Adv Syst Sci Appl 2022; **02**:21–33
Published online at https://ijassa.ipu.ru.

Information Spreading and Evolution of Non-Homogeneous Networks

Natalia M. Markovich[1]*, Maksim S. Ryzhov[1]

[1] *V.A. Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia*

Abstract: Information spreading among nodes of directed random networks by means of the linear preferential attachment (PA) schemes and the well-known SPREAD algorithm is considered. The novelty of the paper is that schemes of the linear preferential attachment proposed in Wan et al. (2020) for the network evolution are also used here for the information spreading. The SPREAD algorithm proposed for undirected random graphs is adapted to directed graphs. Moreover, we deal with non-homogeneous directed networks consisting of nodes whose in- and out-degrees have different power law distributions that is realistic for practice and we find communities in a network that spread the information faster. We compare the minimum number of evolution steps K^* required for the preferential attachment schemes and the well-known algorithm SPREAD to spread a message among a fixed number of nodes. The evolution of the network in time starts from a seed set of nodes. We study the impact of the seed network and parameters of the preferential attachment on K^* for simulated graphs. Real temporal graphs are also investigated in the same way. The PA may be a better spreader than the SPREAD algorithm. This is valid for the sets of the PA parameters with dominating proportions of created new edges from existing nodes to newly appending ones or between the existing nodes only. It is shown both for simulated and real graphs that the communities with the smallest tail indices of the out-degrees and PageRanks may spread the message faster than other communities.

Keywords: non-homogeneous directed network, information spreading time, linear preferential attachment, leading community, tail index

1. INTRODUCTION

The evolution of networks is arising in numerous applications: citation networks [1], the web-page popularity by PageRank during evolutionary changes [2], or the evolution of the network [3].

Information spreading, as a message delivery model in the whole network (the full spreading) [4] or in a part of the network (the partial spreading) [5], has an application for the parallel grid calculations in the computation network. In [6] the selection of a leading community in a network and the comparison of the spreading time by the latter and other communities are observed by an example of homogeneous geometric graphs.

Following [4] we consider nodes with asynchronous clocks. By the proposed model nodes of the network may spread their messages among other nodes by clicks of a global clock. The latter clicks are modelled as Poisson process and the inter-arrival times between clicks are therefore exponentially distributed. Thus, the spreading time required to spread messages may be calculated as a sum of a random number of exponentially distributed random variables. Generally, the clicks can be generated by another random process with another distribution of the inter-click times. In this paper we use the Poisson model and interpret the spreading

*Corresponding author: nat.markovich@gmail.com, markovic@ipu.rssi.ru rate in terms of a minimum number of clicks (or steps) required to spread the information among a part of nodes of the network.

In order to evolve the network starting from a seed set of nodes containing at least one node, we use a linear preferential attachment (PA) schemes proposed in [3]. We consider the PA as a spreading model additionally to its evolution function and compare it with the algorithm SPREAD proposed for spreading purposes in [4].

We aim to consider non-homogeneous networks containing subgraphs with stationary distributed in- and out-degrees of nodes. To this end, we partition the network into communities and use these communities as such subgraphs. A community is a set of highly connected nodes which are weak connected with nodes outside the community. Then we aim to find leading communities that may spread the information faster than the rest of the network.

On each evolution step, a configuration of the evolving network may change due to newly appended directed edges that may change the distribution of the in- and out-degrees of nodes. Apart of the in-degrees, we consider PageRanks as influence indices of the nodes.

Summarizing, the following novelties are implemented. Firstly, the information spreading is studied in a directed evolving graph where a probability to create new edges is provided by the linear PA schemes. The SPREAD algorithm proposed in [4, 5] for undirected stationary graphs is modified for directed graphs. Regarding the PA schemes, the probability to create newly appended edges may depend on the in- or (and) out-degrees and specific PA parameters. Secondly, a basic graph for the spreading and evolution is non-homogeneous as it is made of subgraphs with different distributions of node characteristics. This may lead to a different nodes' ability to spread information for different sub-graphs of the network. Finally, the impact of the heaviness of the distribution tail on the spreading rate is demonstrated on a range of real graphs.

The paper is organized as follows. In Section 2 related works and a necessary methodology are provided. In Section 3 we investigate simulated graphs regarding the spreading capacity of stationary parts of the network. In Section 4 the impact of a seed network on the spreading rate is studied for a real graph. A set of real graphs are compared in Section 5. The exposition is finalized with some conclusions.

2. PRELIMINARIES

2.1. *Information spreading*

The SPREAD algorithm proposed in [4] models a possible information spreading in undirected graphs. Let $G = (V, E)$ be the undirected graph with a set of nodes V and edges E. Considering an asynchronous time model (there are not quite synchronized clocks in all nodes), a node may initiate a communication by ticks of a global clock which are modelled as a Poisson process of rate $n = |V|$, [4, 5]. Let $k \geq 0$ denote the index of a tick, on which at most one node can receive messages by communicating with another node. On a tick one of n nodes of the graph is chosen uniformly. Then this node i chooses a node j uniformly among it neighbors with probability $P_{ij} = 1/D_{\max}$, where $D_{\max} = \max_{i \in V} D_i$, D_i is the degree of node i.

In [4, 5] nodes share with each other all information they have. Here, we consider the case when only a single message is spreading in the network. In Algorithm 1 we slightly change the SPREAD with this respect.

To apply the SPREAD algorithm in directed graphs it is plausible to assume that the node i may share its message with node j, if there is a directed edge $(i \to j)$ from i to j. Thus, we use $P_{ij} = 1/O_i$, where O_i is the out-degree of the node i.

Algorithm 1 The SPREAD

- At the tick k a node i is uniformly chosen in the graph, and it initiates a communication.
- The node i chooses a node j uniformly among its neighbors without the message with probability $P_{ij} = 1/O_i$ and sends its single message to the node j.

2.2. *Preferential attachment*

Now we consider the information spreading in directed graphs. Let us recall the linear PA model that is proposed and studied in [3, 7]. It starts with an initial directed graph $G(k_0)$ with at least one node and $k_0 \in N$ edges. For the non-negative parameters α, β, γ such that $\alpha + \beta + \gamma = 1$ holds, and specific parameters $\Delta_{in}, \Delta_{out}$, the model constructs a growing sequence of directed random graphs $G(k) = (V(k), E(k))$ depending on the step k. A graph $G(k)$ is generated from $G(k-1)$ by adding a directed edge. Furthermore, $N(k)$, $I_k(w)$ and $O_k(w)$ denote the number of nodes, in- and out- degrees of node w in the graph $G(k)$ with k edges. The following three scenarios of the edge creation, which are called $\alpha-$, $\beta-$ and $\gamma-$ schemes, respectively, are proposed in [3, 7]. To select schemes, one has to generate an i.i.d. sequence of trinomial random variables with values 1, 2 and 3 and the corresponding probabilities α, β and γ, see Algorithm 2.

Algorithm 2 The linear preferential attachment schemes

- By the $\alpha - scheme$ we append a new node w_{new} and create an edge $(w_{new} \to w)$ with probability α, where the existing node $w \in V(k-1)$ is chosen with probability

$$P(choose\ w \in V(k-1)) = \frac{I_{k-1}(w) + \Delta_{in}}{k - 1 + \Delta_{in} N(k-1)}.$$

- By the $\beta - scheme$ the existing nodes $w_1, w_2 \in V(k-1)$ are chosen independently with probability

$$P(choose\ w_1, w_2 \in V(k-1)) = \frac{O_{k-1}(w_1) + \Delta_{out}}{k - 1 + \Delta_{out} N(k-1)} \cdot \frac{I_{k-1}(w_2) + \Delta_{in}}{k - 1 + \Delta_{in} N(k-1)}$$

  and a new edge $(w_1 \to w_2)$ is appended with probability β.
- By the $\gamma - scheme$ the existing node $w \in V(k-1)$ is chosen with probability

$$P(choose\ w \in V(k-1)) = \frac{O_{k-1}(w) + \Delta_{out}}{k - 1 + \Delta_{out} N(k-1)},$$

  and an edge $(w \to w_{new})$ is appended with probability γ.

This algorithm implies that $N(k) = N(k-1)$ for β-scheme and $N(k) = N(k-1) + 1$ for the others. These scenarios realize a 'rich-get-richer' mechanism, when a node with a large number of in-/out- edges can likely increase them with a high probability. As mentioned in [3], such model can create multiple edges between pairs of nodes and self loops.

The parameters Δ_{in} and Δ_{out} may be estimated by the semi-parametric extreme value method (EV) based on the maximum-likelihood method or by the Snapshot (SN) method proposed in [8]. The latter is summarized in Algorithm 3. The snapshot graph $G(n) = (N(n), E(n))$ represents a point in time when the data about the graph are available.

Algorithm 3 Snapshot method

1. Let us estimate $\hat{\beta}^{SN}=1-N(n)/v$, where $v=|E(n)|$. We obtain $\hat{\Delta}_{in}^{0}$ and $\hat{\Delta}_{out}^{0}$ by solving the equations

$$\sum_{l=1}^{\infty}\frac{N_{>l}^{in}}{v}\frac{l}{l+\hat{\Delta}_{in}^{0}}(1+\hat{\Delta}_{in}^{0}(1-\hat{\beta}^{SN}))=\frac{\frac{N_{0}^{in}}{v}+\hat{\beta}^{SN}}{1-\frac{N_{0}^{in}}{n}\frac{\hat{\Delta}_{in}^{0}}{1+(1-\hat{\beta}^{SN})\hat{\Delta}_{in}^{0}}},$$
$$\sum_{l=1}^{\infty}\frac{N_{>l}^{out}}{v}\frac{l}{l+\hat{\Delta}_{out}^{0}}(1+\hat{\Delta}_{out}^{0}(1-\hat{\beta}^{SN}))=\frac{\frac{N_{0}^{out}}{v}+\hat{\beta}^{SN}}{1-\frac{N_{0}^{out}}{n}\frac{\hat{\Delta}_{out}^{0}}{1+(1-\hat{\beta}^{SN})\hat{\Delta}_{out}^{0}}},$$

   where N_{l}^{in} and N_{l}^{out} are the number of nodes with in- and out-degree equal to l, and thus, $N_{>l}^{in}=\sum\limits_{l'>l}N_{l'}^{in}$, $N_{>l}^{out}=\sum\limits_{l'>l}N_{l'}^{out}$ are the number of nodes with in- and out-degree larger than l.
2. Estimate $\hat{\alpha}^{0}$ and $\hat{\gamma}^{0}$ by

$$\hat{\alpha}^{0}=\frac{\frac{N_{0}^{in}}{v}+\hat{\beta}^{SN}}{1-\frac{N_{0}^{in}}{v}\frac{\hat{\Delta}_{in}^{0}}{1+(1-\hat{\beta}^{SN})\hat{\Delta}_{in}^{0}}}-\hat{\beta}^{SN},$$
$$\hat{\gamma}^{0}=\frac{\frac{N_{0}^{out}}{v}+\hat{\beta}^{SN}}{1-\frac{N_{0}^{out}}{v}\frac{\hat{\Delta}_{out}^{0}}{1+(1-\hat{\beta}^{SN})\hat{\Delta}_{out}^{0}}}-\hat{\beta}^{SN}.$$

3. Re-normalize the probabilities $(\hat{\alpha}^{SN},\hat{\beta}^{SN},\hat{\gamma}^{SN})=(\frac{\hat{\alpha}^{0}(1-\hat{\beta}^{SN})}{\hat{\alpha}^{0}+\hat{\gamma}^{0}},\hat{\beta}^{SN},\frac{\hat{\gamma}^{0}(1-\hat{\beta}^{SN})}{\hat{\alpha}^{0}+\hat{\gamma}^{0}})$. Obtain $\hat{\Delta}_{in}^{SN}$ and $\hat{\Delta}_{out}^{SN}$ by solving equations

$$\sum_{l=0}^{\infty}\frac{N_{>l}^{in}/v}{l+\hat{\Delta}_{in}^{SN}}-\frac{1-\hat{\alpha}^{SN}-\hat{\beta}^{SN}}{\hat{\Delta}_{in}^{SN}}-\frac{(\hat{\alpha}^{SN}+\hat{\beta}^{SN})(1-\hat{\beta}^{SN})}{1+(1-\hat{\beta}^{SN})\hat{\Delta}_{in}^{SN}}=0,$$
$$\sum_{l=0}^{\infty}\frac{N_{>l}^{out}/v}{l+\hat{\Delta}_{out}^{SN}}-\frac{1-\hat{\gamma}^{SN}-\hat{\beta}^{SN}}{\hat{\Delta}_{out}^{SN}}-\frac{(\hat{\gamma}^{SN}+\hat{\beta}^{SN})(1-\hat{\beta}^{SN})}{1+(1-\hat{\beta}^{SN})\hat{\Delta}_{out}^{SN}}=0.$$

2.3. *Tail index*

Let $\{X_n\}_{n\geq 1}$ be a stationary sequence of independent identically distributed (i.i.d) random variables (r.v.s) with distribution function (df) $F(x)$. The parameter α_{TI} is called the tail index (TI). It may be estimated with the Hill's estimator [9]

$$\widehat{\alpha}^{H}(k)=\left(\frac{1}{k}\sum_{i=1}^{k}log(\frac{X_{(n-i+1)}}{X_{(n-k)}})\right)^{-1}, \tag{2.1}$$

where $X_{(1)}\leq X_{(2)}\leq ...\leq X_{(n)}$ are the order statistics corresponding to the sample and the parameter k is a number of the largest order statistics. $\widehat{\alpha}^{H}(k)$ is derived assuming that the distribution tail is regularly varying with the tail index α_{TI}, i.e. $\overline{F}(x)=P\{X_1>$

$x\} = x^{-\alpha_{TI}}\ell(x)$, where $\ell(x)$ is a slowly varying function. The optimal value of k may be obtained by a smoothing method of the asymptotic mean squared error of the Hill's estimator (SAMSEE) [9]. The confidence interval around the optimal TI value is obtained by the bootstrap procedure with the 95% quantile.

2.4. Test of stationarity

The inhomogeneity is one of the problems of the statistical analysis of graphs. This means that the heaviness of the power law tail distributions of node characteristics, like in- and out-degrees and PageRanks may be different within the graph or its communities. To our best knowledge, there are no stationarity tests for graphs since there is no numeration of nodes in the graphs. Such a numeration may be determined by random walks used in graphs as sample tools. At the same time, there are stationarity tests for random sequences. One of the simplest ways to solve the problem is to use a number of random walks within the graph and to test the stationarity of the obtained sequences of the gathered nodes.
We use the stationarity test statistic

$$V/S = V_n/\widehat{s}^2_{n,q}, \;\; V_n = \frac{1}{n^2}\left[\sum_{k=1}^{n}(S_k^*)^2 - \frac{1}{n}\left(\sum_{k=1}^{n} S_k^*\right)^2\right], \;\; \widehat{s}^2_{n,q} = q^{-1}\sum_{i,j=1}^{q}\widehat{\gamma}_{i-j}, \tag{2.2}$$

$S_k^* = \sum_{j=1}^{k}(X_j - \overline{X}_n)$, $\widehat{\gamma}_j = n^{-1}\sum_{i=1}^{n-j}(X_i - \overline{X}_n)(X_{i+j} - \overline{X}_n)$, $\quad 0 \le j < n$, where the bandwidth $q = q_n$ satisfies $q \to \infty$, $q/n \to 0$ proposed in [10] for stochastic processes with the stationary distributed short range dependent noise. The null hypothesis of stationarity is rejected, if $V/S > c_\rho$, where c_ρ is a quantile of the asymptotic distribution function of the Kolmogorov's statistic $F_K(\pi\sqrt{x})$. $c_\rho \in \{0.190, 0.153\}$ holds for significant level $\rho \in \{5, 10\}\%$, respectively.

2.5. A node's PageRank

By Google's definition [11] the PageRank is determined as the rank of node (Web page) i by

$$R_i = \sum_{j\to i}\frac{c}{O_j}R_j + (1-c)q_i, \;\; i = \overline{1,n}, \tag{2.3}$$

where the sum is taken over all pages with incoming links to node i ($j \to i$ implies that node j is linked to node i, i.e. $(j,i) \in E$). The number of such pages constitutes the in-degree of the node i. O_j denotes the out-degree of the node j, i.e. the number of its outgoing links. $c \in (0,1)$ is a damping factor, i.e. a probability to browse a web-page connected with the current one that is set by Google as $c = 0.85$. $q_i \ge 0$ is a personalization probability of node i. The PageRank is a numeric measure of inter-relations between nodes that reflects the local network structure. To estimate the PageRank we use the iterative formula

$$\widehat{R}_i^{(n,0)} = 1, \;\; \widehat{R}_i^{(n,k)} = \sum_{j\to i}\frac{c}{D_j}\widehat{R}_j^{(n,k-1)} + (1-c), \;\; k \in N, \tag{2.4}$$

proposed in [12] for a given uniform personalization vector $q_i = 1/n, 1 \le i \le n = |V|$. The iteration (2.4) is proceeding until the difference between two consecutive iterations $|R_i^{(n,k)} - R_i^{(n,k-1)}|$ will be small enough which is sufficient for a moderate number of iterations k.

In [13], it is proved that PageRanks of nodes in a directed graph received by the PA are power law distributed. The tail of the latter distribution is heavier than the tail of the limiting in-degree distribution.

3. SIMULATION GRAPHS

Let us describe the information spreading mechanism by means of the PA. Let the initial directed graph $G(k_0)$ with N_0 nodes and k_0 edges be a seed set of nodes having a message to be spread. At global Poissonian clock ticks, we do a step of the PA (Section 2.2) with predefined parameters $\alpha, \beta, \gamma, \Delta_{in}, \Delta_{out}$. The latter parameters may be estimated by in- and out-degrees data sets as proposed in Section 3.3 in [3]. A newly appended edge may increase the number of nodes receiving the message. The message can be delivered from the node i to the node j, if the directed edge $(i \rightarrow j)$ is created from the node i to the node j which has not this message. This is possible only if the edge is created by the $\beta-$ or $\gamma-$ schemes. The $\alpha-$scheme provides edges with the opposite direction, namely, from the newly appearing node without the message to the existing node. Indeed, the message can be transmitted from the node having this message, only. At the kth clock tick we receive a graph $G(k) = (V(k), E(k))$ with $|E(k)| = k + k_0$ edges and $N(k) \leq N_0 + k$ nodes.

The PA schemes in [3] may generate multiple edges and self loops for some nodes. As a result, messages may get a stuck in these nodes and the corresponding spreading time increases.

We compare an ability to spread the information by the PA schemes and by the SPREAD algorithm given in Section 2.1. The graph is first simulated by the PA and the SPREAD is operating in the prepared graph.

We use the following parameters for the PA. The α, β and γ are taken within the interval $[0.04, 0.96]$ with the step 0.04, such that $\alpha + \beta + \gamma = 1$, and $\Delta_{in} = \Delta_{out} = 1$ holds. 100 simulated graphs are provided for the each set of the parameters. The graphs evolve starting from the initial triangle, that is a three nodes connected to each other, i.e. $k_0 = 3$, $N_0 = 3$ hold. At least one of the nodes has a message to spread. Selecting this triple of nodes within one of the communities of the network, we aim to find, what community may spread the message faster.

Let us assume that the number of steps k is limited as $k \leq K'$. We define the number of clock ticks required to disseminate the message from an initial node to n nodes with probability not less than $1 - \delta$ as

$$K^*(n, \delta) = inf\{0 < k \leq K' : Pr(|S(k)| = n) > 1 - \delta\},\ \delta \in (0, 1).$$

Here, $S(k)$ denotes the number of nodes which received the message at step k. Let us take K' equal to 5000. Then $S(K^*) < n$ means that K' steps of the evolution are likely not enough to disseminate the message to n nodes. The delivery delay is not less than $\sum_{i=1}^{K'} \tau_i$ since the inter-arrival times $\{\tau_i\}$ between the evolution steps are exponential distributed and the message cannot be spread if the corresponding edge is created by the $\alpha-$ scheme.

Results of the comparison of the PA and the SPREAD algorithms for $n = 100$ nodes are presented in Fig. 3.1.

Fig. 3.1 (left) shows the average $\langle K^* \rangle$ against the average $\langle q(K^*) \rangle$ over 100 simulated graphs, where $q(k) = \frac{|S(k)|}{N(k)}$ is a proportion of nodes that received the message in the graph $G(k)$. The $q(k)$ that is close to zero means a small part of nodes received the message at the tick k. $q(k) = 1$ means that all nodes in the graph have received the message. In the top line of Fig. 3.1, we obtain for the PA that the spreading rate depends on the β value for the same γ. The β value near 0 corresponds to the information spreading mostly to newly appearing nodes by the $\gamma-$schema. The β value near 1 means the information is delivered mostly between existed nodes by the $\beta-$schema. These situations may need the same tick number for the spreading. For the SPREAD, values of the β and γ affect the number of ticks. The β value near 1 corresponds to the large connections between nodes in some sets, which create the information ”stucking” in nodes. However, for the larger γ $\langle K^* \rangle$ becomes less dependent on the β. In Fig. 3.1 (middle) we show the proportion of the $\{S(K^*) < n\}$ events for the α, β, γ values. For the PA, these events are extremely rare except the case of the value $\gamma \leq 0.04$. This

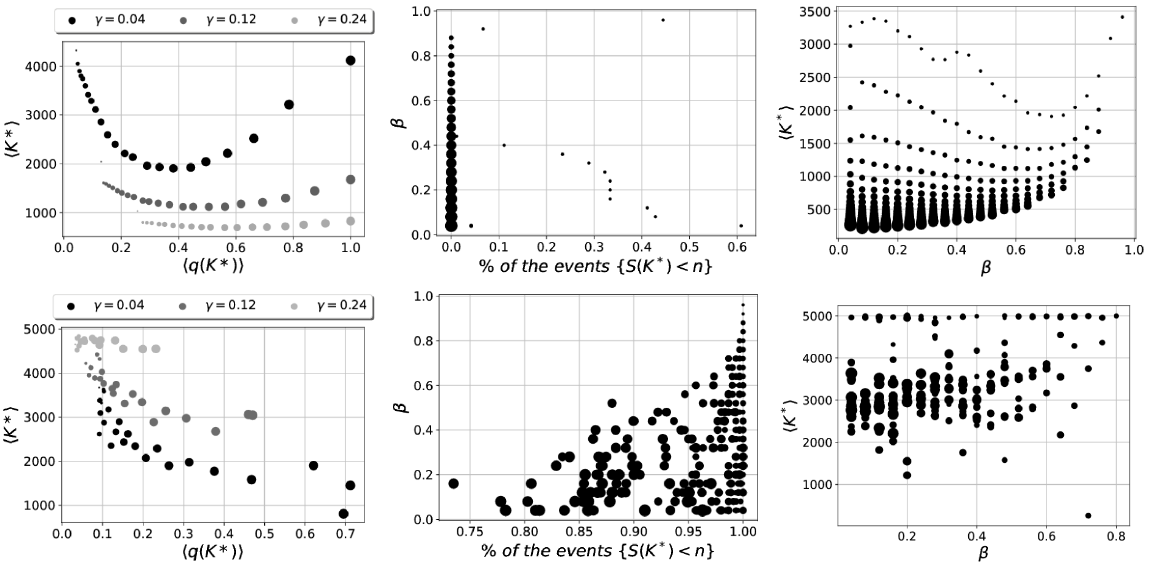

Fig. 3.1. The average clock ticks $\langle K^* \rangle$ against the average $\langle q(K^*) \rangle$ (left); the parameter β against the probability of $S(K^*) < n$ event (middle); $\langle K^* \rangle$ against β. 100 simulations with the same parameters are taken for the PA schemes (top line) and the SPREAD algorithm (bottom line). In the left figures, the point sizes are proportional to the value of $\beta \in [0.04, 0.96]$, in middle and right figures to the value of $\gamma \in [0.04, 0.96]$.

is the opposite to the SPREAD, where the increase of the β value causes the increase of the $\{S(K^*) < n\}$ event probability. In Fig. 3.1 (right) we show $\langle K^* \rangle$ for the α, β, γ values. For the PA and the SPREAD, we receive the same result that the decrease of the β value allows to spread the message quicker. This corresponds to a graph with a less number of edges between the nodes, in which the information may spread quicker. As a result, we notice that the choice of the set α, β, γ can make the spreading by the PA more effective than by the SPREAD.

Fig. 3.2 shows the impact of the parameters β and γ on $\log(K^*_{PA}/K^*_{SPREAD})$ as an averaging over 100 resamples. Let us recall that α is calculated as $\alpha = 1 - \beta - \gamma$. We define the parameter sets corresponding to $K^*_{PA} = K^*_{SPREAD}$, $K^*_{PA} > K^*_{SPREAD}$ and $K^*_{PA} < K^*_{SPREAD}$. It can be seen, that the PA can spread a message better if $\gamma > 0.51$ or $\beta > 0.6$ holds. The options when $\gamma + \beta > 1$ were not considered because $\alpha + \beta + \gamma = 1$ holds.

4. INVESTIGATION OF REAL GRAPHS

Here, we study the impact of a seed network having the message on $K^*(n, \delta)$ by real data. We investigate the graph obtained by the Berkeley-Stanford dataset with 685230 nodes and 7600595 edges [14], which represents Web pages from berkely.edu and stanford.edu domains that are connected in a union network by directed edges as hyperlinks between them. Within this network we select a small part and partition it into communities. Each of the latter communities may be used as the seed network. Starting from the seed network we apply the PA schemes to evolve the graph. Using the obtained evolved graph we apply the SPREAD algorithm and compare its capacity with the PA with regard to the spreading rate.

The underlying directed graph $G = (V, E)$ is partitioned into communities $\{C_i\}_{i=1}^5$, $\bigcup_{i=1}^5 C_i = V$, $C_i \bigcap C_j = \varnothing$, $i \neq j$, by means of the directed Louvain's algorithm in such a way that the graph modularity is maximized [15]. The modularity is a dependence measure of the graph that is determined by the identified communities. The modularity is large when communities are made of highly connected nodes.

We calculate the PageRank of each node of the underlying graph by the iterative formula (2.4). The result of the graph partitioning into communities is shown in Fig. 4.3. Since some

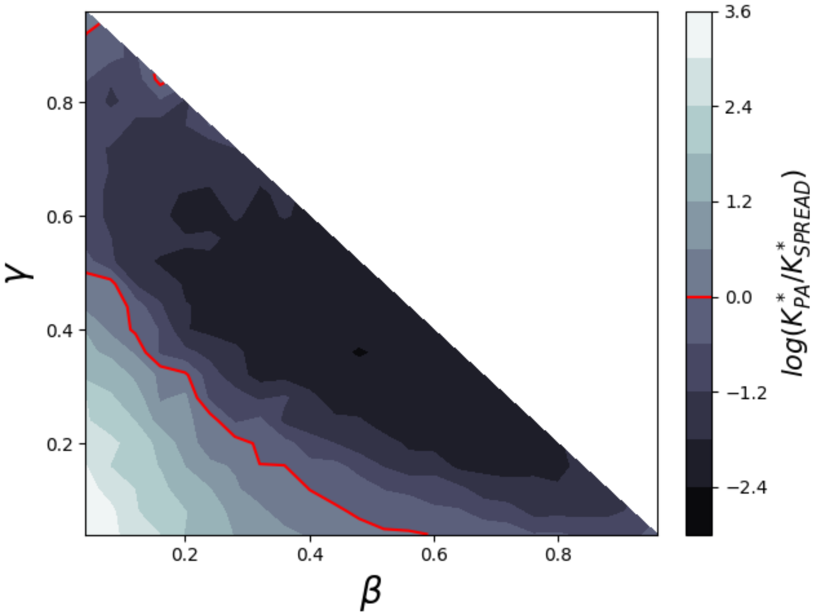

Fig. 3.2. The dependence of $\log(K^*_{PA}/K^*_{SPREAD})$ with the minimum number of steps K^*_{PA} and K^*_{SPREAD} for the PA and SPREAD algorithms, respectively, required to propagate a single message to $n = 100$ new nodes regarding the PA-parameters of the graph evolution. The red line indicates the case $K^*_{PA} = K^*_{SPREAD}$, to the left of the line is $K^*_{PA} > K^*_{SPREAD}$, to the right of the line is $K^*_{PA} < K^*_{SPREAD}$.

communities are highly connected, we may combine such communities together and consider $C_1 \cup C_3 \cup C_5$ and $C_2 \cup C_4$. However, the PageRanks of the latter combined communities may be non-stationary distributed.

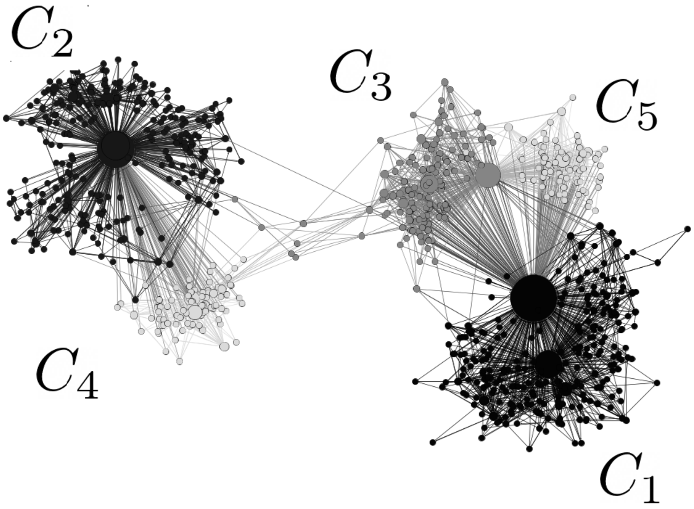

Fig. 4.3. Communities $\{C_i\}_{i=1}^5$ built by the Berkeley-Stanford data, where the point sizes are proportional to the node PageRanks.

We aim to study the impact of the heaviness of tails of the node characteristics, namely, the in- and out-degrees and PageRanks on the spreading rate. The in- and out-degrees are known to be power law distributed [1]. The PageRanks are proved to be regularly varying distributed [16]. Hence, the TIs of node characteristics of each community and their mergers may be estimated by the Hill's estimator (2.1). Indeed, this requires their stationarity that,

	$\|C\|$	V/S_{in}	V/S_{out}	V/S_{PR}
C_1	266	0.0579	0.1040	0.0764
C_2	266	0.0580	0.1058	0.0828
C_3	135	0.0773	0.0827	0.0774
C_4	86	0.0605	0.0632	0.0652
C_5	85	0.0439	0.0757	0.0457

Table 4.1. The V/S statistics (2.2) averaged over 100 random sequences and calculated by the in- and out-degrees and PageRanks of the communities $\{C_i\}_{i=1}^5$ and denoted by V/S_{in}, V/S_{out} and V/S_{PR}, respectively.

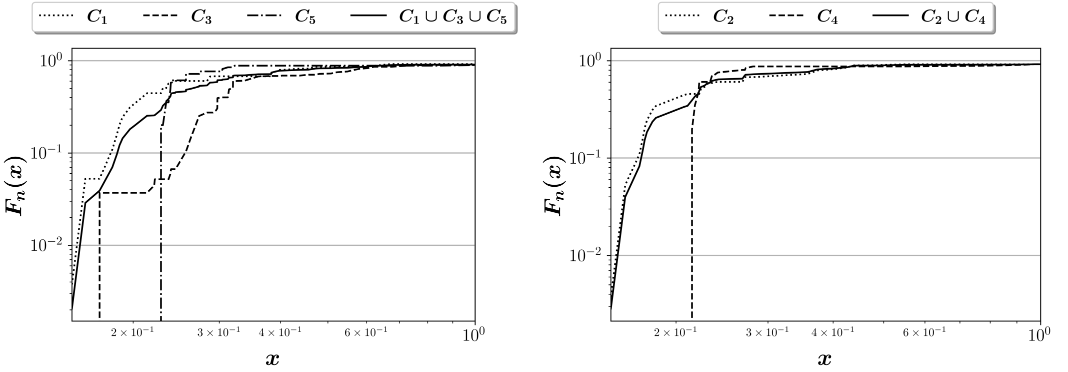

Fig. 4.4. Empirical distribution functions $F_n(x)$ of PageRanks of communities $\{C_i\}_{i=1}^5$ and their mergers $C_2 \cup C_4$ and $C_1 \cup C_3 \cup C_5$.

in practice, cannot be precisely fulfilled due to the inhomogeneous data. The finding of homogeneous communities is an unsolved problem that is out of scope of this paper.

In Table 4.1, the V/S statistics (2.2) and the sample sizes of the communities denoted as $|C|$ are shown. We use 100 sequences of PageRanks which correspond to possible numerations of nodes in the graph built by random walks. The latter are formed by uniform selection of each next node among nearest neighbors of a node. Random walks cover all nodes in the communites. The values in Table 4.1 do not contradict the null hypothesis of stationarity, but one cannot insist on the stationarity.

	$\|C\|$	$\widehat{\alpha}_{in}$	$\widehat{\alpha}_{out}$	$\widehat{\alpha}_{PR}$
C_1	266	0.895 (0.616, 1.175)	1.141 (1.035, 1.247)	0.474 (0.403, 0.544)
C_2	266	0.899 (0.629, 1.169)	1.141 (1.035, 1.247)	0.485 (0.444, 0.526)
C_3	135	1.117 (1.039, 1.195)	5.434 (5.351, 5.518)	0.667 (0.549, 0.786)
C_4	86	0.387 (0.2, 0.576)	3.712 (3.43, 4.009)	0.706 (0.3, 1.112)
C_5	85	0.425 (0.27, 0.58)	3.719 (3.427, 4.011)	1.798 (1.497, 2.099)
$C_1 \cup C_3 \cup C_5$	486	0.885 (0.669, 1.101)	3.236 (3.16, 3.312)	0.592 (0.523, 0.661)
$C_2 \cup C_4$	352	0.76 (0.474, 1.045)	1.48 (1.445, 1.515)	0.54 (0.478, 0.602)

Table 4.2. The TI estimates of the node in- and out-degrees and PageRanks of the communities $\{C_i\}_{i=1}^5$, $C_1 \cup C_3 \cup C_5$ and $C_2 \cup C_4$ denoted as $\widehat{\alpha}_{in}$, $\widehat{\alpha}_{out}$ and $\widehat{\alpha}_{PR}$, respectively, with their bootstrap confidence intervals in brackets.

In Table 4.2, the TIs with their bootstrap confidence intervals are shown. The TIs are estimated by the Hill's estimator (2.1) where the number of the largest order statistics k is calculated by the SAMSEE method [9].

Empirical distribution functions $F_n(x)$ of PageRanks for communities and their mergers are shown in Fig. 4.4. They are in agreement with Table 4.2. One can see that the heaviness of tail of the merges is determined by the dominating community in the merger, namely, the community with the smallest TI. Really, the PageRank distribution and the TI $\hat{\alpha}_{PR}$ of the community C_1 are close to those ones of $C_1 \cup C_3 \cup C_5$. The same is valid for the communities C_2 and the $C_2 \cup C_4$.

Moreover, C_1 and C_2 have close values of the TIs despite they are weak connected and the smallest TIs of the out-degrees and PageRanks. One may expect that the nodes of these communities can be the best spreaders.

Using the communities $\{C_i\}_{i=1}^5$ as seed networks for further evolution, we provide 50 simulations of the evolved graphs by the PA schemes. We check how the SPREAD algorithm and the PA schemes spread one message from each node in $\{C_i\}_{i=1}^5$ to the first arbitrary 100 nodes. We examine values of K^* for the parameters $(\alpha, \beta, \gamma) \in \{(0.4, 0.2, 0.4), (0.3, 0.1, 0.6)\}$ and $\Delta_{in} = \Delta_{out} = 1$ of the PA schemes. The number of the PA steps is upper bounded by $K' = 35 \cdot 10^4$. The resulted triples $(\min K^*, \langle K^* \rangle, \max K^*)$ with the minimum, average and maximum of K^* over 50 simulations of the application of the SPREAD algorithm and the PA schemes to each node of communities $\{C_i\}_{i=1}^5$ are presented in Table 4.3.

		(α, β, γ)	
Algorithm	Community	$(0.4, 0.2, 0.4)$	$(0.3, 0.1, 0.6)$
SPREAD	C_1	$(\mathbf{18866}, 128739.0, 232350)$	$(31675, 158485.4, 322566)$
	C_2	$(46166, 127821.3, 263166)$	$(\mathbf{27075}, 148624.7, 312100)$
	C_3	$(44242, 128236.4, 214050)$	$(37766, 149498.0, 305433)$
	C_4	$(62450, \mathit{120647.3}, 233325)$	$(44366, \mathit{140530.3}, \mathit{250233})$
	C_5	$(57900, 125888.5, \mathit{189200})$	$(82966, 159207.1, 332700)$
PA	C_1	$(\mathbf{2808}, 47019.0, 184560)$	$(4804, 54782.3, 166520)$
	C_2	$(3048, 46940.4, 190466)$	$(\mathbf{3907}, 55853.6, 161633)$
	C_3	$(3061, \mathit{32599.5}, \mathit{103911})$	$(6461, \mathit{41223.9}, \mathit{92542})$
	C_4	$(3960, 39293.6, 116357)$	$(6149, 44848.1, 141400)$
	C_5	$(2885, 42192.8, 117966)$	$(7470, 48430.5, 130366)$
SPREAD	$C_1 \cup C_3 \cup C_5$	$(\mathbf{18866}, 128100.8, \mathit{232350})$	$(31675, 156110.2, 332700)$
	$C_2 \cup C_4$	$(46166, \mathit{126068.6}, 263166)$	$(\mathbf{27075}, \mathit{146630.1}, \mathit{312100})$
PA	$C_1 \cup C_3 \cup C_5$	$(\mathbf{2808}, \mathit{42169.5}, 184560)$	$(4804, \mathit{49905.2}, 166520)$
	$C_2 \cup C_4$	$(4804, 54782.3, 166520)$	$(\mathbf{3907}, 53164.7, \mathit{161633})$

Table 4.3. The triples $(\min K^*, \langle K^* \rangle, \max K^*)$ corresponding to graphs generated by the PA schemes with two sets of parameters where the minimum values of $\min K^*$ are marked by bold, the minimum values of $\langle K^* \rangle$ and $\max K^*$ by italic.

Conclusions from Table 4.3 are the following. The communities C_1 or C_2 have the smallest values of minimum K^* for each set of parameters and each spreading algorithm. Both of them have the lowest TIs α_{out} and α_{PR}, see Table 4.2. Since the TI of the out-degrees is smaller than 2, the out-degrees have an infinite variance. The nodes with such large numbers of outgoing links are in fact the best spreaders. The minimum averages and maximum values of K^* relate to C_3 and C_4 with lighter distribution tails due to the relatively larger TIs.

5. COMPARISON OF SPREADING MODELS FOR REAL GRAPHS

Here, we investigate the effectiveness of the SPREAD and the PA spreading models (see, Sections 2.1 and 2.2) for real graphs. The temporal graphs (i.e. networks where edges have timestamps[†]) provided in [17] were used as an example. Among them there are graphs of messages and comments from websites (sx-mathoverflow, sx-askubuntu, CollegeMsg),

[†] For instance, a directed edge (u, v, t) means that person u sent an e-mail to person v at time t.

graphs of bitcoin transactions (soc-sign-bitcoin-otc, soc-sign-bitcoin-alpha) and graphs of e-mail communication (email-Eu, email-Eu-Dept1 and others). Their description can be found in Table 5.4.

Name	Number of nodes	Number of temporal edges	Description
sx-mathoverflow (M-Overlow)	24818	506550	Comments, questions, and answers on Math Overflow
sx-askubuntu (AskUb)	159316	964437	Comments, questions, and answers on Ask Ubuntu
email-Eu (EU)	986	332334	E-mails between users at a research institution
email-Eu-Dept1 (Eu-Dept1)	309	61046	
email-Eu-Dept2 (Eu-Dept2)	162	46772	
email-Eu-Dept3 (Eu-Dept3)	89	12216	
email-Eu-Dept4 (Eu-Dept4)	142	48141	
CollegeMsg (ColMsg)	1899	20296	Messages on a Facebook -like platform at UC-Irvine
soc-sign-bitcoin-otc (Bit-otc)	5881	35592	Bitcoin OTC web of trust network
soc-sign-bitcoin-alpha (Bit-alpha)	3783	24186	Bitcoin Alpha web of trust network

Table 5.4. Temporal Networks from [17] with their description.

The size of some graphs makes it computationally difficult to model the information spreading by the PA and SPREAD directly. Therefore, we conducted our evolutionary simulations starting from a single node to spread one message to 100 new nodes.

For each real graph the parameters of the PA model are evaluated by means of the Snapshot method (see, Algorithm 3) and denoted as $(\hat{\alpha}^{SN}, \hat{\beta}^{SN}, \hat{\gamma}^{SN}, \hat{\Delta}_{in}^{SN}, \hat{\Delta}_{out}^{SN})$. The effectiveness of both the SPREAD and PA algorithms is compared for different values of the latter parameters. The results are presented in Fig. 5.5, left, where $\hat{\Delta}_{in}^{SN}$ and $\hat{\Delta}_{out}^{SN}$ are found to be close to 1 for all graphs. Moreover, we estimate the TIs of the in- and out-degrees of nodes of all real graphs which are presented in Fig. 5.5, right.

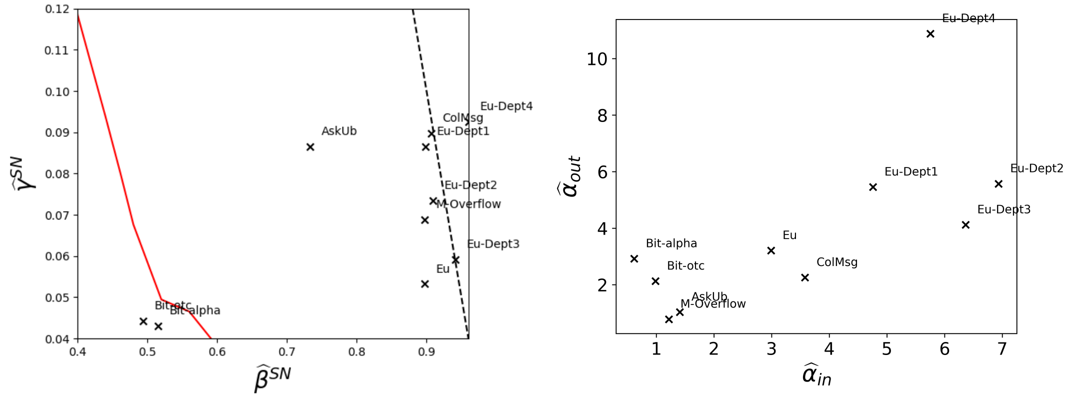

Fig. 5.5. The minimum $\log(K^*_{PA}/K^*_{SPREAD})$ for each graph achieved for the Snapshot estimates $\widehat{\beta}^{SN}$ and $\widehat{\gamma}^{SN}$ of the PA parameters, where the red line indicates the case $K^*_{PA} = K^*_{SPREAD}$, the area to the left of the line corresponds to $K^*_{PA} > K^*_{SPREAD}$, and to the right of the line to $K^*_{PA} < K^*_{SPREAD}$, the condition $\widehat{\alpha}^{SN} + \widehat{\beta}^{SN} + \widehat{\gamma}^{SN} = 1$ is fulfilled for the area to the left of the dotted line (left); the TIs of the in-degrees $\widehat{\alpha}_{in}$ and out-degrees $\widehat{\alpha}_{out}$ for the real networks (right).

For most of the investigated graphs the PA delivers messages faster, with the exception of the bitcoin graphs (soc-sign-bitcoin-otc and soc-sign-bitcoin-alpha). The TIs of the in-degrees of the latter graphs are less than 1, $\widehat{\alpha}_{in} < 1$. This implies that their in-degrees have an infinite variance according to the properties of the power law distribution. The PageRanks may have an even smaller TI due to [13]. Such likely large in-degrees appear in the bitcoin graphs due to the dominating effect of the $\alpha-$ scheme. That is a consequence of the small values of β and γ for these graphs, see Fig. 5.5, left. Since the $\alpha-$ scheme creates a new edge directed from the newly appending node to a node existed before, it cannot increase the number of nodes receiving the message at the evolution step. This shows the impact of the heaviness of tail of the node influence indices on the spreading rate.

6. CONCLUSIONS

The novelties of this paper are that the linear PA schemes are applied for the information spreading purpose, the SPREAD algorithm is reconsidered for directed graphs and both the PA and the SPREAD are applied to non-homogeneous graphs. A message from one node is spreading to a fixed number of nodes in the network. The information spreading is investigated both for simulated (Section 3) and real (Section 4) non-homogeneous graphs. The nodes in the latter graphs may have different distributions of their in- and out-degrees. We compare the PA and the SPREAD algorithm on directed graphs, which may contain cycles and multiple edges generated by the PA with different sets of parameters.

Considering the simulated graphs which are assumed to be homogeneous, one may conclude that the PA may be the better spreader than the SPREAD algorithm for such sets of its parameters (α, β, γ) where α is sufficiently small. The rest of the parameters $(\Delta_{in}, \Delta_{out})$ were taken equal to 1.

Regarding the real non-homogeneous graphs consisting of the interconnected communities of nodes with different tail indices of the in- and out-degrees and PageRanks, we found that some nodes in the community with the smallest tail index of the out-degrees and PageRanks may spread the message faster than other nodes.

We have classified the real temporal graphs in Section 5 by the number of steps required to disseminate a message from one node to a fixed number of nodes. It was found that the PA is a better spreader than the SPREAD tool for most of the considered graphs for which the PA α-scheme is not significant. The latter conclusion is plausible since the α-scheme creates new edges directed from the new nodes to the existing ones that cannot be useful for information spreading.

ACKNOWLEDGEMENTS

The reported study was funded by the Russian Science Foundation RSF, project number 22-21-00177 (recipient N.M. Markovich, conceptualization, methodology development, formal analysis, writing – original draft preparation; recipient M.S. Ryzhov, software, data validation).

REFERENCES

1. Newman, M. E. J. (2018). *Networks: An Introduction*. Oxford University Press, Second edition.
2. Avrachenkov, K. & Lebedev, D. (2006). PageRank of scale-free growing networks, *Internet Mathematics*, 3(2), 207-231.
3. Wan, P., Wang, T., Davis, R. A. & Resnick, S.I. (2020). Are extreme value estimation methods useful for network data? *Extremes*, 23, 171-195.

4. Mosk-Aoyama, D. & Shah, D. (2006). Computing separable functions via gossip, *Proc. of the 25th ACM symposium on Principles of distributed computing (PODC '06)*, ACM, New York, USA. 113-122.
5. Censor-Hillel, K. & Shachnai, H. (2010). Partial Information Spreading with Application to Distributed Maximum Coverage. *Proc. of the 29th ACM symposium on Principles of distributed computing (PODC '10)*, ACM, New York, USA. 161-170.
6. Markovich, N.M. & Ryzhov, M.S. (2020). Leader Nodes in Communities for Information Spreading, *LNCS* 12563, 475-484.
7. Bollobás, B., Borgs, C., Chayes, J. & Riordan, O. (2003). Directed scale-free graphs. *Proc. of the fourteenth annual ACM-SIAM symposium on Discrete algorithms (SODA '03)*, Society for Industrial and Applied Mathematics, USA. 132–139.
8. Wan, P., Wang, T., Davis, R. A. & Resnick, S.I. (2017). Fitting the linear preferential attachment model, *Electron. J. Statist.*, 11(2), 3738-3780.
9. Schneider, L.F., Krajina, A. & Krivobokova, T. (2021). Threshold selection in univariate extreme value analysis, *Extremes*, 2021, 24, 881–913.
10. Giraitis, L., Leipus, R. & Philippe, A. (2006). A test for stationarity versus trend and unit root for a wide class of dependent errors, *Econometric Theory*, 22, 989–1029.
11. Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine, *Computer Networks and ISDN Systems*, 30(1-7), 107-117.
12. Chen, N., Litvak, N. & Olvera-Cravioto, M. (2014). *PageRank in Scale-Free Random Graphs*. Bonato A. et al. (ed.), Springer, WAW 2014, LNCS 8882, 120–131.
13. Banerjee, S. & Olvera-Cravioto, M. (2021). PageRank Asymptotics on Directed Preferential Attachment Networks, arXiv:2102.08894
14. Leskovec, J., Lang, K., Dasgupta, A. & Mahoney, M. (2009). Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters, *Internet Mathematics*, 6(1), 29-123.
15. Dugué, N. & Perez, A. (2015). Directed Louvain: maximizing modularity in directed networks, Université d'Orléans.
16. Volkovich, Y. & Litvak, N. (2010). Asymptotic analysis for personalized web search, *Adv. Appl. Prob.*, 42(2), 577-604.
17. Leskovec, J. & Krevl, A. (2014). SNAP Datasets: Stanford Large Network Dataset Collection, https://snap.stanford.edu/data.