Quantification and Comparison of Degree Distributions in Complex Networks

Sadegh Aliakbary
Sharif University of Technology
Tehran, Iran
aliakbary@ce.sharif.edu

Jafar Habibi
Sharif University of Technology
Tehran, Iran
jhabibi@sharif.edu

Ali Movaghar
Sharif University of Technology
Tehran, Iran
movaghar@sharif.edu

***Abstract*—The degree distribution is an important characteristic in complex networks. In many applications, quantification of degree distribution in the form of a fixed-length feature vector is a necessary step. Additionally, we often need to compare the degree distribution of two given networks and extract the amount of similarity between the two distributions. In this paper, we propose a novel method for quantification of the degree distributions in complex networks. Based on this quantification method, a new distance function is also proposed for degree distributions, which captures the differences in the overall structure of the two given distributions. The proposed method is able to effectively compare networks even with different scales, and outperforms the state of the art methods considerably, with respect to the accuracy of the distance function.**
The datasets and more detailed evaluations are available upon request.

***Index Terms*—Social Network, Complex Network, Degree Distribution, Feature Extraction, Distance Function, Power-law, Kolmogorov-Smirnov Test.**

I. Introduction

Real-world networks, such as social networks and communication networks, display common topological features that discriminate them from random graphs. Networks with such non-trivial properties are often called complex networks. Among the features, small path lengths (small-world property), high clustering, community structure and heavy-tailed degree distribution are well studied in the literature. Although the degree distribution is an important network characteristic, its quantification (feature extraction) is not a trivial task. Representing the network as a fixed-size feature vector is an important step in every data analysis process [1], [2]. In order to employ the degree distribution in such applications, a procedure is needed for extracting a feature vector from the degree distribution. Additionally, the quantified feature vector is useful in developing a distance function for comparing two degree distributions because we often need to compare complex networks according to their degree distribution. For example, in evaluation of sampling algorithms, we usually compare the given network instance with its sampled counterpart to ensure that the structure of the degree distribution is preserved [3]. Hence, quantification and comparison of degree distributions is an important research problem with many applications.

Currently, there exist three main approaches for comparing degree distributions in the literature: Kolmogorov-Smirnov (KS) test [4], [5], comparison based on fitted power-law exponent [6], [7], and comparison based on distribution percentiles [2]. The "power-law exponent" approach is based on the assumption that the degree distributions obeys the power-law model. This assumption is invalid for many complex networks [8], [9]. KS-test is based on a point-to-point comparison of the distributions, which is not a good approach for comparing networks with different ranges of node degrees. Percentile method is also too sensitive to the outlier values of node degrees. As a result, the existing methods are actually inappropriate for comparing the degree distribution of networks, particularly when the target networks have different sizes and scales. We propose to consider the mean and standard deviation of the degree distribution in the quantification phase in order to make the comparison process more accurate and more robust, particularly with respect to scale variation. In our proposed "quantification process", a feature vector of real numbers is extracted from the degree distribution, which can be used in data analysis applications, data-mining algorithms and comparison of degree distributions. In "comparison task", we define a distance function that computes the distance (amount of dissimilarity) between two given network degree distributions. The proposed method offers an effective quantified representation of the degree distributions and outperforms the baseline methods with respect to the accuracy of the distance function. Although our proposed approach is applicable to other network types, in this paper we focus on simple undirected networks.

The rest of this paper is organized as follows: In section II, we briefly overview the related works. In section III, we propose a new method for degree distribution quantification and comparison. In section IV, we evaluate the proposed method and we compare it with baseline methods. Finally, we conclude the paper in section V.

II. Related Works

The degree distribution of many real-world networks follow the power-law model [5]. In power-law degree distribution the number of nodes with degree d is proportional to $d^{-\gamma}$ ($N_d \propto d^{-\gamma}$) where γ is a positive number called "the power-law exponent". The exponents of the fitted power-law

can be used to characterize graphs [7]. A possible approach for quantifying the degree distribution is to fit a power-law distribution on the network distribution and to find its power-law exponent (γ). Although we can compare networks according to their fitted power-law exponents, the power-law exponent is too limited to represent a whole degree distribution. This approach also follows the assumption that the degree distribution shows a power-law model, which is not always valid, because many networks follow other degree distribution models such as log-normal distribution [8], [9]. In addition, the power-law exponent does not reflect the deviation of the degree distribution from the fitted power-law distribution. As a result, two completely different distributions may have similar quantified value for fitted power-law exponent.

Degree distribution is a kind of probability distribution and there are a variety of measures for calculating the distance between two probability distributions. In this context, the most common method is the Kolmogorov-Smirnov (KS) test, which is equal to the maximum distance between the cumulative distribution functions (CDF) of the two probability distributions [4]. KS-test is frequently used for comparing two degree distributions [3]. The KS distance of two distributions is calculated according to Equation 1, in which $S_1(d)$ and $S_2(d)$ are the CDFs of the two degree distributions, and d indicates the node degree. KS-test is a method for comparing the degree distributions and calculating their distance, and it does not provide a quantification or feature extraction mechanism. Hence, we should maintain the CDF of the degree distributions so that we can compare them according to KS-test. KS-test is also sensitive to the scale and size of the networks, since it performs a point-to-point comparison of CDFs. KS-test does not provide a quantification mechanism and hence can not help in feature extraction and feature-based data analysis tasks. However, we include this method in the baseline methods as a distance function for degree distributions.

$$distance_{KS}(S_1, S_2) = \max_d |S_1(d) - S_2(d)| \tag{1}$$

Janssen et. al., [2] propose another method for quantification of degree distributions. In this method, the degree distribution is divided into eight equal-sized regions and the sum of degree probabilities in each region is extracted as distribution percentiles. This method is sensitive to the range of node degrees and also to outlier values of degrees. We recall this technique as "Percentiles" and we include it in baseline methods, along with "KS-test" and "Power-law" (the power-law exponent) to evaluate our proposed method. The proposed method is called "Degree Distribution Quantification and Comparison (DDQC)".

III. Proposed Method

We propose a new method for quantifying and comparing the degree distribution of networks. In this method, a vector of at least four real numbers is extracted from the degree distribution. A distance function is also suggested for comparing the quantified vectors. An appropriate distance metric for degree distributions should be able to effectively compare networks, even if they have different range of node degrees. To eliminate impact of the network size from the quantification of its degree distribution, we consider the mean and standard deviation of the degree distribution in the quantification procedure. The following two subsections show our proposed method for quantification and comparison of degree distributions.

A. Quantification of Degree Distribution

The degree distribution of a network is described in Equation 2 as a probability distribution function. In this equation, $D(v)$ shows the degree (number of connected edges) of node v. The aim of "quantification" task is to extract a fixed-length vector of real numbers as the representative of the degree distribution. We can use the quantified feature vector in network analysis and network comparison tasks. In the first step of quantification, we define four regions (R_G) in the degree distribution of a given network according to Equation 3. In this equation, $min(D(v))$ shows the minimum of all the existing degrees in the degree distribution, μ_G is the mean of degrees according to their probabilities, σ_G is the standard deviation of the degrees, α is a configurable parameter (it specifies the width of the regions), and $max(D(v))$ is the maximum existing node degree in the network. The smallest possible feature vector in our proposed method is a vector of four numbers, each of which showing the sum of the probability of degrees in one of the four specified regions. For finer comparison of distributions, we can further divide each region into L equal-size intervals. In our experiments, L is set to 2^β, where β is a positive integer value ($\beta \geqslant 0$) and the second configurable parameter of our method. Larger values of β results in a more fine-tuned quantification and also more elements in the feature vector. While even small values for β (e.g., $\beta = 1$) brings a more accurate distance metric compared to the baseline methods, larger values of β improves the accuracy of the distance function. Therefore, tuning β parameter is a tradeoff between the accuracy of the algorithm and the size of the quantified feature vector.

Equation 4 shows the length of each region ($|R_G(r)|$), which is equal to the absolute difference between the region end-points. Each region is then divided into $L = 2^\beta$ equal-length intervals. Equation 5 shows the interval points ($IP_G(b, L)$) and Equation 6 shows the defined ranges for intervals ($I_G(i, L)$), in which b is the interval point counter, $L = 2^\beta$ is the split factor of each region, and i is the interval identifier. The "interval degree probability (IDP_G)" is defined in Equation 7 as the sum of degree probabilities in a specified interval. Equation 8 shows the final quantified feature vector, which contains $4L = 4 \times 2^\beta$ elements, each of which is the IDP for one of the defined intervals. In this equation, $Q_\beta(G)$ shows the quantified representation of the degree distribution as a fixed-length feature vector with $4 \times 2^\beta$ elements.

$$P_G(d) = P(D(v) = d); v \in V(G) \tag{2}$$

$$R_G(r) = \begin{cases} [min_G(D(v)), \mu_G - \alpha\sigma_G] & \text{if } r = 1 \\ [\mu_G - \alpha\sigma_G, \mu_G] & \text{if } r = 2 \\ [\mu_G, \mu_G + \alpha\sigma_G] & \text{if } r = 3 \\ [\mu_G + \alpha\sigma_G, max_G(D(v))] & \text{if } r = 4. \end{cases} \quad (3)$$

$$|R_G(r)| = max(max(R_G(r)) - min(R_G(r)), 0) \quad (4)$$

$$IP_G(b, L) = \begin{cases} min_G(D(v)) + \frac{(b-1)\times|R_G(1)|}{L} & 1 \le b \le L \\ \mu_G - \alpha\sigma_G + \frac{(b-L-1)\times|R_G(2)|}{L} & L+1 \le b \le 2L \\ \mu_G + \frac{(b-2L-1)\times|R_G(3)|}{L} & 2L+1 \le b \le 3L \\ \mu_G + \alpha\sigma_G + \frac{(b-3L-1)\times|R_G(4)|}{L} & 3L+1 \le b \le 4L+1. \end{cases} \quad (5)$$

$$I_G(i, L) = [IP_G(i, L), IP_G(i+1, L)]\,; i = 1..4L \quad (6)$$

$$IDP_G(I) = P(min(I) \le D(v) < max(I)); v \in V(G) \quad (7)$$

$$Q_\beta(G) = \left\langle IDP_G(I_G(i, 2^\beta)) \right\rangle_{i=1..4\times2^\beta} \quad (8)$$

B. Comparison of Degree Distributions

Now, we can compare the degree distribution of two networks G_1 and G_2 according to their quantified feature vectors. We assume that the two degree distributions are quantified with the same configuration parameters of α and β. As a result, the size of the quantified vectors $Q_\beta(G)$ will be equal for the two networks. For small values of β, $Q_\beta(G)$ will show a coarse-grained representation of the degree distribution with few real numbers. For larger values of β, more fine-grained intervals of the degree distribution are available. According to Equation 9, we can simply compute the elements of $Q_\beta(G)$ based on the elements in $Q_{\beta+1}(G)$. In other words, it is possible to calculate IDP_G for smaller values of β (coarse-grained quantification) using IDP_G with larger values of β (fine-grained quantification).

Finally, we propose the Equation 10 for comparing two degree distributions. This equation compares two networks based on their corresponding IDP_G values for different granularities, from larger intervals (with $s = 0$) to smaller intervals (with $s = \beta$). A coefficient (γ^s), which is the third configurable parameter of our framework, is also included to influence the impact of different granularities. Intuitively, $d(G_1, G_2)$ compares the corresponding interval degree probabilities of the two networks, sums their differences, and also includes a discount factor of γ for the more fine-granularity intervals to raise the impact of course-grained intervals. Equation 10 is a distance function for degree distribution of networks, and it is the result of a comprehensive study of different artificial and real networks.

$$IDP_G(I_G(i, L) = IDP_G(I_G(2i-1, 2L) + IDP_G(I_G(2i, 2L) \quad (9)$$

$$d(G_1, G_2) = distance(G_1, G_2) = \sum_{s=0}^{\beta} \gamma^s \sum_{i=1}^{4\times2^s} |IDP_{G_1}(I_{G_1}(i, 2^s) - IDP_{G_2}(I_{G_2}(i, 2^s)| \quad (10)$$

IV. Evaluation

A. Datasets

In our problem setting, we aim a distance function that given the degree distribution of two networks, calculates how similar they are. But what benchmark is available for evaluating such a distance function? For evaluating different distance metrics, an approved dataset of networks with known distances of its instances is required. Although there is no such an accepted benchmark of networks with known "distance values", there exist some similarity witnesses among the networks. For evaluating different distance metrics, we have prepared two network datasets with admissible similarity witnesses among the networks of these datasets:

1) Real-world Networks

We have collected a dataset of 33 real-world networks of different types, most of them are publicly available in the web. The networks are selected from six different network categories. The category of networks is a sign of similarity: networks of the same type usually follow similar link formation procedures and produce similar degree distributions. Therefore, when comparing two network instances, we expect the distance metric to return small distances (in average) for networks of the same type and relatively larger distances for networks with different types. The networks of this dataset are categorized as following: ***Citation Networks***. In these networks, the edges show the citations between the papers or patents. The members of this class are: Cit-HepPh[1], Cit-HepTh[1], dblp_cite[2], and CitCiteSeerX[3]. ***Collaboration Networks***. This class shows the graph of collaboration or co-authorships. The members of this class are: CA-AstroPh[1], CA-CondMat[1], CA-HepTh[1], CiteSeerX_Collaboration[3], com-dblp.ungraph[1], dblp_collab[2], refined_dblp20080824[4], IMDB-USA-Commedy-09[5], CA-GrQc[1], and CA-HepPh[1]. ***Communication Networks***. These networks show the graph of some people who had electronically communicated with each other. The dataset consists of the following communication networks: Email[6], Email-Enron[1], Email-EuAll[7], and WikiTalk[1]. ***Friendship Networks***. These networks show the interactions of some social entities. The networks in this category are: Facebook-links[8], Slashdot0811[1], Slashdot0902[1], soc-Epinions1[1], Twitter-Richmond-FF[5], youtube-d-growth[8]and dolphins[9]. ***Web-graph Networks***. These networks show the graph of some web pages in which the edges correspond the hyperlinks. The members of this category are: Web-BerkStan[1], web-Google[1], web-NotreDame[1], and web-Stanford[1]. ***P2P Networks***. These networks represent peer-to-peer computer networks. In this class, the following net-

[1]http://snap.stanford.edu
[2]http://dblp.uni-trier.de/xml/
[3]http://citeseerx.ist.psu.edu
[4]http://www.sommer.jp/graphs/
[5]http://giuliorossetti.net/about/ongoing-works/datasets
[6]http://deim.urv.cat/ aarenas/data/welcome.htm
[7]http://konect.uni-koblenz.de/
[8]http://socialnetworks.mpi-sws.org
[9]http://www-personal.umich.edu/ mejn/netdata/

works are prepared: p2p-Gnutella04[10], p2p-Gnutella05[10], p2p-Gnutella06[10], and p2p-Gnutella08[10].

2) Artificial Networks

We have generated 8,000 artificial networks using eight generative models (1,000 network instances for each generative model). The selected generative models are Barabási-Albert model [10], copying model [11], [12], Erdős-Rényi [13], Forest Fire [14], Kronecker model [15], random power-law [16], Small-world (WattsStrogatz) model [17], and regular graph model. For each generative model, 1,000 network instances are generated using completely different parameters. The number of nodes in generated networks ranges from 1,000 to 5,000 nodes with the average of 2,936.34 nodes in each network instance. The average number of edges is 13,714.75. In this dataset, the generative models (generation methods) are the witnesses of the similarity: the networks generated from the same model follow identical link formation rules, and their degree distributions are considered similar. In both real and artificial networks datasets, it is possible for two different-class networks to be more similar than two same-class networks. But we can assume that the overall "expected similarity" among networks of the same class is more than the expected similarity of different-class networks. This definition of similarity based on the network types is frequently utilized in the literature (e.g., [18]).

B. Evaluation Criteria

In the section IV-A, we described our two network datasets and we introduced different signs and witnesses of similarities among networks of these datasets. We evaluate the network distance functions based on their consistency to mentioned witnesses of the similarity. For this purpose, we consider the following criteria:

1) kNN-Accuracy

The k-Nearest-Neighbor rule (kNN) is a common classification method which categorizes an unlabeled example by the majority label of its k-nearest neighbors in the training set. The performance of kNN is essentially dependent on the way that similarities are computed between different examples. Therefore, better distance metrics result in better classification accuracy of kNN. In a dataset of labeled instances, the KNN-accuracy of a distance metric d is the probability that the predicted class of an instance is equal to its actual class, when the distance metric d is used in the KNN classifier. In order to evaluate the accuracy of different distance functions, we employ them in kNN classification and we test the accuracy of this classifier.

2) Inter-class distances

An appropriate distance metric should return larger distances if the two compared networks are chosen from different classes. In other words, the distance metric is expected to report a small distance between two classmate networks and large distance between two networks of different classes. In order to evaluate a distance metric based on this requirement, we calculate the distance between any pair of networks of a dataset and we check the distance among non-classmate instances to be relatively larger. In order to compare different distance metrics, we normalize distances of each distance metric according to its mean and standard deviation. As Equation 13 shows, the *z-score* is used for normalizing distance values. In this formula, $\mu_{S,d}$ shows the average pairwise-distances for networks in dataset S, according to the d distance metric (Equation 11), and $\sigma_{S,d}$ indicates the standard deviation of the pairwise-distances (Equation 12). $nd_{S,d}(G_1, G_2)$ shows the normalized distance between G_1 and G_2 networks based on the population of S dataset of networks and d distance metric. Normalized distance (nd) is an appropriate base for evaluating the accuracy of distance metrics, since it is a dimensionless quantity (it shows the number of standard deviations that a distance is above the average distance). Z-score is widely used in the literature for similar purposes [19], [20], [21]. The normalization emphasizes the relative magnitude of the distances rather than their absolute magnitude, which is important for the comparison of computed distances in different distance functions. The average of normalized distances (nd) in a dataset is equal to zero, similar instances result a small (negative) normalized distance and dissimilar instances show large (positive) normalized distances. Equation 14 defines the average of normalized inter-class distances ($INTER_d$). $INTER_d$ shows the distance among networks with different classes, hence an appropriate distance function should indicate a large $INTER_d$ value.

$$\mu_{S,d} = \frac{1}{|S| \times (|S|-1)} \sum_{G_1,G_2 \in S, G_1 \neq G_2} d(G_1, G_2) \tag{11}$$

$$\sigma_{S,d} = \sqrt{\frac{1}{|S| \times (|S|-1)} \sum_{G_1,G_2 \in S, G_1 \neq G_2} (d(G_1, G_2) - \mu_{S,d})^2} \tag{12}$$

$$nd_{S,d}(G_1, G_2) = \frac{d(G_1, G_2) - \mu_{S,d}}{\sigma_{S,d}} \tag{13}$$

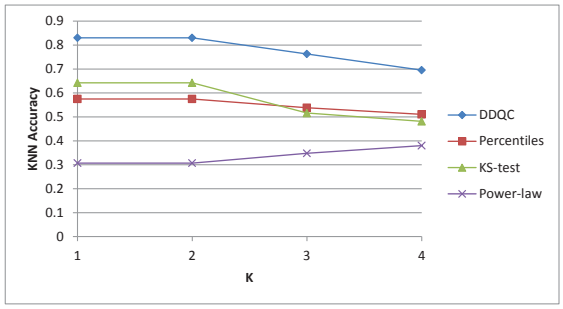
$$INTER_{S,d} = average(nd_{S,d}(G_1, G_2)); \quad G_1, G_2 \in S, class(G_1) \neq class(G_2) \tag{14}$$

Using the specified criteria, we compare our proposed method with three existing baseline methods: "Power-law", "KS-test" and "Percentiles". It is worth noting that "KS-test" actually does not include a quantification mechanism and needs the whole degree distributions to operate. This is a drawback of KS-test, since other baseline methods and our proposed distance metric need a small quantification of the degree distributions (e.g., a feature vector of eight real numbers) for comparing two networks.

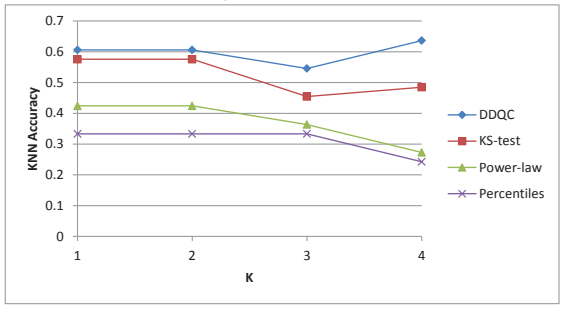
C. Evaluation Results

In this subsection, we comprehensively evaluate the proposed method (DDQC) and compare it with the baseline methods. As described in section III, the proposed method is configurable by three parameters: α, β and γ. We start the

[10]http://snap.stanford.edu

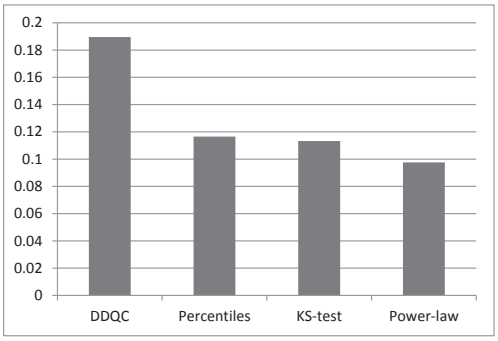


(a) kNN Accuracy in artificial networks dataset

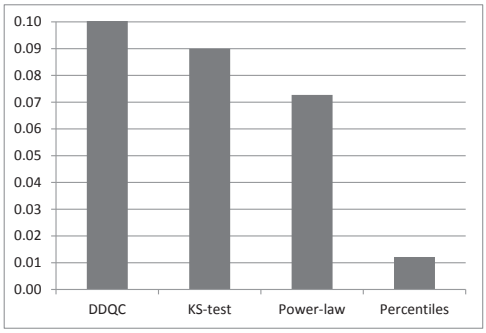


(b) kNN Accuracy in real-world networks dataset

Fig. 1: kNN Accuracy for different methods.



(a) $INTER_d$ for artificial networks



(b) $INTER_d$ for real-world networks

Fig. 2: $INTER_d$ for artificial and real-world datasets.

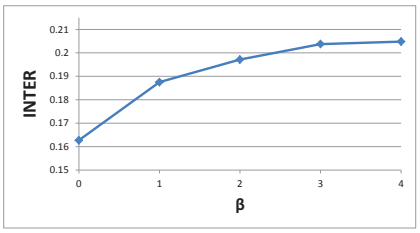


Fig. 3: The effect of β parameter on the accuracy of the proposed method in artificial networks dataset, for $INTER_d$

evaluations by setting $\alpha = 1$, $\beta = 1$ and $\gamma = 0.8$. Later, we will show the best values for these parameters.

For evaluating kNN accuracy on artificial networks dataset (Figure 1a), we iteratively created a small subset of this dataset and performed kNN on all instances of the formed subset. In each iteration of this experiment, we randomly selected 50 network instances from the dataset and computed the kNN accuracy for the set of these instances. Figure 1a shows this evaluation and reports the average of kNN accuracy on 100 independent iterations, for several values of K. Figure 1b shows the evaluation of different methods based on their kNN accuracy for real-world networks dataset. In this experiment, 33 real network instances, with known class labels, are classified using kNN algorithm and the average accuracy of the classifier is measured. According to Figures 1a and 1b, DDQC outperforms all the baseline methods considerably with respect to kNN-accuracy, in both datasets of real networks and artificial networks. The evaluations are performed for different values of K to ensure that the superiority of DDQC is not dependent on a particular K value.

In the next experiment, we evaluate different methods based on $INTER_d$ (Equation 14). As Figure 2 indicate, DDQC outperforms all the baseline methods with respect to $INTER_d$, in both datasets of real networks and artificial networks. As discussed before, a good distance metric should have a meaningfully larger values for $INTER_d$.

In order to evaluate the effect of β parameter on the accuracy of our proposed distance metric, we repeated the previous experiment with different values of β in the range of integer numbers from 0 to 4. As Figure 3 shows, the distance metric is improved by increasing the value of β and it asymptotically becomes stable with values larger than $\beta = 3$. Hence, $\beta = 3$ is an appropriate setting as a tradeoff between the accuracy of the distance metric and the size of the quantified vector (with $\beta = 3$ we will have $4 \times 2^3 = 32$ real numbers in the quantification of the degree distribution).

Finally, we examine different values of α and γ configuration parameters to find their best settings. Five values are tested for α as $\alpha = \langle 2^i \rangle_{i=-2,-1,0,1,2,3}$. Setting α to values out of this range (i.e., $\alpha > 8$ or $\alpha < 0.25$) makes the two middle regions of the degree distribution too wide or too narrow. For γ parameter, 20 different values are tested ($\gamma = \langle \frac{i}{10} \rangle_{i=1..20}$). Figure 4 shows the average inter-class distances of DDQC for artificial networks dataset, using the described values for α and γ. As Figure 4 indicates, the best parameter setting is $\alpha = 1$ and $\gamma = 0.8$ for both the diagrams. This setting is used for the parameters in our reported experiments. The diagram indicates a convex space with no other local optimum in this search experiment. The parameters may be further tuned via a

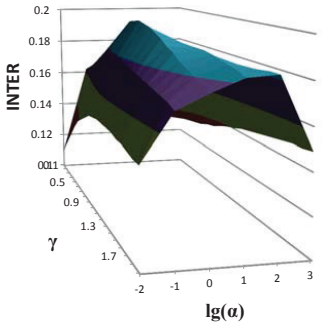


Fig. 4: Effect of α and β parameters on $INTER_d$.

fine-grained search in the set of real numbers. Since the search space (the collection of all possible solutions) is prohibitively large, intelligent search algorithms such as genetic algorithm or simulated annealing will improve the performance of the search.

V. Conclusion

In this paper, we first discussed the notion of distance and similarity for degree distributions. The degree distribution is an indicator of the link formation process in the network which reflects the overall pattern of connections [22]. Similarly evolving networks have analogous degree distributions, hence we derive similarity of degree distributions according to the similarity of link formation process in the networks. We proposed a novel method for quantification and comparison of network degree distributions. In order to derive the amount of similarity between the networks, we introduced admissible witnesses for network similarity: similarity among same-type real networks and same-model artificial networks. Kolmogorov-Smirnov (KS) test is currently the most common method for comparing the degree distributions. But KS-test does not support quantification and needs the whole degree distribution. Power-law exponent and Percentiles [2] are other measures for comparing degree distributions. Our proposed method, named DDQC, outperforms the existing algorithms with regard to its accuracy in various evaluation criteria.

As the future works, we will use the proposed quantification and comparison method in other application domains. Our proposed method enables the data analysis applications and data mining algorithms to employ the feature of the degree distribution as a fixed-length set of real numbers. It is now possible to represent a network instance with a record of features (including clustering coefficient, average path length and the quantified degree distribution) and use such records in data analysis applications. We will combine different network features along with the quantified degree distribution in an integrated distance metric for complex networks. Such an integrated distance metric will be the main building block of our future researches in evaluation and selection of network generative models and sampling methods.

References

[1] S. Motallebi, S. Aliakbary, and J. Habibi, "Generative model selection using a scalable and size-independent complex network classifier," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 23, no. 4, p. 043127, 2013.

[2] J. Janssen, M. Hurshman, and N. Kalyaniwalla, "Model selection for social networks using graphlets," *Internet Mathematics*, vol. 8, no. 4, pp. 338–363, 2012.

[3] J. Leskovec and C. Faloutsos, "Sampling from large graphs," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2006, pp. 631–636.

[4] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM review*, vol. 51, no. 4, pp. 661–703, 2009.

[5] L. Muchnik, S. Pei, L. C. Parra, S. D. Reis, J. S. Andrade Jr, S. Havlin, and H. A. Makse, "Origins of power-law degree distribution in the heterogeneity of human activity in social networks," *Scientific reports*, vol. 3, 2013.

[6] A. Sala, L. Cao, C. Wilson, R. Zablit, H. Zheng, and B. Y. Zhao, "Measurement-calibrated graph models for social network experiments," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 861–870.

[7] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *ACM SIGCOMM Computer Communication Review*, vol. 29, no. 4. ACM, 1999, pp. 251–262.

[8] N. Z. Gong, W. Xu, L. Huang, P. Mittal, E. Stefanov, V. Sekar, and D. Song, "Evolution of social-attribute networks: measurements, modeling, and implications using google+," in *Proceedings of the 2012 ACM conference on Internet measurement conference*. ACM, 2012, pp. 131–144.

[9] M. Kim and J. Leskovec, "Multiplicative attribute graph model of real-world networks," *Internet Mathematics*, vol. 8, no. 1-2, pp. 113–160, 2012.

[10] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[11] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal, "Stochastic models for the web graph," in *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*. IEEE, 2000, pp. 57–65.

[12] J. M. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins, "The web as a graph: Measurements, models, and methods," in *Proceedings of the International Conference on Combinatorics and Computing*. Springer, 1999, pp. 1–17.

[13] P. Erdös and A. Rényi, "On the central limit theorem for samples from a finite population," *Publications of the Mathematical Institute of the Hungarian Academy*, vol. 4, pp. 49–61, 1959.

[14] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, 2005, pp. 177–187.

[15] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and Z. Ghahramani, "Kronecker graphs: An approach to modeling networks," *The Journal of Machine Learning Research*, vol. 11, pp. 985–1042, 2010.

[16] D. Volchenkov and P. Blanchard, "An algorithm generating random graphs with power law degree distributions," *Physica A: Statistical Mechanics and its Applications*, vol. 315, no. 3, pp. 677–690, 2002.

[17] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[18] A. Mehler, "Structural similarities of complex networks: A computational model by example of wiki graphs," *Applied Artificial Intelligence*, vol. 22, no. 7-8, pp. 619–683, 2008.

[19] L. d. F. Costa, F. A. Rodrigues, G. Travieso, and P. Villas Boas, "Characterization of complex networks: A survey of measurements," *Advances in Physics*, vol. 56, no. 1, pp. 167–242, 2007.

[20] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Physics reports*, vol. 424, no. 4, pp. 175–308, 2006.

[21] M. Salehi, H. R. Rabiee, and M. Jalili, "Motif structure and cooperation in real-world complex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 389, no. 23, pp. 5521–5529, 2010.

[22] V. Boginski, S. Butenko, and P. M. Pardalos, "Mining market data: a network approach," *Computers & Operations Research*, vol. 33, no. 11, pp. 3171–3184, 2006.