangles_2K` and `mcmc_improved_2_5_K` of the software published by Gjoka et al. [13], respectively.

We measure the difference between two discrete distributions using the normalized mean absolute error (NMAE) defined below.

$$\mathrm{NMAE}(\widehat{\vec{x}}, \vec{x}) = \frac{\sum(|\widehat{x}_i - x_i|)}{\sum x_i}$$

$\widehat{\vec{x}}$ and $\vec{x}$ are vectors that correspond to the real and estimated discrete distributions.

Additionally, this experiment is conducted under the environment described in Table 2.

Table 3: Graph generation time in seconds.

Dataset	0K	1K	1K+	2K	2.5K
Caltech [14]	0.0658	1.17	1.13	0.532	11.3
Rice [14]	0.629	64.0	53.1	5.09	84.7
wiki-Vote [20]	0.382	47.2	42.5	3.29	62.7

4.2 Measurement

For each dataset, 1) we calculate the number of nodes, number of edges, degree distribution, JDD, and degree-dependent clustering coefficients, 2) generate 100 graphs with each random graph generation model using the corresponding properties, and 3) for their generated graphs, calculate the NMAE of the following graph properties.

- Shortest path distribution
- Maximal clique distribution
- Cycles distribution
- Spectrum (20 largest eigenvalues)
- Closeness centrality

4.2.1 Property Analysis. The distributions of the above five properties for the Caltech dataset and its generated graphs are shown in Figure 1.

The red line in each figure plots the distribution of the target graphs, while the other plots the distribution of the generated graphs. As d of dK increases, some properties become closer to the distribution of the red line.

4.2.2 Error Distribution. For each of the graphs generated from the three datasets Caltech, Rice, and wiki-Vote, the error distributions of the above five properties are shown in Figure 2.

For all properties, the error variation is small regardless of the generation model: 0K is very small for all properties.

For each dK, the variation of the shortest path distribution, spectrum, and closeness centrality is stably small, and the error tends to decrease as d increases.

The errors and their variations of the maximum clique and cycle distributions are larger than those of the other properties, and the distributions vary depending on the topology of the data set.

4.2.3 Performance. The execution time to generate one graph for each generation model is shown in Table 3. Ten graphs were generated, and the average is shown. The 0K and 1K graphs were run with Python 3.6.3, and 2K and 2.5K were run with Python 2.7.10.

The 0K graphs can be generated the fastest, and the generation of 2.5K graphs is the most time consuming.

5 1K+ GRAPH GENERATION MODEL

We propose a generation model that fixes the degree distribution and makes the degree-dependent clustering coefficients closer to the input than that of the 1K graph. In this generation model, because graphs generated by conventional models tend not to have large size cliques compared to real graphs, as shown in Figure 1, we focus on cliques and generate graphs by two steps: 1) generating cliques according to the number of triangles for degree k, and 2) matching the degree of each node using the conventional configuration model. We call the graph generated by this generation model a 1K+ graph.

In the proposed generation model, for each degree, we generate cliques that can form the closest number of triangles to the target number of triangles among nodes with the same degree.

Algorithm 1 illustrates the implementation of 1K+. Let the input be $ntri[k]$, which is the sum of the number of triangles joined by the degree k nodes, computed from the degree distribution $P(d_i = k)$ and the degree-dependent clustering coefficient $\bar{c}(k)$.

The graphs generated by this 1K+ algorithm can ensure that the degree-dependent clustering coefficients are closer to the target graph than the graphs generated by the 0K, 1K, and 2K graph generation models. Figure 3 shows the distribution of degree-dependent clustering coefficients of the generated graphs. The red dots in the figure plot the distribution of the target graphs, and the others are the distribution of the 100 generated graphs. The distribution of the degree-dependent clustering coefficients of the 1K+ graphs overlaps well with the red dots compared to the graphs generated by the 0K, 1K, and 2K approaches.

The experiment discussed in Chapter 4 is repeated for the 1K+ generation model. Using the degree distribution and the distribution of the degree-dependent clustering coefficient as inputs, we generate 100 graphs with the 1K+ model in Python 3.6.3, as described in Algorithm 1.

The distribution of properties for the Caltech dataset is shown in Figure 1. As shown in Figure 2, the graphs generated by the proposed 1K+ algorithm have smaller errors than the 2K graphs; for example, with respect to the shortest path distribution, 1K+ has the smallest error in the Caltech graph and the second smallest error, after 2.5K, in the Rice graph. Additionally, the errors of the maximal clique and cycle distributions in the 1K+ graph are the smallest, on average, in the Wiki-Vote graph, with some improvement from 1K in Caltech and Rice. Furthermore, as can be seen from Table 3, although the 1K+ graph requires more inputs than the 1K graph, it can be generated in a shorter time.

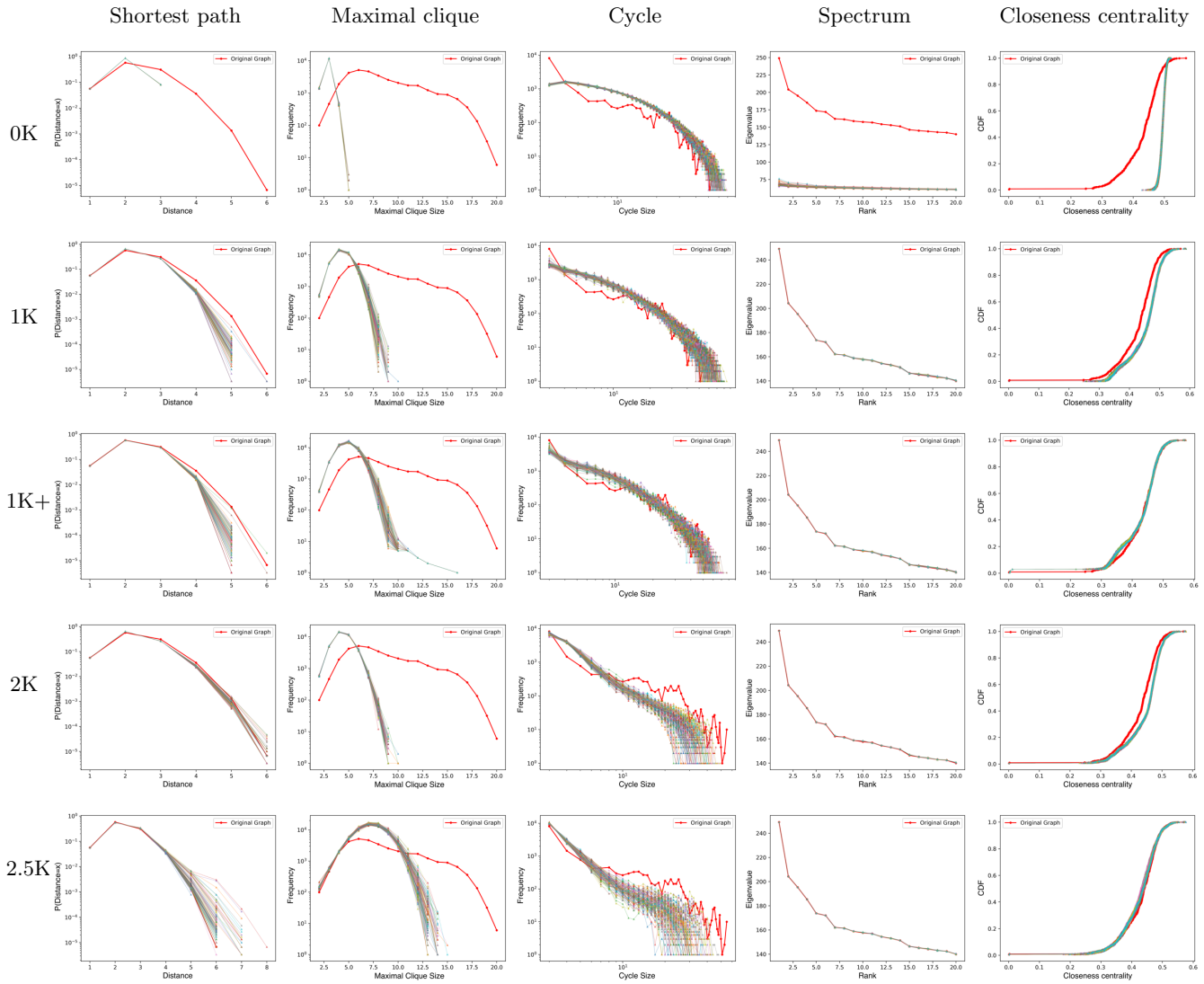
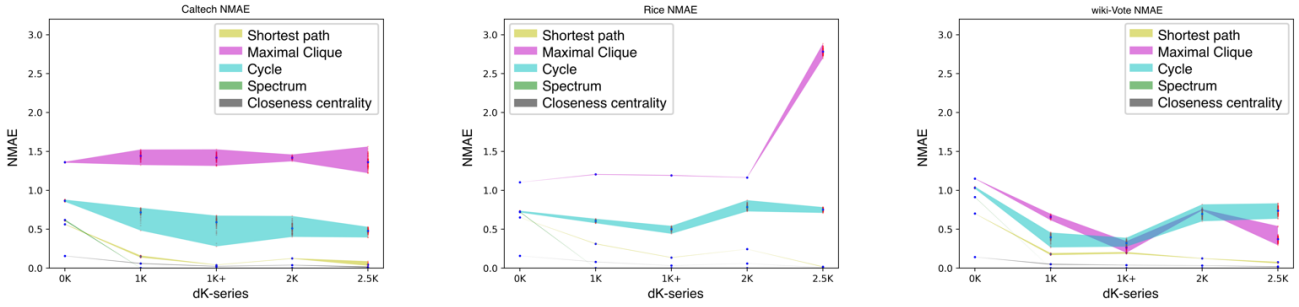


Figure 1: Distribution of properties (Caltech).



(a) Caltech **(b) Rice** **(c) wiki-Vote**

Figure 2: Error distribution.

Algorithm 1 Generate 1K+ graph.

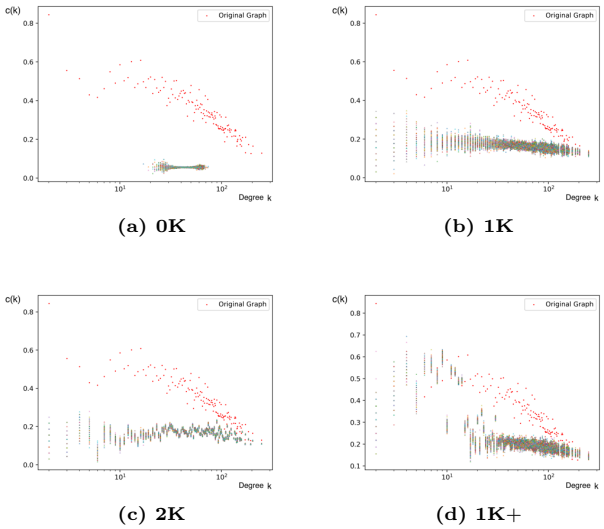
Require: $d_1 \cdots d_n$: degree sequence
  $ntri[k]$: the sum of the number of triangles joined by node $v \in V_k$
Ensure: 1K+ graph
  $V' \leftarrow \{1, \cdots n\}$
  $E' \leftarrow$ empty set
  $V'_k \leftarrow$ degree k node v_i
  for all degree k **do**
    if $|V'_k| < k+1$ **then**
      if $ntri[k] <$ the number of triangles can be generated by the size $|V'_k|$ clique **then**
        generate clique with nodes $v \in V'_k$ whose size can generate the nearest number of triangles to $ntri[k]$
        update E'
      else
        generate size $|V'_k|$ clique with nodes $v \in V'_k$
        update E'
      end if
    else
      if $|V'_k| \geq$ the number of triangles can be generated by the size k clique **then**
        generate size $k+1$ clique with nodes $v \in V'_k$
        update E'
      else
        generate clique with nodes $v \in V'_k$ whose size can generate the nearest number of triangles to $ntri[k]$
        update E'
      end if
    end if
  end for
  match the degree with configuration model
  $G' \leftarrow (V', E')$
  return G'

6 CONCLUSION AND FUTURE WORK

In this study, we examined the variation of properties of graphs generated by the graph generation model based on dK-series. We also considered the distribution of errors: the error variation is small for properties not given as inputs in the generation models.

Furthermore, we proposed a model for generating a 1K+ graph that focuses on cliques and constructs triangles using degree-dependent clustering coefficients as input. We showed that this model is faster than 1K and 2.5K graph generation and, depending on the topology, can reduce the error of the properties compared to those of the 2K graph.

The future task is to improve the 1K+ model by making its degree-dependent clustering coefficients closer to the inputs and to stably reduce the errors in the properties, such as the maximum clique distribution.



(a) 0K (b) 1K

(c) 2K (d) 1K+

Figure 3: Distribution of degree-dependent clustering coefficient (Caltech).

ACKNOWLEDGMENTS

This work was supported by New Energy and Industrial Technology Development Organization (NEDO).

REFERENCES

[1] Yong-Yeol Ahn, Seungyeop Han, Haewoon Kwak, Sue Moon, and Hawoong Jeong. 2007. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th international conference on World Wide Web*. 835–844.
[2] Shweta Bansal, Shashank Khandelwal, and Lauren Ancel Meyers. 2009. Exploring biological network structure with clustered random networks. *BMC Bioinformatics* 10, 1 (2009), 405.
[3] Edward A. Bender and E. Rodney Canfield. 1978. The Asymptotic Number of Labeled Graphs with Given Degree Sequences. *Journal of Combinatorial Theory, Series A* 24, 3 (1978), 296–307.
[4] Béla Bollobás. 1980. A Probabilistic Proof of an Asymptotic Formula for the Number of Labelled Regular Graphs. *European Journal of Combinatorics* 1, 4 (1980), 311–316.
[5] Nicholas C. Wormald. 1981. The asymptotic connectivity of labelled regular graphs. *Journal of Combinatorial Theory, Series B* 31, 2 (1981), 156–167.
[6] Fan Chung and Linyuan Lu. 2002. Connected Components in Random Graphs with Given Expected Degree Sequences. *Annals of Combinatorics* 6, 2 (2002), 125–145.
[7] Fan Chung and Linyuan Lu. 2004. The Average Distance in a Random Graph with Given Expected Degree. *Internet Mathematics* 1, 1 (2004), 91–113.
[8] NetworkX 1.10 documentation. 2015. gnm_random_graph. Retrieved February 12, 2020 from https://networkx.github.io/documentation/networkx-1.10/reference/generated/networkx.generators.random_graphs.gnm_random_graph.html
[9] Rick Durrett. 2006. *Random Graph Dynamics*.
[10] Mark E.J. Newman, Steven H. Strogatz, and Duncan J Watts. 2001. Random Graphs with Arbitrary Degree Distributions and their Applications. *Physical review. E* 64, 2 (2001), 026118.
[11] P. Erdős and A. Rényi. 1959. On Random Graphs I. *Publicationes Mathematicae Debrecen* 6 (1959), 290.
[12] Emilio Ferrara. 2014. *Measurement and Analysis of Online Social Networks Systems*. 891–893.
[13] Minas Gjoka. 2013. 2.5K-Graphs Software. Retrieved February 12, 2020 from http://www.minasgjoka.com/2.5K/instructions/

[14] Minas Gjoka. 2018. Index of /2.5K/graphs. Retrieved February 12, 2020 from http://www.minasgjoka.com/2.5K/graphs/
[15] Minas Gjoka, Maciej Kurant, and Athina Markopoulou. 2013. 2.5 k-graphs: from sampling to generation. In *2013 Proceedings IEEE INFOCOM*. 1968–1976.
[16] James P. Gleeson. 2009. Bond percolation on a class of clustered random networks. *Physical review. E* 80, 3 (2009), 036107.
[17] Charo I Del Genio, Hyunju Kim, Zoltan Toroczkai, and Kevin Bassler. 2010. Efficient and Exact Sampling of Simple Graphs with Given Arbitrary Degree Sequence. *PloS one* 5, 4 (2010), e10012.
[18] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue B. Moon. 2010. What is Twitter, a social network or a news media?. In *Proceedings of the 19th international conference on World Wide Web*. 591–600.
[19] Jure Leskovec and Christos Faloutsos. 2006. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 631–636.
[20] Jure Leskovec and Andrej Krevl. 2014. SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data.
[21] T. Luczak. 1989. Sparse random graphs with a given degree sequence. In *Proceedings of the Symposium on Random Graphs, Poznan*. 165–182.
[22] Junling Ma, P. van den Driessche, and Frederick H. Willeboordse. 2013. Effective degree household network disease model. *Journal of Mathematical Biology* 66, 1 (2013), 75–94.
[23] Priya Mahadevan, Dmitri Krioukov, Kevin Fall, and Amin Vahdat. 2006. Systematic topology analysis and generation using degree correlations. *ACM SIGCOMM Computer Communication Review* 36, 4 (2006), 135–146.
[24] Michael Molloy and Bruce A. Reed. 1995. A Critical Point for Random Graphs with a Given Degree Sequence. *Random structures & algorithms* 6, 2-3 (1995), 161–180.
[25] Michael Molloy and Bruce A. Reed. 1998. The Size of the Giant Component of a Random Graph with a Given Degree Sequence. *Combinatorics, Probability & Computing* 7, 3 (1998), 295–305.
[26] Mark EJ Newman. 2009. Random Graphs with Clustering. *Physical review letters* 103 (2009), 058701.
[27] Chiara Orsini, Marija Mitrović Dankulov, Pol Colomer de Simon, Almerima Jamakovic, Priya Mahadevan, Amin Vahdat, Kevin E. Bassler, Zoltán Toroczkai, Marián Boguñá, Guido Caldarelli, Santo Fortunato, and Dmitri V. Krioukov. 2015. Quantifying randomness in real networks. *Nature communications* 6, 1 (2015), 1–10.
[28] M Angeles Serrano and Marián Boguná. 2005. Tuning clustering in random networks with arbitrary degree distributions. *Physical review. E* 72 (2005), 036133.
[29] Isabelle Stanton and Ali Pinar. 2011. Sampling Graphs with a Prescribed Joint Degree Distribution Using Markov Chains. In *Proceedings of the 13th Workshop on Algorithm Engineering and Experiments*. 151–163.
[30] Isabelle Stanton and Ali Pinar. 2012. Constructing and Sampling Graphs with a Prescribed Joint Degree Distribution. *Journal of Experimental Algorithmics* 17 (2012), 3–1.
[31] Bálint Tillman, Athina Markopoulou, Minas Gjoka, and Carter T Buttsc. 2019. 2K+ Graph Construction Framework: Targeting Joint Degree Matrix and Beyond. *IEEE/ACM Transactions on Networking* 27, 2 (2019), 591–606.
[32] Cheng Wang, Omar Lizardo, and David Hachen. 2014. Algorithms for generating large-scale clustered random graphs. *Network Science* 2, 3 (2014), 403–415.
[33] Duncan J. Watts and Steven H. Strogatz. 1998. Collective dynamics of 'small-world' networks. *Nature* 393, 6684 (1998), 440–442.