Statistical Clustering of a Random Network by Extremal Properties

Natalia M. Markovich[1(✉)], Maxim S. Ryzhov[1], and Udo R. Krieger[2]

[1] V.A. Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Profsoyuznaya Str. 65, 117997 Moscow, Russia
markovic@ipu.rssi.ru

[2] Fakultät WIAI, Otto-Friedrich-Universität, An der Weberei 5, 96047 Bamberg, Germany
udo.krieger@ieee.org

Abstract. We propose the new EI-clustering method for random networks. Regarding the underlying graph of a random network, EI-clustering is an advanced statistical tool for community detection and based on the estimation of the extremal index (EI) associated with each node. The EI metric is estimated by samples of indices of the node influences. The latter quantities are determined by the PageRank and a Max-Linear Model. The EI values of both models are estimated by a blocks estimator for each node which is considered as the root of a Thorny Branching Tree. Generations of descendant nodes related to the root node of the tree are used as blocks. The reciprocal of the EI value indicates the average number of influential nodes per generation containing at least one influential node. In the context of random graphs the EI metric indicates the ability of a randomly selected node to attract highly ranked nodes in its orbit. Looking at the changing shape of a plot of the EI metric versus the node number, the node communities are detected. The EI-clustering method is compared with the conductance measure regarding the data set of a real Web graph.

Keywords: Clustering · Node influence · PageRank
Max-Linear Model · Extremal index · Web graph

1 Introduction

Random graphs are accepted as realistic models of real world networks such as technological networks, e.g. Internet, P2P, and transportation networks, social and information networks, [6,7,15]. Given the massive amount of data about real networks, the determination of the importance of nodes, a fast finding of the most influential nodes in a graph and the efficient detection of communities, i.e. clustering of nodes that are similar in some sense, constitute important research problems.

Clustering tools for random graphs such as the "null model" or the "configuration model" (see [6,15] for a survey) do not take into account the distributions

of extremes regarding the influence indices of nodes and their dependency. The intuition of those clustering methods is mostly based on the idea that nodes which are interconnected by a large number of edges are likely belonging to the same community. Then one has to find sets of nodes with a high internal connectivity that are highly disconnected between each other by calculating the number of edges of all the vertices in the network. Such approaches avoid the consideration of random graphs as "locally tree-like" structures due to loops and possible edges between vertices belonging to the same generation of a root node. A random graph with such a tree structure is called Thorny Branching Tree (TBT), [4].

These sketched clustering methods are based on an a-posteriori state of the network and do not allow to take into account random changes of links within the network and an on-line detection of structures. Moreover, a sudden explosion of edges of a node caused by local or temporary rare events like catastrophes in markets or sport events in social networks, that may generate new giant clusters, cannot be predicted by such tools. Those features require a new methodology concerning extremes of nodes in random graphs. It was theoretically derived in [11] that the tail index and extremal index of the PageRank (PR) and the Max-Linear Model (MLM) that are used as influence indices of the nodes in random graphs coincide. In [13] we have checked this property by means of real data of a Web graph.

In this paper it is our first objective to develop a clustering algorithm for the detection of communities of similar nodes in a random network. To partition a random network into (weakly dependent) communities around highly ranked nodes, we propose to use the extremal index (EI). The latter is a measure of dependence of extremes, [2,8]. The reciprocal of the EI approximates the mean cluster size of a structure. In this context a cluster is determined as a block of data with at least one exceedance over a given threshold. In a random graph the EI metric indicates the ability of a node u to perform clustering or, in other words, to attract influential nodes following u in its orbit.

In [1] semi-supervised learning methods are proposed for weighted similarity graphs. Two sets of nodes are determined as similar if they are connected by an edge and the weight of the edge indicates the strength of the similarity. We determine a community as a galaxy or a cluster of nodes related to a node with a large influence index, i.e. we detect extremes of the network and node followers of an extremal node as its community. Our approach is somewhat similar to [1], where local learning sets of similar nodes are used to build classifiers. We estimate the EI value of each node considering this entity as the root of a TBT, i.e. a branching tree with possible loops.

Our second objective is to compare the clustering approach that is based on the change of the conductance of a graph proposed in [10] by our EI-clustering based on the EI indices of the nodes.

The paper is organized as follows. In Sect. 2 basic results related to the detection of node communities and EI-estimation methods are sketched. Section 3 presents the EI-clustering algorithm and its comparison with the clustering based

on the conductance measure proposed in [10] by means of a real Web data set. The exposition is finalized by some conclusions.

2 Fundamental Properties of Random Networks

2.1 Community Detection by Clustering Methods

Surveys about graph partitioning techniques are presented in [6,10]. They reveal that the conductance metric and the clustering coefficient are important characteristics of random graphs.

The conductance measure of a set S of nodes in a random graph is calculated as

$$\phi = s/v, \quad \text{or} \quad \phi = s/(s + 2e). \tag{1}$$

Here s is the number of edges with one endpoint in S and one endpoint in $\overline{S}$, where $\overline{S}$ denotes the complement of S. v is the sum of degrees of nodes in S, and e is the number of edges with both endpoints is S, [10]. Small conductance of the set means that it is densely linked inside itself. It is remarkable that shape changes of the plot of ϕ against the number of nodes, called the Network Community Profile plot, indicate disconnected clusters.

The clustering coefficient C of a random network [15] is determined as

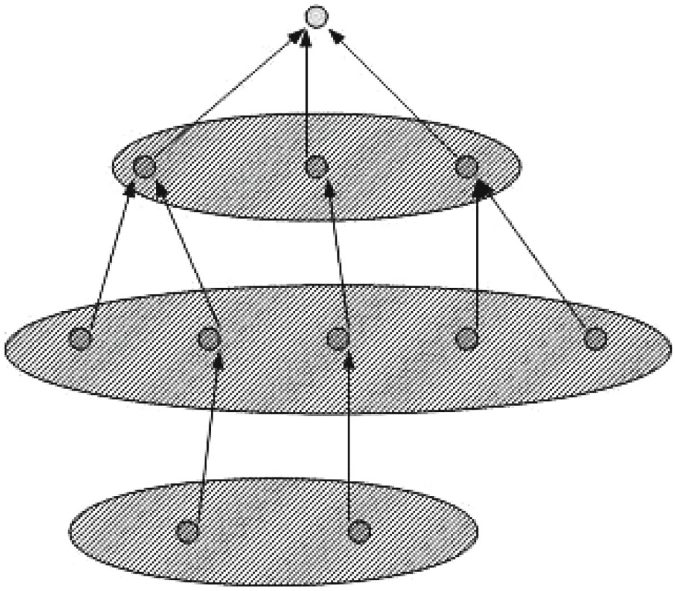
$$C = \frac{3 \cdot \text{number of triangles in the network}}{\text{number of connected triples}},$$

where a connected triple implies a single vertex connected by edges to two others. The triangle of nodes is considered as a basic social community. $C = 0$ means the lack of triangles. Then a part of the network associated with some node can be represented as a branching tree (see Fig. 1(a)). Descendants of each generation of the node taken as the root of a TBT are not linked and thus, independent. In reality, due to links between descendants within and between generations (see Fig. 1(b)), the dependence is determined in [15] by means of triangles.

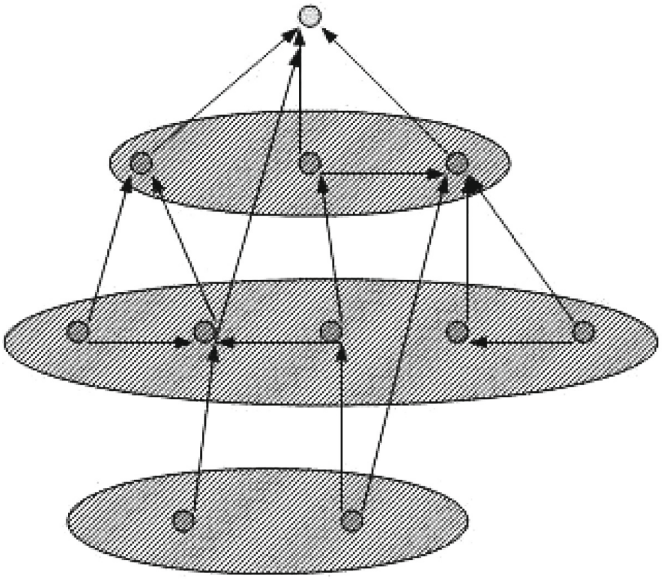
Our approach is to estimate the extremal index (EI) of each node influence in the random network. The EI value shows the ability of a node to create a cluster. It reveals the community built around the underlying node as a part of the Thorny Branching Tree associated with this node as its root.

2.2 Influence Indices of a Node

In fact, the in- and out-degrees of nodes are the only statistics gathered about a random network, [10]. The influence of a node may be determined by its in-degree, its PageRank (PR) and the Max-Linear Model (MLM), [12,13]. Both latter statistics are calculated by the in-degree and out-degree of a node and the PR values of those nodes that point to it. The PR of a node, e.g. a Web page, grows with the PRs of those nodes pointing to it and with the in-degree of the node. It can be calculated by different methods, see for instance [3,4,14].



(a)



(b)

Fig. 1. Branching tree corresponding to a cluster coefficient $C = 0$ (a); Branching tree with short loops containing triangles and single edges corresponding to $0 < C < 1$ (b); Generations of nodes following the root node are shown by grey ellipses.

2.3 The Extremal Index

Definition 1. ([8, p. 53])
The stationary sequence $\{R_n, n \geq 1\}$ *is said to have extremal index* $\theta \in [0,1]$ *if for each* $0 < \tau < \infty$ *there is a sequence of real numbers* $u_n = u_n(\tau), n \in \mathbb{N}$, *such that it holds*

$$\lim_{n\to\infty} n\overline{F}(u_n) = \lim_{n\to\infty} n(1 - F(u_n)) = \tau, \quad \lim_{n\to\infty} P\{M_n \leq u_n\} = e^{-\tau\theta}, \tag{2}$$

where $M_n = \max\{R_1, ..., R_n\} = \bigvee_{j=1}^{n} R_j$ *holds.*

Conditions (2) determine the thresholds u_n in such a way that the probability to exceed it are very small. This feature corresponds to high quantiles of $\{R_n\}$. For independent r.v.s $\theta = 1$ holds, but the converse is not true. $\theta \approx 0$ implies a strong dependence. $\theta = 0$ implies that the maximum M_n likely does not exceed a sufficiently high threshold $u \in \mathbb{R}$. As it holds [2]

$$\theta = \lim_{n\to\infty} \frac{P\{M_{r_n} > u_n\}}{r_n(1 - F(u_n))},$$

where $r_n = o(n)$ as $n \to \infty$, $P\{M_{r_n} > u_n\}$ implies the probability that a data block of size r_n contains at least one exceedance over the threshold u_n (such block is called a cluster) and $r_n(1 - F(u_n))$ shows the fraction of exceedances in the sample. Hence,

$$1/\theta \approx \frac{\text{number of exceedances}}{\text{number of clusters}} \tag{3}$$

approximates the mean cluster size.

With regard to a random graph the EI value of a node influence implies that each generation of descendants of the node that are accepted as blocks has on average $[1/\theta]$ nodes with high influence values. As the influence of a root node decreases over generations, one can use a truncated TBT such that its leaves do not impact significantly on the PR of the root. Such a truncation rule is specified in [13].

$\theta = 1$ or $\theta \approx 1$ mean that followers of a node which have links to a root node have independent or approximately independent influence characteristics. $\theta \approx 0$ implies a condensed cluster with regard to dependent influence values of followers.

2.4 Bias-Reduced Estimation of the Extremal Index

Nonparametric estimators of the extremal index (EI), like the blocks, runs, or intervals estimator, [2], are calculated by sequences of an underlying node characteristic. These estimators are usually calculated by random sequences or time series. They are distinguished by the definitions of the cluster. In this respect their application to random graphs constitutes further problems. The EI-estimators require one or two parameters. To apply the blocks estimator, for

instance, one can split the sample $\{X_1, ..., X_n\}$ of size n into $m_n = \lfloor n/r_n \rfloor \in \mathbb{N}$ blocks of length $r_n \in \mathbb{N}$. Then the blocks estimator is determined by

$$\widehat{\theta}_n = \frac{\sum_{j=1}^{m_n} \mathbb{1}\{\max_{(j-1)r_n < i \le jr_n} X_i > u_n\}}{\sum_{j=1}^{m_n} \sum_{i=(j-1)r_n+1}^{jr_n} \mathbb{1}\{X_i > u_n\}} \tag{4}$$

for a sequence of thresholds $u_n \in \mathbb{R}$ satisfying $r_n \overline{F}(u_n) \to 0$, but $n\overline{F}(u_n) \to \infty$.

Dealing with graphs and to create a sequence, we can only use the blocks estimator and its modifications to avoid the numeration of nodes in the graph. The blocks estimator requires the block size $r \in \mathbb{N}$ and the threshold $u \in \mathbb{R}$ as its parameters.

There are bias-reduced estimators of the EI which avoid a selection of u, [5,16]. These estimators have the advantage that their plots $\widehat{\theta}_n$ against u are stable. The stable plateau that is close to a constant indicates the estimated EI-value. Hence, one has to select only an appropriate parameter r.

Let $\{X_i, 1 \le i \le n \in \mathbb{N}\}$ be an univariate stationary time series with distribution function F and extremal index $\theta \in (0, 1]$. One may use the following bias-reduced estimator [5]

$$\widehat{\theta}_{n,t} = \frac{\sum_{j=1}^{m_n} \mathbb{1}\{\max_{(j-1)r_n < i \le jr_n} X_i > X_{n-\lceil nv_n t \rceil, n}\}}{\sum_{j=1}^{m_n} \sum_{i=(j-1)r_n+1}^{jr_n} \mathbb{1}\{X_i > X_{n-\lceil nv_n t \rceil, n}\}}, \quad t \in (0, 1], \tag{5}$$

where the sequence $\{v_n, n \in \mathbb{N}\}$ is such that $r_n v_n \to 0$, $nv_n \to \infty$ as $n \to \infty$.

In [12,13] generations of descendants of the TBT root node (see Fig. 1(b)) were proposed as blocks. Due to loops these blocks can be overlapping since the same node can be assigned to different generations. Such node may appear in one of the blocks or in all blocks which contain it. One can consider sets of nodes located on a path with m links (edges) from the root as the mth generation. To find the block size automatically and to build confidence intervals of the EI-estimate we use the bootstrap method described in [13].

3 The EI-Clustering Method

3.1 The EI-Clustering Algorithm

We study the clustering of n nodes in a random network using the extremal index of an influence characteristic of the nodes.

Algorithm 31

1. *Estimate the PR and the MLM values of each node by one of the recurrent methods in [13].*
2. *Estimate the EI values using samples of the PRs and MLMs of the lengths $k \in \{1, \dots, n\}$ by means of the blocks estimator (4) or by its bias-reduced modification (5). Find a threshold u by the bootstrap method [13] for a given block size r in the case (4). In the case (5) r can be found by the bootstrap method for a given parameter $0 < t \le 1$ (u is not required).*

3. *Given the EIs of the samples and enlarging the lengths* $k, k+1, \ldots, n$, *one calculates the average EI among the nodes of each of those samples.*
4. *Partition the nodes into clusters according to changes of the shape regarding the curves (node number, PR) or (node number, MLM).*

3.2 The Bootstrap Algorithm

Let us further interpret k and k_1 as the total numbers of exceedances over the threshold $u \in \mathbb{R}$ in the sample $\{X_i, 1 \leq i \leq n \in \mathbb{N}\}$ and in the bootstrap re-sample $\{X_i{}^*, 1 \leq i \leq n_1 \in \mathbb{N}\}$ of a smaller size $n_1 < n$, respectively, i.e.

$$k = \sum_{i=1}^{n} \mathbb{1}(X_i > u), \quad k_1 = \sum_{i=1}^{n_1} \mathbb{1}(X_i{}^* > u). \tag{6}$$

Then one can find u corresponding to the selected k and find the estimate of the extremal index $\widehat{\theta}(u)$.

Algorithm 32

1. *Generate* B *re-samples* $\{X_1^*, \ldots, X_{n_1}^*\}$ *of size* $n_1 < n$ *with replacement from the original observations* $\{X_i, i = 1, \ldots, n\}$, *where* n_1 *is defined as*

$$n_1 = n^{\beta_b}, \qquad 0 < \beta_b < 1.$$

The number of the largest order statistics $k_1 \in \{1, ..., n_1 - 1\}$ *corresponding to any re-sample relates to* k *and* n *by*

$$k = k_1 \left(\frac{n}{n_1}\right)^{\alpha_b}, 0 < \alpha_b < 1. \tag{7}$$

2. *Estimate* B *values* $\widehat{\theta}_{n_1}$ *using the blocks estimator (4) by each of* B *re-samples.*
3. *Calculate the mean squared error (MSE) by the re-samples,*

$$MSE\,(n_1, k_1) = (bias\,(n_1, k_1))^2 + var\,(n_1, k_1)\,, \tag{8}$$

where the bias and variance are the following quantities

$$bias\,(n_1, k_1) = \frac{1}{B}\sum_{b=1}^{B} \widehat{\theta}_{n_1} - \widehat{\theta}_n,$$

$$var\,(n_1, k_1) = \frac{1}{B-1}\sum_{b=1}^{B}\left(\frac{1}{B}\sum_{b=1}^{B}\widehat{\theta}_{n_1} - \widehat{\theta}_{n_1}\right)^2,$$

and find a minimal $MSE\,(n_1, k_1)$ *among different* $k_1 \in \{1, ..., n_1 - 1\}$.
4. *Using the obtained* k_1 *find the optimal* k *by (7) and then the corresponding estimate* $\widehat{\theta}_n$ *by (4).*

In this case the values α_b and β_b are not precisely known due to the lack of theory and we may take $\alpha_b = 2/3$ and $\beta_b = 1/2$ similar to the tail index estimation [13].

3.3 Comparison of Clustering Approaches

Similar to [13] we study a Web graph of the Berkeley-Stanford dataset [10]. Therein, nodes represent Web pages and edges represent hyperlinks between those pages, [9]. The selected graph contains 685230 nodes and 7600595 edges.

Figures 2 and 3 visualize the clustering of the 10000 most influential nodes regarding their influence indices based on PR and the MLM as well as the EI metrics that are estimated by the MLM values of each node by the blocks estimator (4) combined with the bootstrap method.

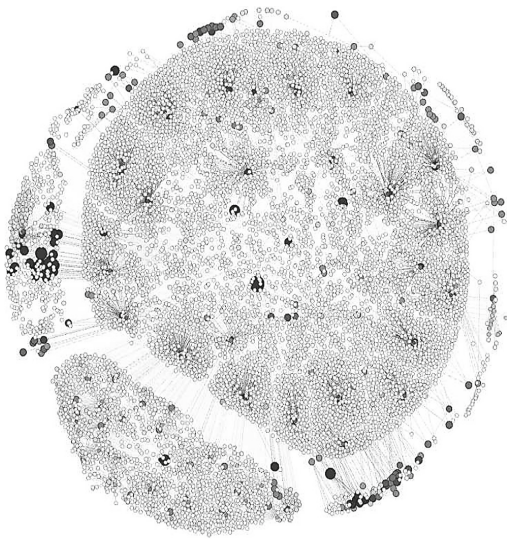


Fig. 2. Clustering of the sub-graph of the 10000 most influential nodes with regard to PR and the MLM from the Berkeley-Stanford dataset: the more intensive colour reflects the larger value of PR and, the bigger the circle is the larger is the MLM.

We apply Algorithm 31 to partition the network nodes into clusters both by the EI-values and conductance metrics for all nodes of the considered sub-graph. Figures 4 and 5 show that the changing shape of the conductance metrics (1) is close to those ones arising from the EI plots of the PR and the MLM of corresponding nodes. The EI-plot of PR is more sensitive than that one of the MLM regarding the community detection. The EI-values of PR and MLM tend to be similar for larger values of k which is in the agreement with the theoretical conclusions [11].

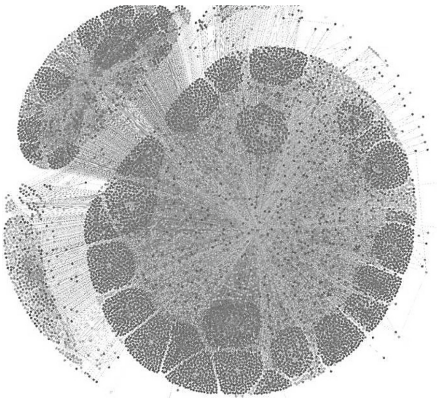


Fig. 3. Clustering of the same graph as in Fig. 2 with regard to EI values of the MLM: the more and the less intensive colours imply the EIs that are close to 1 and to 0, respectively.

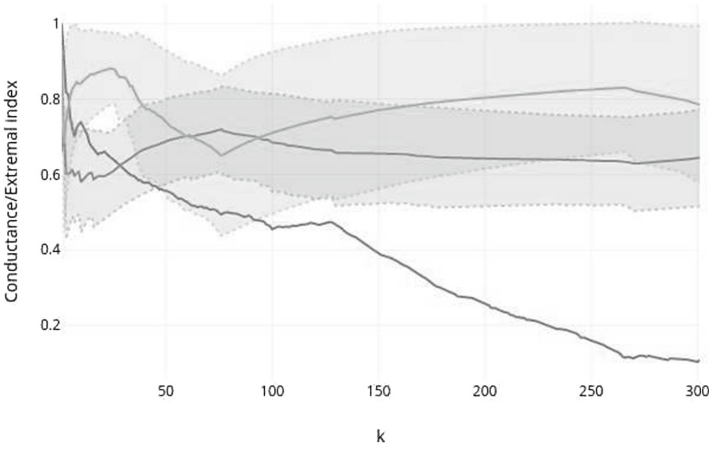
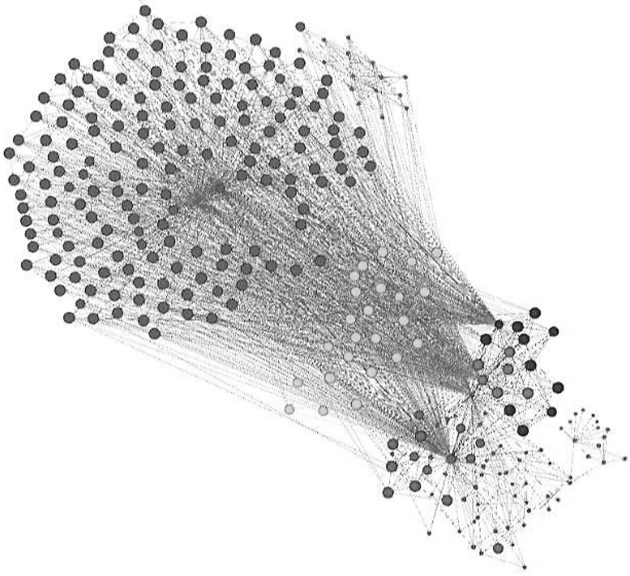
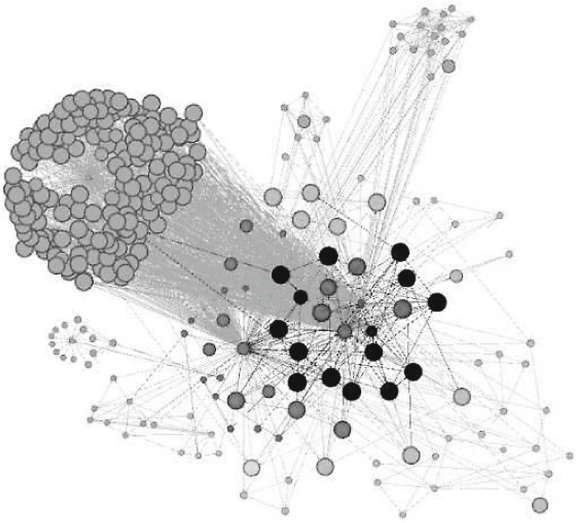


Fig. 4. Conductance (lower line at $k = 300$), the blocks estimates of the EI values of PR (upper line at $k = 300$) and the MLM (middle line at $k = 300$) of nodes with their 95% bootstrap confidence intervals against the numbers of nodes k.



(a)



(b)

Fig. 5. Clusters of nodes corresponding to conductance steps (a) and to steps of the EI-plot of the PR (b); Grey circles at left-hand side on top correspond to the smallest conductance values and EIs equal to 0.7.

4 Conclusions

We have proposed the EI-clustering algorithm for random networks. It is a new tool for community detection in random graphs based on the estimation of the extremal index (EI) of each node from an underlying vertex set. In contrast to known approaches, the EI index provides the dependence of extremes in the graph and shows the ability of a randomly selected node to attract highly ranked nodes in its orbit.

The EI-plots built by the PageRanks and the Max-Linear Model of nodes are compared with the conductance plot for a real data set. Finally, communities of nodes are partitioned corresponding to the changes of the shapes regarding the latter plots.

Our future study will elaborate on an on-line clustering algorithm for random networks.

References

1. Avrachenkov, K., Gonçalves, P., Sokol, M.: On the choice of kernel and labelled data in semi-supervised learning methods. In: Bonato, A., Mitzenmacher, M., Prałat, P. (eds.) WAW 2013. LNCS, vol. 8305, pp. 56–67. Springer, Cham (2013). https://doi.org/10.1007/978-3-319-03536-9_5
2. Beirlant, J., Goegebeur, Y., Teugels, J., Segers, J.: Statistics of Extremes: Theory and Applications. Wiley, Chichester (2004)
3. Borkar, V.S., Mathkar, A.S.: Reinforcement learning for matrix computations: pagerank as an example. In: Natarajan, R. (ed.) ICDCIT 2014. LNCS, vol. 8337, pp. 14–24. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04483-5_2
4. Chen, N., Litvak, N., Olvera-Cravioto, M.: Pagerank in scale-free random graphs. In: Bonato, A., Graham, F.C., Prałat, P. (eds.) WAW 2014. LNCS, vol. 8882, pp. 120–131. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13123-8_10
5. Drees, H.: Bias correction for estimators of the extremal index. arXiv:1107.0935 (2011)
6. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**(3–5), 75–174 (2010)
7. van der Hofstad, R.: Random Graphs and Complex Networks. Cambridge University Press, Cambridge (2016)
8. Leadbetter, M.R.: Probability Theory and Related Fields. Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete **65**(2), 291–306 (1983)
9. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford Large Network Dataset Collection (2014). http://snap.stanford.edu/data
10. Leskovec, J., Lang, K. J., Dasgupta, A., Mahoney, M. W.: Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. arXiv:0810.1355 (2008)
11. Markovich, N.M.: Extremes in Random Graphs Models of Complex Networks. arXiv:1704.01302v1 [math.ST], 5 April 2017
12. Markovich, N.M.: Analysis of clusters in network graphs for personalized web search. IFAC-PapersOnLine **50**(1), 5178–5183 (2017)
13. Markovich, N.M., Ryzhov, M., Krieger, U.R.: Nonparametric analysis of extremes on web graphs: pagerank versus max-linear model. In: Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V. (eds.) DCCN 2017. CCIS, vol. 700, pp. 13–26. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66836-9_2

14. Nazin, A.V., Polyak, B.T.: Randomized algorithm to determine the eigenvector of a stochastic matrix with application to the PageRank problem. Autom. Remote Control **72**(2), 342–352 (2011)
15. Newman, M.E.J.: Random graphs with clustering. Phys. Rev. Lett. **103**, 058701 (2009)
16. Sun, J., Samorodnitsky, G.: Estimating the extremal index, or, can one avoid the threshold-selection difficulty in extremal inference? Reports of Cornell University (2010)
17. Volkovich, Y., Litvak, N.: On the exceedance point process for a stationary sequence. Adv. Appl. Prob. **42**(2), 577–604 (2010)