PROJECT PERIODIC REPORT

Grant Agreement number: 288956

Project acronym: NADINE

Project title: New tools and Algorithms for Directed Network analysis

Funding Scheme: Small or medium-scale focused research project (STREP)

Periodic report: **1st X 2nd**

Period covered: **from 1.5.2012** **to 31.10.2013**

Name, title and organisation of the scientific representative of the project's coordinator[1]:

Dr. Dima Shepelyansky

Directeur de recherche au CNRS

Lab de Phys. Theorique, Universite Paul Sabatier, 31062 Toulouse, France

Tel: +331 5 61556068, Fax: +33 5 61556065, Secr.: +33 5 61557572

E-mail: dima@irsamc.ups-tlse.fr; URL: www.quantware.ups-tlse.fr/dima

Project website address: **www.quantware.ups-tlse.fr/FETNADINE/**

[1] Usually the contact person of the coordinator as specified in Art. 8.1. of the grant agreement

NADINE DELIVERABLE D3.1.

It is based on milestones M3, M5, M12(in progress), M12(in progress) with deliverable publications:

[3] P1.3 V.Kandiah and D.L.Shepelyansky, **"PageRank model of opinion formation on social networks"**, Physica A v.391, p.5779 (2012) (arXiv:1204.3806v1 [physics.soc-ph], 2012)

[6] P1.6 L.Ermann, K.M.Frahm and D.L. Shepelyansky **"Spectral properties of Google matrix of Wikipedia and other networks"**, Eur. Phys. J. B v.86, p.193 (2013) (arXiv:1212.1068 [cs.IR], 2012

[9] P1.9 L.Chakhmakhchyan and D.L. Shepelyansky, **"PageRank model of opinion formation on Ulam networks"**, submitted to Phys. Lett. A (2013) (arXiv:1305.7395 [nlin.CD], 2013)

[12] P1.12 K.M.Frahm, Y.-H.Eom and D.L. Shepelyansky, **"Google matrix of the citation network of Physical Review"**, submitted to Phys. Rev. E Oct 21, 2013 (arXiv:1310.5624 [physics.soc-ph], 2013)

[19] P3.1 R.Pavlovics, and A.A.Benczur, **"Temporal influence over the Last.fm social network"**, The 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2013 Niagara Falls, Canada, August 25-28, 2013

[25] P4.1 L.Backstrom, P.Boldi, M.Rosa, J.Ugander, and S.Vigna. **"Four degrees of separation"**, ACM Web Science 2012: Conference Proceedings, pages 45-54, ACM Press (2012); best paper award, highlighted by New York Timse; (arXiv:1111.4570, 2012)

[26] P4.2 P.Boldi and S.Vigna. **"Four degrees of separation, really"** 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012), IEEE, 2012: 1222-1227 (arXiv:1205.5509, 2012)

[27] P4.3 P.Boldi, M.Rosa, **"Arc-Community Detection via Triangular Random Walks"**, LA-WEB 2012: 48-56 (2012)

[28] P.4.4 P.Boldi, M.Rosa, S.Vigna, **"Robustness of social and web graphs to node removal"**, Social Network Analysis and Mining, Springer: 1-14 (2012)

[29] P.4.5 P.Boldi, F.Bonchi, A.Gionis, T.Tassa, **"Injecting Uncertainty in Graphs for Identity Obfuscation"**, PVLDB 5(11): 1376-1387 (2012)

[30] P4.6 P.Boldi and S.Vigna. **"Axioms for centrality"** accepted for publications on Internet Mathematics (arXiv:1205.5509, 2012)

Physica A 391 (2012) 5779–5793

Contents lists available at SciVerse ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

PageRank model of opinion formation on social networks

Vivek Kandiah, Dima L. Shepelyansky *

Laboratoire de Physique Théorique du CNRS, IRSAMC, Université de Toulouse, UPS, F-31062 Toulouse, France

ARTICLE INFO

Article history:
Received 22 April 2012
Received in revised form 18 June 2012
Available online 23 June 2012

Keywords:
Voting
PageRank
Opinion formation

ABSTRACT

We propose the PageRank model of opinion formation and investigate its rich properties on real directed networks of the Universities of Cambridge and Oxford, LiveJournal, and Twitter. In this model, the opinion formation of linked electors is weighted with their PageRank probability. Such a probability is used by the Google search engine for ranking of web pages. We find that the society elite, corresponding to the top PageRank nodes, can impose its opinion on a significant fraction of the society. However, for a homogeneous distribution of two opinions, there exists a bistability range of opinions which depends on a conformist parameter characterizing the opinion formation. We find that the LiveJournal and Twitter networks have a stronger tendency to a totalitarian opinion formation than the university networks. We also analyze the Sznajd model generalized for scale-free networks with the weighted PageRank vote of electors.

1. Introduction

To understand the nature and origins of mass opinion formation is an outstanding challenge of democratic societies [1]. In the last few years the enormous development of such social networks as LiveJournal [2], Facebook [3], Twitter [4], and VKONTAKTE [5], with up to hundreds of millions of users, has demonstrated the growing influence of these networks on social and political life. The small-world scale-free structure of the social networks (see, e.g., Refs. [6,7]), combined with their rapid communication facilities, leads to a very fast information propagation over networks of electors, consumers, and citizens, making them very active on instantaneous social events. This invokes the need for new theoretical models which would allow one to understand the opinion formation process in modern society in the 21st century.

The important steps in the analysis of opinion formation have been done with the development of various voter models, described in great detail in Refs. [8–15]. This research field became known as sociophysics [8,10,12]. In this work, we introduce several new aspects which take into account the generic features of social networks. First, we analyze the opinion formation on real directed networks taken from the Academic Web Link Database Project of British university networks [16], the LiveJournal database [17], and the Twitter dataset [18]. This allows us to incorporate the correct scale-free network structure instead of unrealistic regular lattice networks, often considered in voter models [13,14]. Second, we assume that the opinion at a given node is formed by the opinions of its linked neighbors weighted with the PageRank probability of these network nodes. We argue that the introduction of such a weight represents the reality of social networks: all the network nodes are characterized by the PageRank vector which gives the probability of finding a random surfer on a given node, as described in Refs. [19,20]. This vector gives a steady-state probability distribution on the network which provides a natural ranking of node importance, or elector or society member importance. The PageRank vector is the right eigenvector with unit eigenvalue of the Google matrix constructed from the adjacency matrix of a given directed network. A detailed

* Corresponding author. Tel.: +33 561556068; fax: +33 561556065.
E-mail address: dima@irsamc.ups-tlse.fr (D.L. Shepelyansky).
URL: http://www.quantware.ups-tlse.fr/dima (D.L. Shepelyansky).

0378-4371/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.physa.2012.06.047

description of this vector and of Google matrix construction is given in Ref. [20]. The PageRank vector is used by the Google search engine for an efficient ranking of web pages [19,20].

In a certain sense, the top nodes of PageRank correspond to a political elite of the social network whose opinion influences the opinions of other members of the society [1]. Thus the proposed PageRank opinion formation (PROF) model takes into account the situation in which an opinion of an influential friend from high ranks of the society counts more than an opinion of a friend from a lower society level. We argue that the PageRank probability is the most natural form of ranking of society members. Indeed, the efficiency of PageRank rating is demonstrated for various types of scale-free network, including the World Wide Web (WWW) [19,20], *Physical Review* citation network [21,22], scientific journal rating [23], ranking of tennis players [24], Wikipedia articles [25], the world trade network [26], and others. Due to the above argument, we consider that the PROF model captures the reality of social networks, and below we present an analysis of its interesting properties.

We note that social networks have typical features which also appear in various sciences, including the economy [27,28], trader markets [29], world trade [26], and epidemic propagation [30,31], and hence we hope that the results presented in this work will find a broad field of applications there.

The paper is composed as follows. The PROF model is described in Section 2, and the numerical results on its properties are presented in Section 3 for British university networks. In Section 4, we combine the PROF model with the Sznajd model [13,32] and study the properties of the PROF–Sznajd model. In Section 5, we analyze the models on an example of a large social network, namely LiveJournal [17]. The results for the Twitter dataset [18] are presented in Section 6. A discussion of the results is presented in Section 7.

2. PageRank opinion formation (PROF) model description

The PROF model is defined in the following way. In agreement with the standard PageRank algorithm [20], we determine the PageRank probability P_i for each node i and arrange all N nodes in monotonic decreasing order of the probability. In this way each node i has a probability $P(K_i)$, and the PageRank index K_i with the maximal probability is at $K_i = 1$ ($\sum_{i=1}^{N} P(K_i) = 1$). We use the usual damping factor value $\alpha = 0.85$ to compute the PageRank vector of the Google matrix of the network (see, e.g., Refs. [19,20,33,34]). In addition, a network node i is characterized by an Ising spin variable σ_i which can take values $+1$ or -1, coded also by red or blue color, respectively. The sign of a node i is determined by its direct neighbors j, which have PageRank probabilities P_j. For that we compute the sum Σ_i over all directly linked neighbors j of node i:

$$\Sigma_i = a \sum_j P^{+}_{j,in} + b \sum_j P^{+}_{j,out} - a \sum_j P^{-}_{j,in} - b \sum_j P^{-}_{j,out}, \quad a + b = 1, \tag{1}$$

where $P_{j,in}$ and $P_{j,out}$ denote the PageRank probability P_j of a node j pointing to node i (incoming link) and a node j to which node i points to (outgoing link), respectively. Here, the two parameters a and b are used to tune the importance of incoming and outgoing links with the imposed relation $a + b = 1$ ($0 \leq a, b \leq 1$). The values P^{+} and P^{-} correspond to red and blue nodes, respectively. The spin σ_i takes the value 1 or -1, respectively, for $\Sigma_i > 0$ or $\Sigma_i < 0$. In a certain sense we can say that a large value of parameter b corresponds to a conformist society in which an elector i takes an opinion of other electors to which he/she points (nodes with many incoming links are on average at the top positions of PageRank). In contrast, a large value of a corresponds to a tenacious society in which an elector i takes mainly the opinion of those electors who point to him/her.

The condition (1) on spin inversion can be written via the effective Ising Hamiltonian H of the whole system of interacting spins:

$$H = -\sum_{i,j} J_{ij}\sigma_i\sigma_j = -\sum_i B_i\sigma_i = \sum_i \epsilon_i, \tag{2}$$

where the spin–spin interaction J_{ij} determines the local magnetic field B_i on a given node i:

$$B_i = \sum_j (aP_{j,in} + bP_{j,out})\sigma_j, \tag{3}$$

which gives the local spin energy $\epsilon_i = -B_i\sigma_i$. According to (2) and (3), the interaction between a selected spin i and its neighbors j is given by the PageRank probability: $J_{ij} = aP_{j,in} + bP_{j,out}$. Thus from a physical viewpoint the whole system can be viewed as a disordered ferromagnet [12,14]. In this way, condition (1) corresponds to a local energy ϵ_i minimization done at zero temperature. We note that such an analogy with spin systems is well known for opinion formation models on regular lattices [12–14]. However, it should be noted that generally we have asymmetric couplings $J_{ij} \neq J_{ji}$, which is unusual for physical problems (see the discussion in Ref. [35]). In view of this analogy, it is possible to introduce a finite temperature T and then to make a probabilistic Metropolis-type condition [36] for the spin i inversion determined by a thermal probability $\rho_i = \exp(-\Delta\epsilon_i/T)$, where $\Delta\epsilon_i$ is the energy difference between on-site energies ϵ_i with spin up and down. During the relaxation process, each spin is tested on an inversion condition that requires N steps and then we do t iterations of N such steps. We discuss the results of the relaxation process at both zero temperature and at finite temperature T in the next section. We use a standard random number generator to create an initial random distribution of spins σ_i up and down on

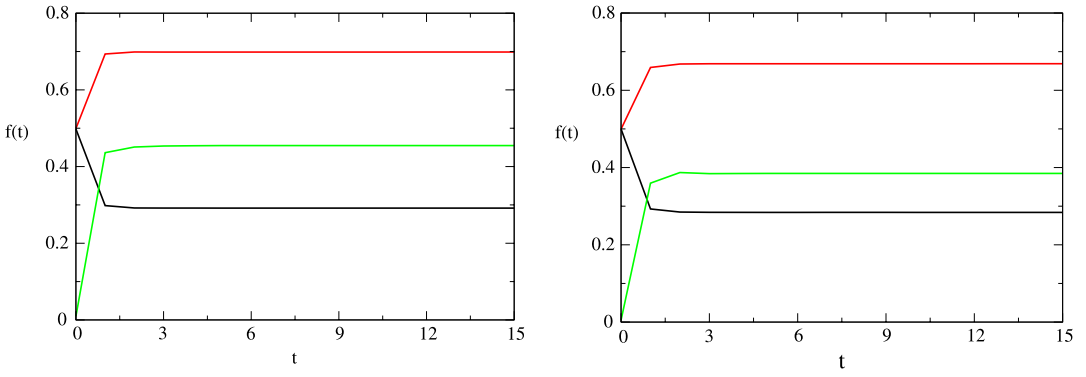


Fig. 1. (Color online) Time evolution of opinion given by a fraction of red nodes $f(t)$ as a function of number of iterations t. The red and black curves (top and bottom curves at $t = 15$, respectively) show evolution for two different realizations of a random distribution of color with the same initial fraction $f_i = 0.5$ at $t = 0$. The green curve (middle curve at $t = 15$) shows the dependence $f(t)$ for the initial state with N_{top} all red nodes with top PageRank K indexes (highest $P(K_i)$ values, $1 \le K \le N_{top}$). The evolution is done at $a = b = 0.5$ and temperature $T = 0$. *Left panel:* Cambridge network with $N_{top} = 2000$. *Right panel:* Oxford network with $N_{top} = 1000$.

nodes of a given network. We do averaging over $N_r \le 10^4$ such random generations to obtain statistically stable results for the final opinion distributions. The Metropolis Monte Carlo simulations follow the standard procedure described in Ref. [36].

3. Numerical results for the PROF model on university networks

Here we present results for the PROF model considered on the networks of the Universities of Cambridge and Oxford in 2006, taken from Ref. [16]. The properties of PageRank distribution $P(K)$ for these networks have been analyzed in Refs. [33,34]. The total numbers of nodes N and links N_ℓ are $N = 212710$, $N_\ell = 2015265$ (Cambridge); and $N = 200823$, $N_\ell = 1831542$ (Oxford) [34]. Both networks are characterized by an algebraic decay of PageRank probability $P(K) \propto 1/K^\beta$ and approximately usual exponent value $\beta \approx 0.9$; additional results on the scale-free properties of these networks are given in Refs. [33,34]. We usually discuss the fraction of red nodes, since by definition all other nodes are blue.

Typical examples of time evolution of the fraction of red nodes $f(t)$ with the number of time iterations t are shown in Fig. 1. We see the presence of bistability in the opinion formation: two random states with the same initial fraction of red nodes $f_i = f(t = 0)$ evolve to two different final fractions of red nodes f_f. The process gives an impression of convergence to a fixed state after $t_c \approx 10$ iterations. A special check shows that all node colors become fixed after this time (t_c). The convergence time to a fixed state is similar to those found for opinion formation on regular lattices, where $t_c = O(1)$ [13,14,37]. The corresponding time evolution of colors is shown in Fig. 2 for the first 10% of nodes ordered by the PageRank index K.

The results of Fig. 1 show that for a random initial distribution of colors we may have different final states with ± 0.2 variation compared to the initial $f_i = 0.5$. However, if we consider that N_{top} nodes with the top K index values (from 1 to N_{top}) have the same opinion (e.g. red nodes), then we find that even a small fraction of the total number of nodes N (e.g. N_{top} of about 0.5% or 1% of N) can impose its opinion on a significant fraction of nodes of about $f_f \approx 0.4$. This shows that in the frame of PROF model the society elite, corresponding to the top K nodes, can significantly influence the opinion of the whole society under the condition that the elite members have a fixed opinion between themselves.

We also considered the case when the red nodes are placed on $N_{top} = 2000$ top nodes of the CheiRank index K^*. This ranking is characterized by the CheiRank probability $P^*(K^*)$ for a random surfer moving in the inverted direction of links, as described in Refs. [25,34]. On average $P^*(K^*)$ is proportional to the number of outgoing links. However, in this case, the top nodes with small f_i values are not able to impose their opinion, and the final fraction becomes blue. We attribute this to the fact that the opinion condition (1) is determined by the PageRank probability $P(K)$ and that the correlations between CheiRank and PageRank are not very strong (see the discussion in Refs. [25,34]).

To analyze how the final fraction of red nodes f_f depends on its initial fraction f_i, we study the time evolution $f(t)$ for a large number N_r of initial random realizations of colors following it up to the convergence time for each realization. We find that the final red nodes are homogeneously distributed in K. Thus there is no specific preference for top society levels for an initial random distribution. The probability distribution W_f of final fractions f_f is shown in Fig. 3 as a function of initial fraction f_i at three values of parameter a. These results show two main features of the model: a small fraction of red opinion is completely suppressed if $f_i < f_c$ and its larger fraction dominates completely for $f_i > 1 - f_c$; there is a bistability phase for the initial opinion range $f_b \le f_i \le 1 - f_b$. Of course, there is a symmetry in respect to exchange of red and blue colors. For the small value $a = 0.1$ we have $f_b \approx f_c$ with $f_c \approx 0.25$, while for the large value $a = 0.9$ we have $f_c \approx 35$, $f_b \approx 0.45$.

Our interpretation of these results is the following. For small values of $a \to 0$ the opinion of a given society member is determined mainly by the PageRank of neighbors *to whom he/she points* (outgoing links). The PageRank probability P of nodes to which many nodes point is usually high, since P is proportional to the number of ingoing links [20]. Thus, at

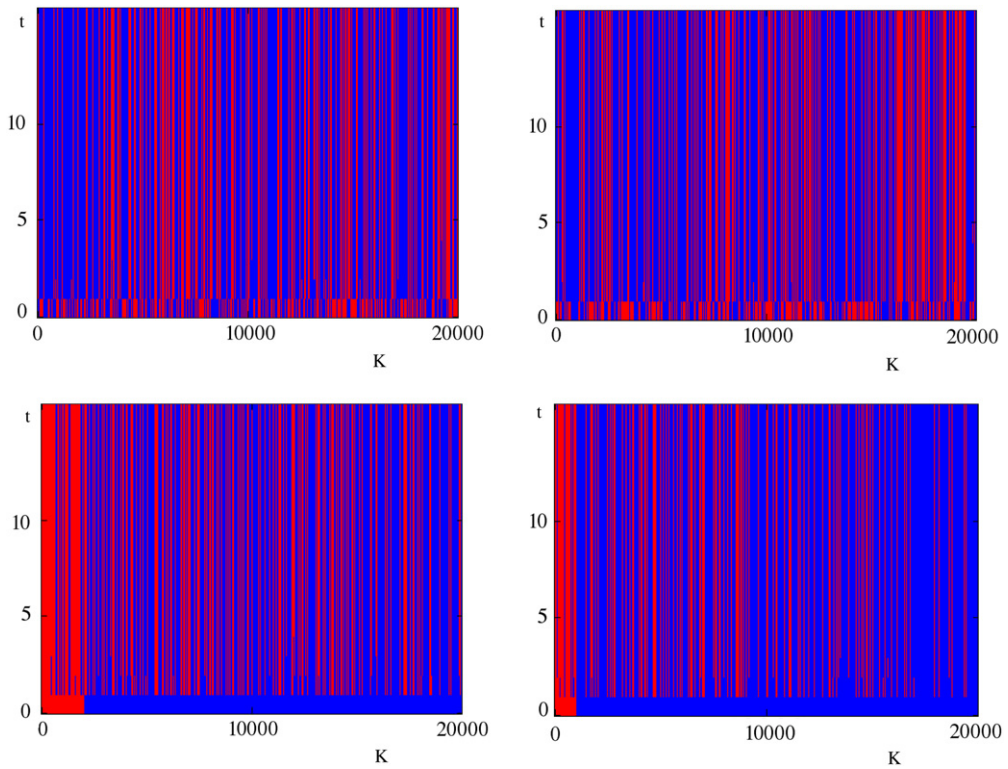


Fig. 2. (Color online) Time evolution of opinion colors (red/gray and blue/black) for the parameters of Fig. 1: the left/right column is for the Cambridge/Oxford network. The initial fraction of red colors is $f_i = 0.5$ (top panel), and N_{top} nodes have red color for the bottom panels, with $N_{top} = 2000$ and 1000 for the Cambridge network and the Oxford network, respectively. Nodes are ordered by the PageRank index K, and the color plot shows only $K \leq 20000$.

$a \to 0$, the society is composed of members who form their opinion by listening to an elite opinion. In such a society its elite with one color opinion can impose this opinion on a large fraction of the society. This is illustrated in Fig. 4, which shows a dependence of the final fraction f_f of red nodes on parameter a for a small initial fraction of red nodes in the top values of the PageRank index ($N_{top} = 2000$). We see that $a = 0$ corresponds to a conformist society which follows in its great majority the opinion of its elite. For $a = 1$, this fraction f_f drops significantly, showing that this corresponds to a regime of a tenacious society. It is somewhat surprising that the tenacious society ($a \to 1$) has a well-defined and relatively large fixed opinion phase with a relatively small region of bistability phase. This is in contrast to the conformist society at $a \to 0$, where the opinion is strongly influenced by the society elite. We attribute this to the fact that in Fig. 3 we start with a randomly distributed opinion, because the opinion of the elite has two fractions of two colors that create a bistable situation, since the two fractions of society follow the opinions of this divided elite, which makes the situation bistable on a larger interval of f_i compared to the case of a tenacious society at $a \to 1$.

To stress the important role of PageRank in the dependence of f_f on f_i presented in Fig. 3, we show in Fig. 5 the same analysis at $a = 0.5$, but for the case when in Eq. (1) for the spin flip we take all $P = 1$ (equal weight for all nodes). The data of Fig. 5 clearly demonstrate that in this case the bistability of opinion disappears. Thus the PROF model is qualitatively different from the case when only the links without their PageRank weight are counted for the spin flip condition. We also test the sensitivity in respect to PageRank probability by replacing P by $\sqrt{P}$ in Eq. (1), as is shown in Fig. 5 (bottom panels). We see that compared to the case $P = 1$ we start to have some signs of bistability, but still they remain rather weak compared to the case of Fig. 3.

In fact the spin flip condition (1) can be viewed as a relaxation process in a disordered ferromagnet (since all $J_{ij} \geq 0$ in (2) and (3)) at zero temperature. Such a type of analysis of voter model relaxation processes on regular lattices is analyzed in Refs. [13,14]. From this viewpoint it is natural to consider the effect of finite temperature T on this relaxation. At finite T, the flip condition is determined by the thermal Metropolis probability $\exp(-\Delta\epsilon_i/T)$, as described above. We follow this thermodynamic relaxation process at finite temperature up to $t = 200$ iterations, and in this way obtain the probability distribution of the final fraction f_f of red nodes obtained from the initial fraction f_i of red nodes randomly distributed over the network at $t = 0$. The results obtained at finite temperatures are shown in Fig. 6. They show that a finite temperature T allows a finite fraction f_f of red nodes when for their small initial fraction f_i all final f_f were equal to zero. Also, the bistability splitting is reduced and it disappears at larger values of T. Thus finite T introduces a certain smoothing in the W_f distribution.

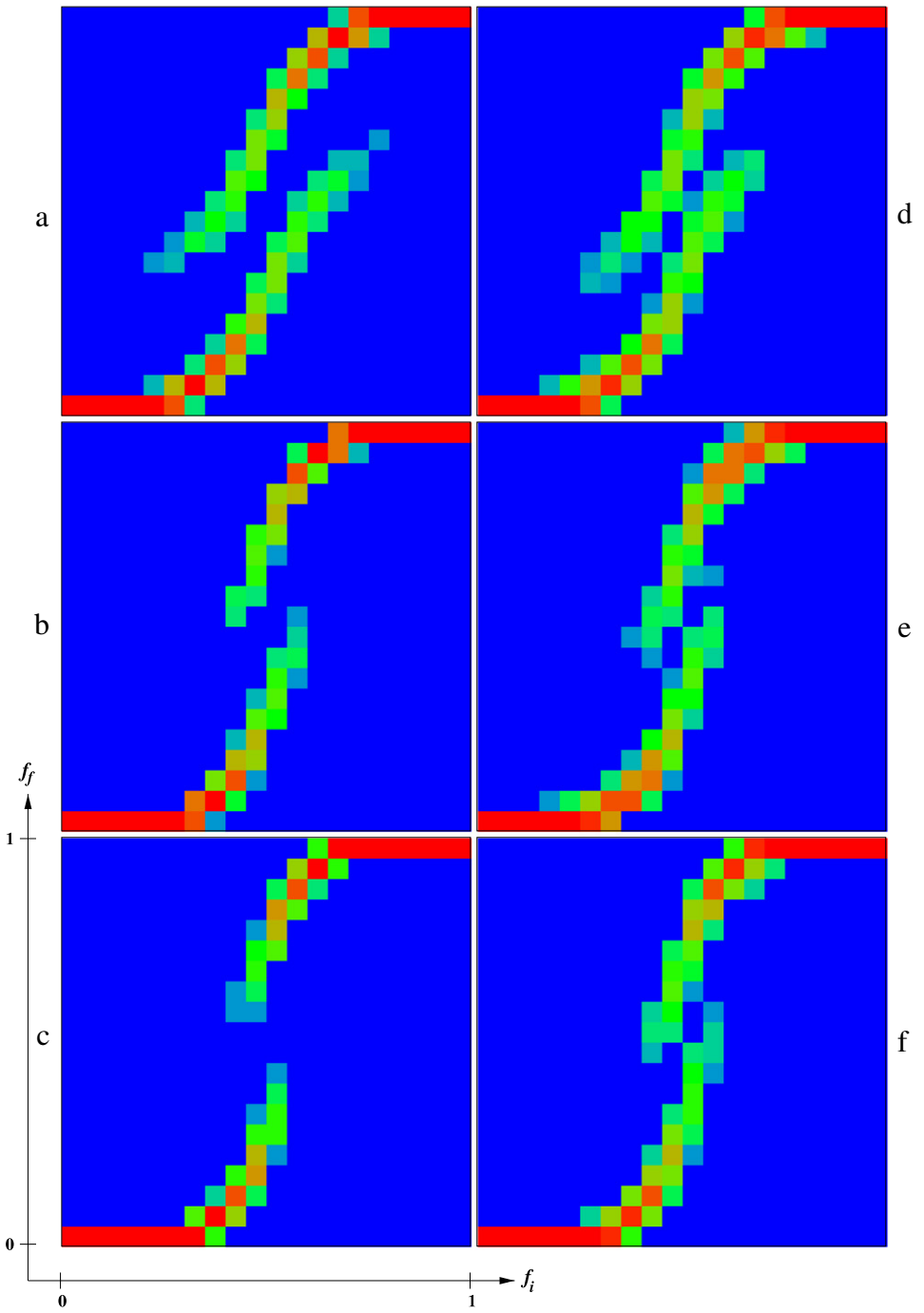


Fig. 3. (Color online) Density plot of probability W_f to find the dependence of the final red fraction f_f, shown on the y-axis, on the initial red fraction f_i, shown on the x-axis; data are shown inside the unit square $0 \leq f_i, f_f \leq 1$. The values of W_f are defined as the relative number of realizations found inside each of 20×20 cells which cover the whole unit square. Here, $N_r = 10^4$ realizations of randomly distributed colors are used to obtain the W_f values; for each realization, the time evolution is followed up to the convergence time with up to $t = 20$ iterations; here $T = 0$. *Left column:* Cambridge network (a, b, c); *right column:* Oxford network (d, e, f); here, $a = 0.1$ (a, d), 0.5 (b, e), 0.9 (c, f) from top to bottom. The probability W_f is proportional to color changing from zero (blue/black) to unity (red/gray).

However, the relaxation process at finite temperatures does not lead to the thermal Boltzmann distribution. Indeed, in Fig. 7, we show the probability distribution $w_i(\epsilon_i)$ as a function of local energies ϵ_i defined in (2) and (3). The distribution $w_i(\epsilon_i)$ is obtained from the relaxation process with many initial random spin realizations N_r. Even if the temperature T is

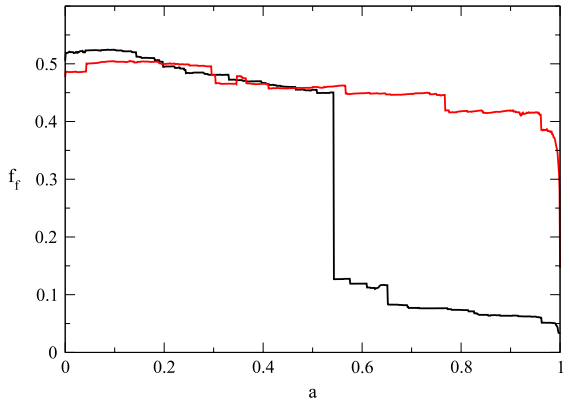


Fig. 4. (Color online) Dependence of the final fraction of red nodes f_f on the tenacious parameter a (or conformist parameter $b = 1 - a$) for initial red nodes in $N_{top} = 2000$ values of the PageRank index ($1 \leq K \leq N_{top}$); black and red/gray curves show data for Cambridge and Oxford networks; here, $T = 0$.

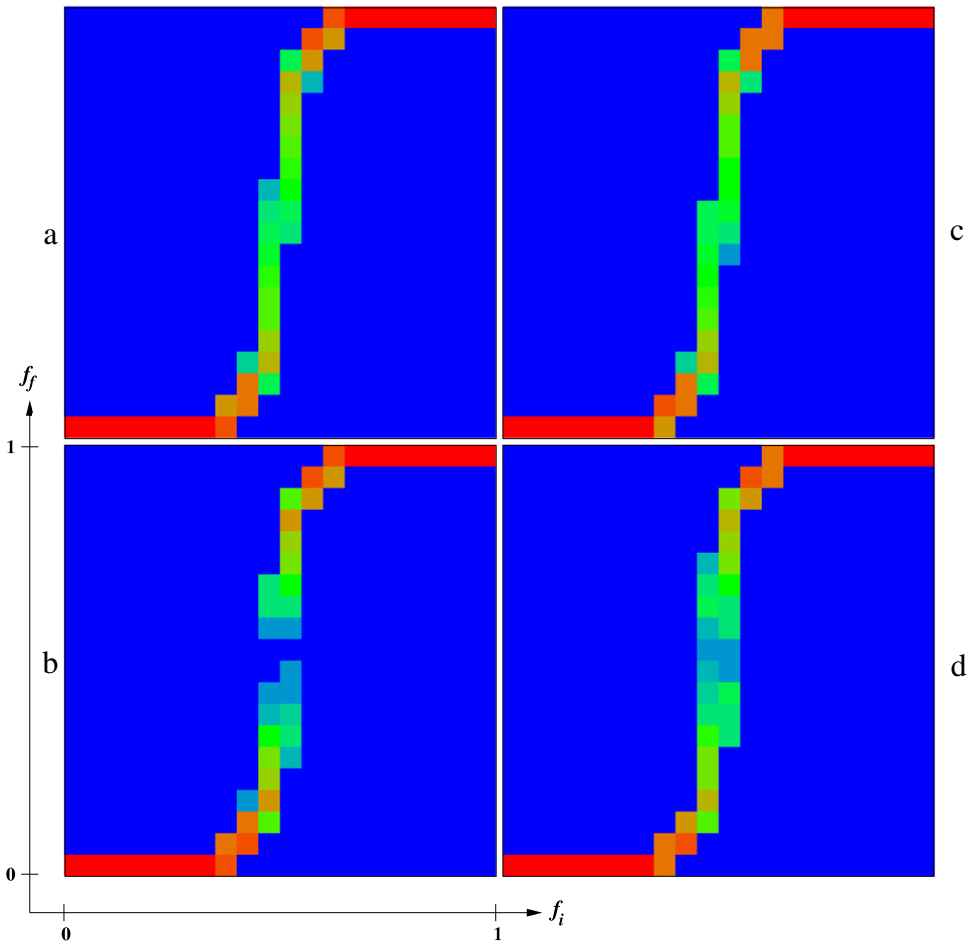


Fig. 5. (Color online) The same as in Fig. 3 (middle panels) at $a = 0.5$ but with uniform condition for spin flip being independent of PageRank probability (top panels (a, c): $P = 1$ in Eq. (1)) and PageRank probability P replaced by $\sqrt{P}$ in Eq. (1) (bottom panels (b, d)); the left and right panels correspond to Cambridge (a, b) and Oxford (c, d) networks; here, $T = 0$, and $N_r = 10^4$ realizations are used.

comparable with typical values of local energies ϵ_i, we still obtain a rather peaked distribution at $\epsilon_i \approx 0$ being very different from the Boltzmann distribution.

We argue that a physical reason of significantly non-Boltzmann distribution is related to the local nature of the spin flip condition which does not allow the production of a good thermalization on the scale of the whole system. Indeed, there are various energetic branches, and probably nonlocal thermalization flips of group of spins are required for a better

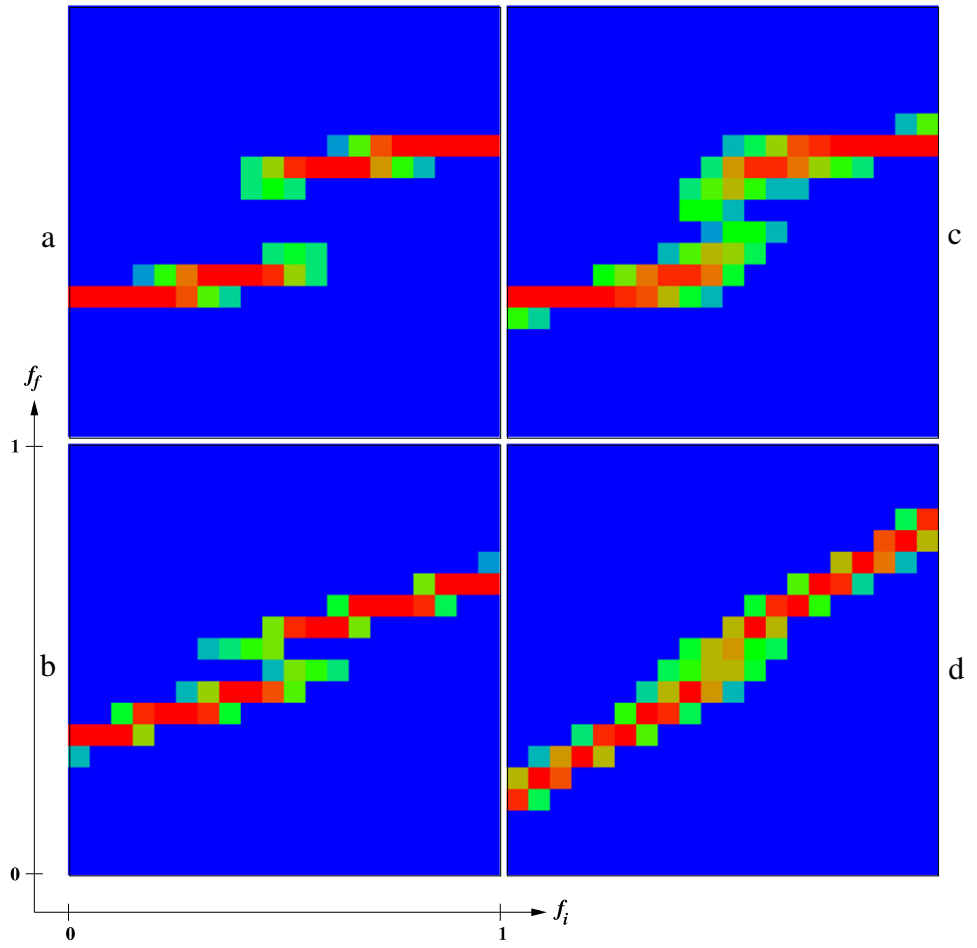


Fig. 6. (Color online) The same as in Fig. 3 (middle panel) at $a = 0.5$, but at finite temperature T during the relaxation process with $T = 0.001$ (top panels (a, c)) and $T = 0.01$ (bottom panels (b, d)); the number of random initial realizations is $N_r = 6000$, and the relaxation is done during $t = 200$ iterations. Left and right columns correspond to Cambridge (a, b) and Oxford (c, d) networks.

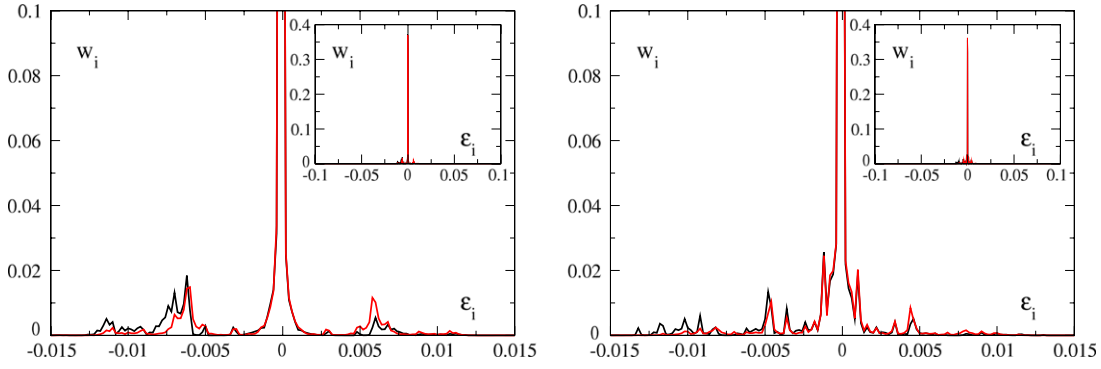


Fig. 7. (Color online) Normalized histograms of probability distribution w_i over local energies ϵ_i obtained from the relaxation process during $t = 10^3$ time iterations at temperatures $T = 0.01$ (black curve) and $T = 0.05$ (red/gray curve); the average is taken over $N_r = 200$ random initial realizations. The insets show the distributions on a large scale including all local energies ϵ_i. The left and right panels show Cambridge and Oxford networks.

thermalization. However, voting is a local process that involves only direct neighbors, which seems to be not sufficient for the emergence of a global thermal distribution. The presence of a few energy branches is well visible from the data of Fig. 8 obtained at $T = 0$. This figure shows the dependence of the final fraction f_f of red nodes on their initial fraction f_i and the total initial energy $E_i = \sum_{m=1}^{N} \epsilon_m$ of the whole system corresponding to a chosen initial random configuration of spins. Most probably, these different branches prevent efficient thermalization of the system with only local spin flip procedure.

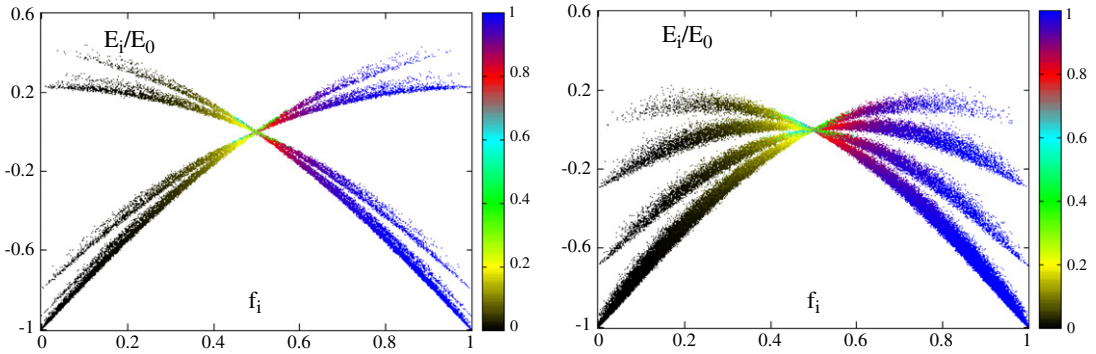


Fig. 8. (Color online) This diagram shows the final fraction of red nodes f_f, coded by color from $f_f = 0$ (black) to $f_f = 1$ (blue/dark gray), as a function of initial fraction of red nodes f_i and the total initial energy E_i; each of $N_r > 3.5 \times 10^4$ random realizations is shown by color point; data are shown after $t = 20$ time iterations at $T = 0$. The energy E_0 is the modulus of total energy with all spin up; here, $a = 0.5$. Left and right panels show data for Cambridge ($E_0 = 341.20$) and Oxford ($E_0 = 254.28$) networks; bars show color attribution to final probability f_f.

In addition to the above points, the asymmetric form of J_{ij} couplings plays an important role, generating a more complicated picture compared to the usual image of thermal relaxation (see, e.g., Ref. [35]). We also note that thermalization is absent in voter models on regular lattices [13].

4. PROF–Sznajd model

The Sznajd model [32] nicely incorporates the well-known trade union principle "United we stand, divided we fall" into the field of voter modeling and opinion formation on regular networks. A review of various aspects of this model is given in Ref. [13]. Here, we generalize the Sznajd model to include in it the features of the PROF model, and consider it on social networks with their scale-free structure. This gives us the PROF–Sznajd model, which is constructed in the following way. For a given network, we determine the PageRank probability $P(K_i)$ and the PageRank index K_i for all i nodes. We introduce the definition of a *group* of nodes. A group of nodes is defined by the following rule applied at each time step τ.

- (i) Pick by random a node i in the network and consider the polarization of the $N_g - 1$ highest PageRank nodes pointing to it.
- (ii) If node i and all other $N_g - 1$ nodes have the same color (same spin polarization), then these N_g nodes form a group whose effective PageRank value is the sum of all the member values $P_g = \sum_{j=1}^{N_g} P_j$; if this is not the case, then we leave the nodes unchanged and perform the next time step.
- (iii) Consider all the nodes *pointing to any member of the group* (this corresponds to model *option* 1) or consider *all the nodes pointing to any member of the group and all the nodes pointed by any member of the group* (this corresponds to model *option* 2); then check all these nodes n directly linked to the group: if an individual node PageRank value P_n is less than P_{group} then this node joins the group by taking the same color (polarization) as the group nodes; if this is not the case, then the node is left unchanged; the PageRank values of added nodes are then added to the group PageRank P_{group} and the group size is increased.

The above time step is repeated many times during time τ, counting the number of steps, by choosing a random node i on each next step. This procedure effectively corresponds to the zero-temperature case in the PROF model.

A typical example of the time evolution of the fraction of red nodes $f(\tau)$ in the PROF–Sznajd model is shown in Fig. 9. It shows that the system converges to a steady state after a time scale $\tau_c \approx 10N$ that is comparable with the convergence times for the PROF models studied in previous sections. We see that there are still some fluctuations in the steady-state regime which are visibly smaller for the option 2 case. We attribute this to a larger number of direct links in this case. The number of group nodes N_g gives some variation of f_f, but these variations remain on a relatively small scale of a few percent. Here, we should point on the important difference between the PROF and PROF–Sznajd models: for a given initial color realization, in the first case we have convergence to a fixed state after some convergence time, while in the second case we have convergence to a steady state which continues to fluctuate in time, keeping the color distribution only on average.

The dependence of the final fraction of red nodes f_f on its initial value f_i is shown by the density plot of probability W_f in Fig. 10 (option 1 of the PROF–Sznajd model). The probability W_f is obtained from many initial random realizations in a similar way to the case of Fig. 3. We see that there is a significant difference compared to the PROF model (Fig. 3): now even at small values of f_i we find small but finite values of f_f, while in the PROF model the red color disappears at $f_i < f_c$. This feature is related to the essence of the Sznajd model: here, even small groups can resist the totalitarian opinion. Other features of Fig. 10 are similar to those found for the PROF model: we again observe bistability of opinion formation. The number of nodes N_g, which form the group, does not significantly affect the distribution W_f: we have smaller fluctuations at larger N_g values but the model already works in a stable way at $N_g = 3$. The results for option 2 of the PROF–Sznajd model are shown in Fig. 11. In this case, the opinions with a small initial fraction of red nodes f_i are suppressed in a significantly

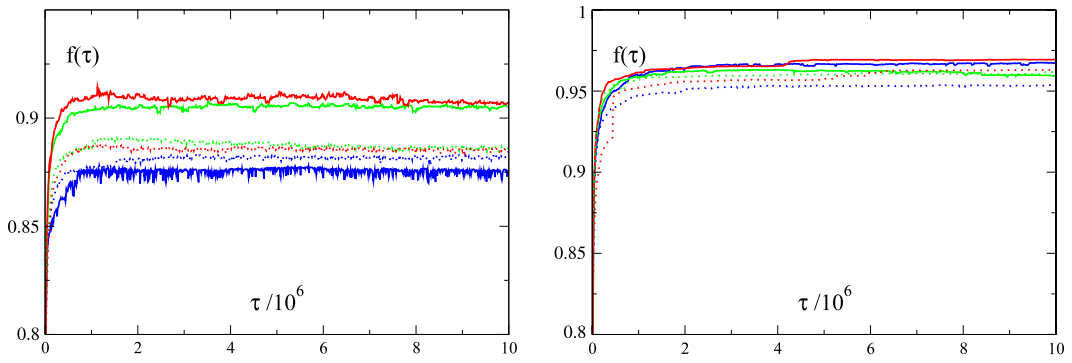


Fig. 9. (Color online) Time evolution of the fraction of red nodes $f(\tau)$ in the PROF–Sznajd model with the initial fraction of red nodes $f_i = 0.7$ at one random realization. The curves show data for three values of group size $N_g = 3$ (blue/black); 8 (green/light gray); and 13 (red/gray). Full/dashed curves are for Cambridge/Oxford networks; the left panel is for option 1; the right panel is for option 2.

stronger way compared to option 1. We attribute this to the fact that large groups can suppress small groups in a stronger way, since the outgoing direct links are taken into account in this option.

The significant difference between the two options of the PROF–Sznajd model is well seen from the data of Fig. 12. Here, all N_{top} nodes are taken in red (compare with the PROF model in Fig. 4). For option 1, the society elite succeeds in imposing its opinion on a significant fraction of nodes, which is increased by a factor 5–10. Visibly, this increase is less significant than in the PROF model. However, for option 2 of the PROF–Sznajd model there is practically no increase of the fraction of red nodes. Thus, in option 2 the society members are very independent and the influence of the elite on their opinion is very weak.

5. PROF models on the LiveJournal network

Even if one can expect that the properties of university networks are similar to those of real social networks, it is important to analyze the previous PROF models in the frame of a real social network. For that we use the LiveJournal network, collected, described, and presented in Ref. [17]. From this database we obtain a directed network with $N = 3577166$ nodes and $N_\ell = 44913072$ links, which are mainly directed (only about 30% of links are symmetric). The Google matrix of the network is constructed in the usual way [20], and its PageRank vector is determined by the iteration process at damping factor $\alpha = 0.85$. For the time evolution of fraction of red nodes f we use time iterations in t and τ defined as in previous sections.

The PageRank probability decay $P(K)$ is shown in Fig. 13. It is well described by an algebraic law $P(K) \propto 1/K^\beta$ with $\beta = 0.448 \pm 0.000046$. The convergence of a fraction of red nodes $f(t)$ takes place approximately on the same convergence time scale $t_c \sim 5 \sim O(1)$ even though the size of the network is increased almost by a factor 20.

In a way similar to the university networks we find that the homogeneous opinion of the society elite presented in a small fraction of N_{top} nodes influences a large fraction of the whole society especially when the parameter a is not very large (see Fig. 14 in comparison with Fig. 4). The influence of the elite at 1% of red nodes is larger in the case of the LiveJournal network. It is possible that this is related to a 30% larger number of links, but it is also possible that other structural network parameters also play a role here.

In spite of certain similarities with the previous data for university networks discussed before, we find that the opinion diagram for the LiveJournal network (see Fig. 14 right panel) is very different from those obtained for the university networks (see Fig. 3): the bistability has practically disappeared. We think that this difference originates from a significantly slower decay exponent for PageRank probability $P(K)$ in the case of LiveJournal. To check this assumption we compare the probability distribution W_f of final opinion f_f for an initial opinion fixed at $f_i = 0.4$ using the PROF model with the usual linear weight P in Eq. (1) and a quadratic weight proportional to P^2 (see Fig. 15). For the linear weight, we find that only very small values of $f_f \approx 0.005$ can be found for initial $f_i = 0.4$, while for the quadratic weight we obtain a rather broad distribution of f_f values in the main range $0 < f_f < 0.15$ with a few large values $f_f \approx 0.6$. Thus we see that the final opinion is rather sensitive to the weight used in Eq. (1). However, in contrast to the university networks (see Figs. 3 and 5), where we have narrow one-peak or double-peak distributions of f_f, for the LiveJournal network with quadratic weight we find a rather broad distribution of f_f. In the spirit of a renormalization map description considered in Ref. [10] (see Figs. 1, 2 there), it is possible to assume that one or two peaks corresponds to one or two fixed point attractors of the map. We make a conjecture that a broad distribution as in Fig. 15 (right panel) can correspond to a regime of a strange chaotic attractor appearing in the renormalization map dynamics. In principle, such a chaotic renormalization dynamics is known to appear in coupled spins lattices when three-spin couplings are present (see Ref. [38] and the references therein). It is possible that the presence of weight probability associated with the PageRank in a certain power may lead to chaotic dynamics which would generate a broad distribution of final opinions f_f.

We also made tests for the PROF–Sznajd model (option 1) for the LiveJournal database. However, in this case, at $f_i = 0.4$ and $a = 0.5$, we found only small f_f values (similar of those in Fig. 15, left panel) both for linear and quadratic weights in Eq. (1). It is possible that the Sznajd groups are less sensitive to the probability weight.

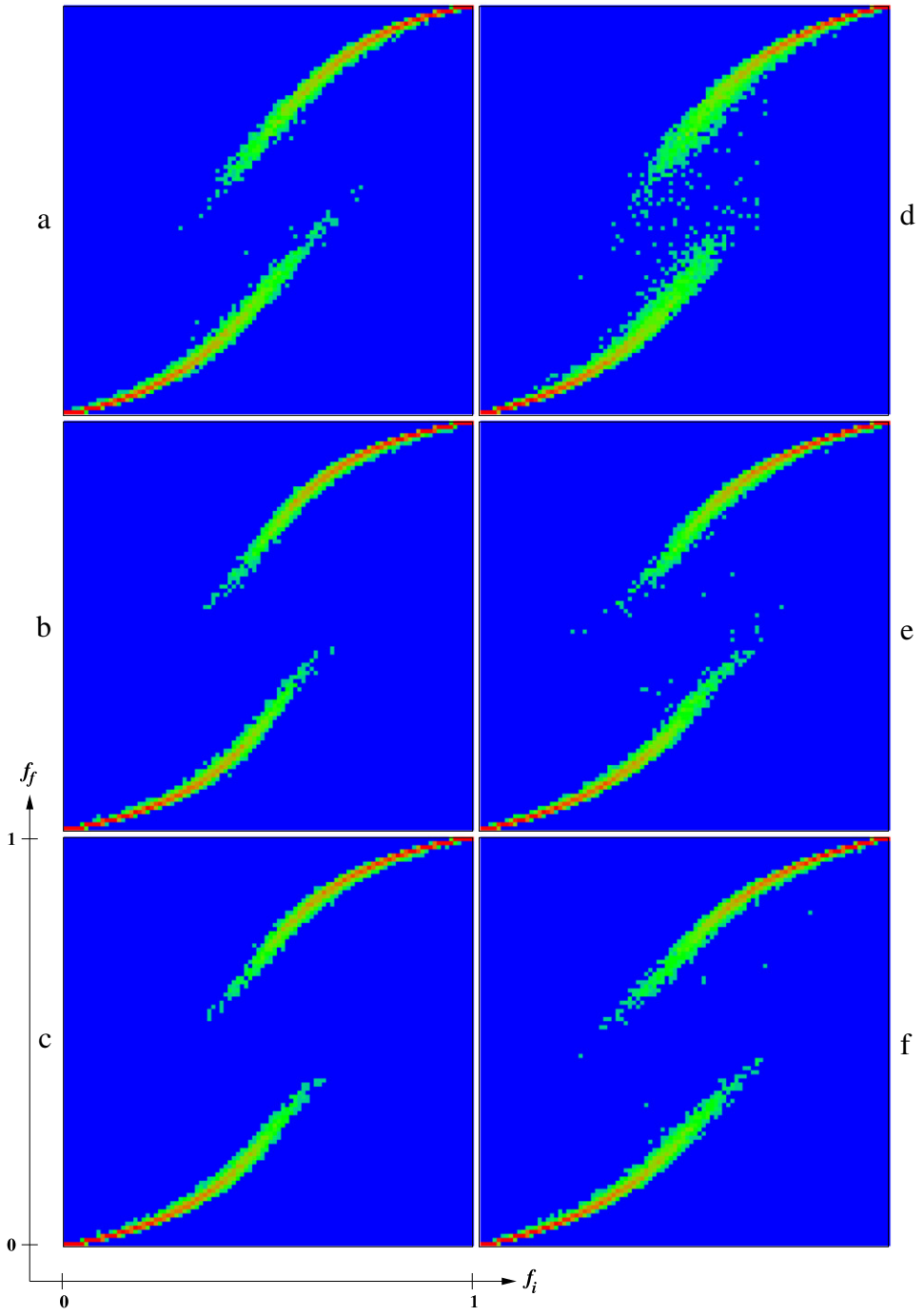


Fig. 10. (Color online) PROF–Sznajd model, option 1: density plot of probability W_f to find the dependence of the final red fraction f_f, shown on the y-axis, on the initial red fraction f_i, shown on the x-axis; data are shown inside the unit square $0 \leq f_i, f_f \leq 1$. The values of W_f are defined as the relative number of realizations found inside each of 100×100 cells which cover the whole unit square. Here, $N_r = 10^4$ realizations of randomly distributed colors are used to obtained W_f values; for each realization the time evolution is followed up to the convergence time with up to $\tau = 10^7$ steps. *Left column:* Cambridge network (a, b, c); *right column:* Oxford network (d, e, f); here, $N_g = 3$ (a, d), 8 (b, e), 13 (c, f) from top to bottom. The probability W_f is proportional to color changing from zero (blue/black) to unity (red/gray).

6. PROF models for the Twitter dataset

We also analyzed the opinion formation on the Twitter dataset with $N = 41\,652\,230$, $N_\ell = 1468\,365\,182$ taken from Ref. [18]. This is the entire size of Twitter at the corresponding moment of time [18]. The size is rather large, and due to that we present only the main features of the PROF model for this directed network.

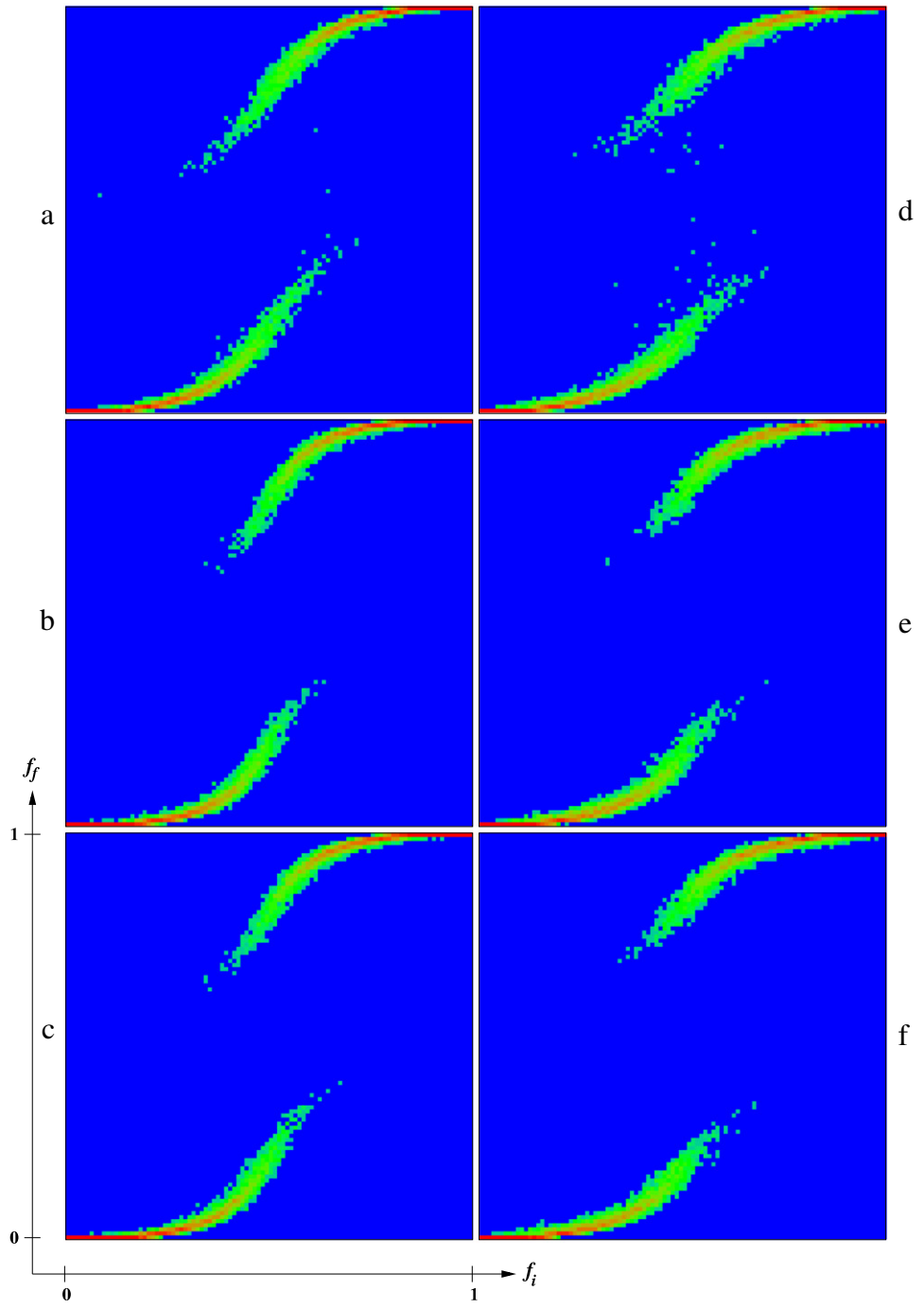


Fig. 11. (Color online) The same as in Fig. 10 but for PROF–Sznajd model, option 2.

The dependence of PageRank P on its index K is shown in Fig. 16 (left panel). For the range $1 \leq \log_{10} K \leq 5.5$, we find that the decay exponent $\beta \approx 0.51$, being similar to that of the LiveJournal network (see Fig. 13) even if there is a faster drop of P at larger K values. We note that the value $\beta \approx 0.5$ is rather different from the value usually found for the Zipf law [39] and the WWW [20], with $\beta \approx 1$. It is possible that this is related to a significantly larger average number of links per node, which is increased by a factor 3.5 for the Twitter network compared to the university networks analyzed in the previous sections.

The effect of the homogeneous elite opinion of all red N_{top} nodes is shown in Fig. 16 (right panel). We see that on the Twitter network a small fraction of elite with fixed opinion ($N_{top}/N \approx 3 \times 10^{-5}$) can impose this opinion on practically

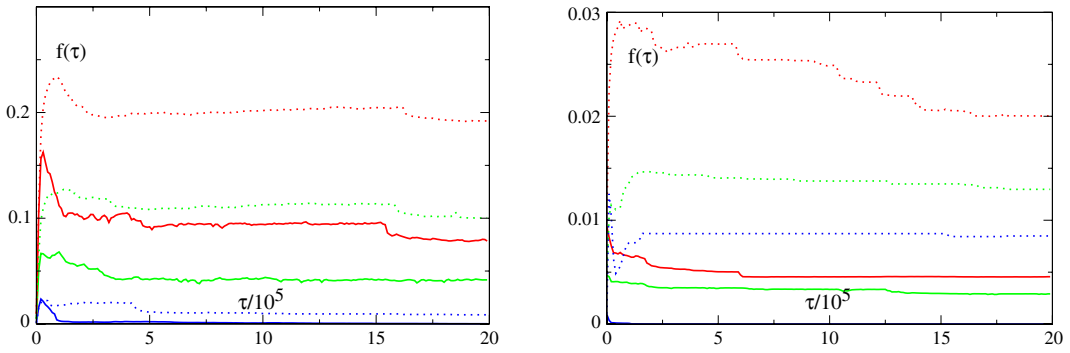


Fig. 12. (Color online) Time evolution of the fraction of red nodes $f(\tau)$ in the PROF–Sznajd model with the initial red nodes for the top PageRank nodes: $N_{top} = 200$ (blue/black); 1000 (green/light gray); 2000 (red/gray); here, $N_g = 8$. Full/dashed curves are for Cambridge/Oxford networks; the left panel is for option 1; the right panel is for option 2. The color of curves is red, green, blue, from top to bottom at maximal τ on both panels.

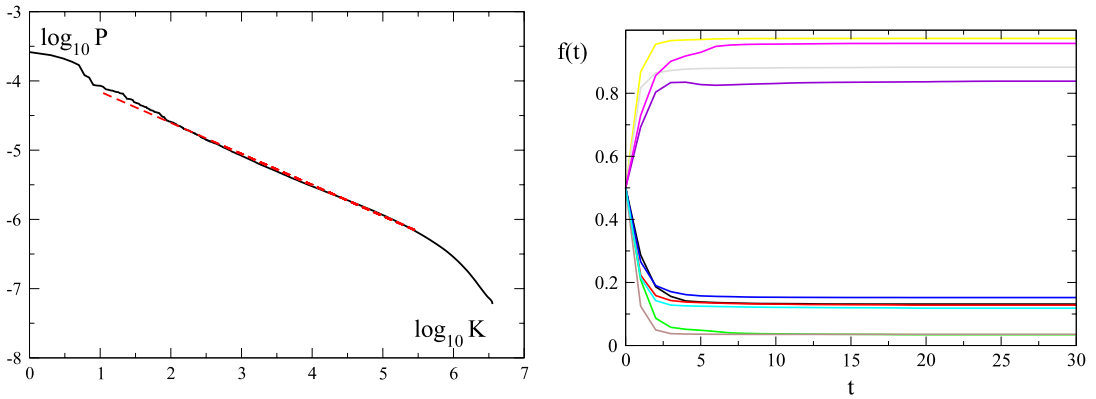


Fig. 13. (Color online) Data for the LiveJournal network. *Left panel:* PageRank probability decay with PageRank index K (full curve); the fitted algebraic dependence is shown by the dashed line $y = b - \beta x$ (for $1 \leq \log_{10} K \leq 5.5$) with the exponent $\beta = 0.448 \pm 0.000046$ and $b = -3.70 \pm 0.00023$. *Right panel:* time evolution of opinion given by a fraction of red nodes $f(t)$ as a function of number of iterations t (cf. Fig. 1) at $a = 0.5$; a few random initial realizations with $f_i = 0.5$ are shown.

the whole community for all values of the conformist parameter $1 - a$. We find that for $N_{top} > 1300$ all f_f values are very close to unity, while for $N_{top} < 1200$ we find $f_f = 0$, as is seen in Fig. 16, right panel. Thus, the transition is very sharp. We attribute such a strong influence of elite opinion to the very connected structure of Twitter network with a significantly larger average number of links per node compared to the university and LiveJournal networks.

At $a = 0.5$, for a fixed fraction of initial opinion $f_i = 0.4$, we find that the probability distribution W_f of final opinion f_f is located in the range of small values $0.0006 < W_f < 0.0007$ for both the linear P and quadratic P^2 weights used in Eq. (1) (we do not show these data). For the linear weight, the situation is rather similar to the case of LiveJournal (see Fig. 15), but for the quadratic weight we find a significant difference between the two networks (see Fig. 15). The reason for such a significant difference for the quadratic weight case requires a more detailed comparison of network properties.

The large size of the Twitter network makes numerical simulations of the PROF–Sznajd model rather difficult, and therefore we did not study this model for this network.

7. Discussion

In this work we have proposed the PageRank model of opinion formation of social networks and analyzed its properties on examples of four different networks. For two university networks we find rather similar properties of opinion formation. Opinion formation is characterized by an important feature according to which the society elite with a fixed opinion can impose its opinion on a significant fraction of the society members which is much larger than the initial elite fraction. However, when the initial opinions of society members, including the elite, are presented by two options, then we find a significant range of opinion fraction within a bistability regime. This range depends on the conformist parameter, which characterizes the local aspects of opinion formation of linked society members. The generalization of the Sznajd model for scale-free social networks gives interesting examples of opinion formation where finite small-size groups can keep their

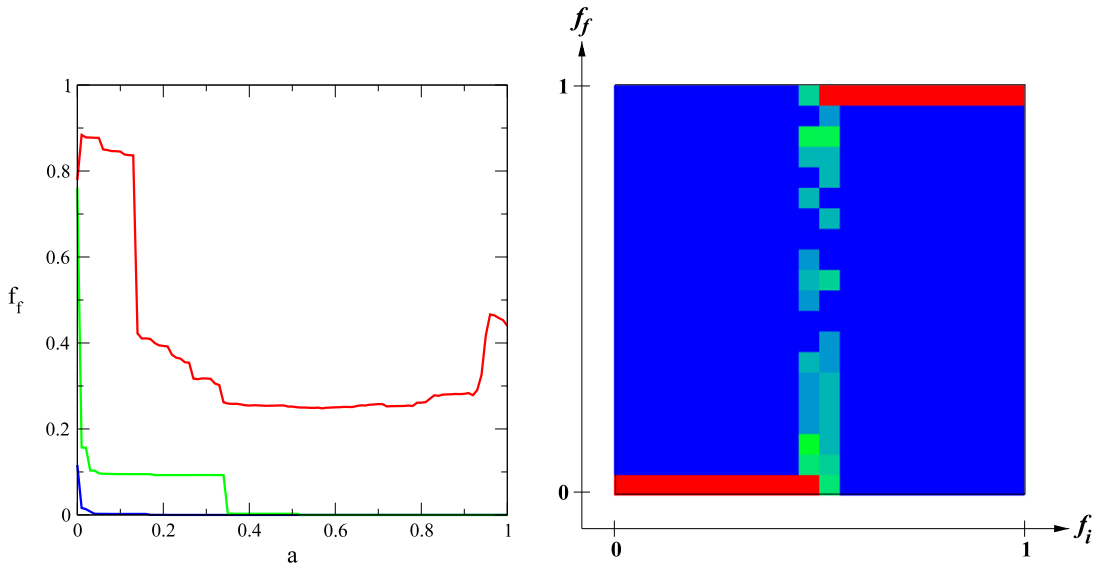


Fig. 14. (Color online) Data for the LiveJournal network. *Left panel:* dependence of the final fraction of red nodes f_f on the tenacious parameter a (or conformist parameter $b = 1 - a$) in the PROF model for initial red nodes in N_{top} values of the PageRank index ($1 \le K \le N_{top}$; cf. Fig. 4). Here, $N_{top} = 2000$ blue, 10,000 green, and 35,000 red curves (from bottom to top at $a = 0.5$); $T = 0$. *Right panel:* the same data as in Fig. 3 at $a = 0.5$ with the same parameters but for the LiveJournal network.

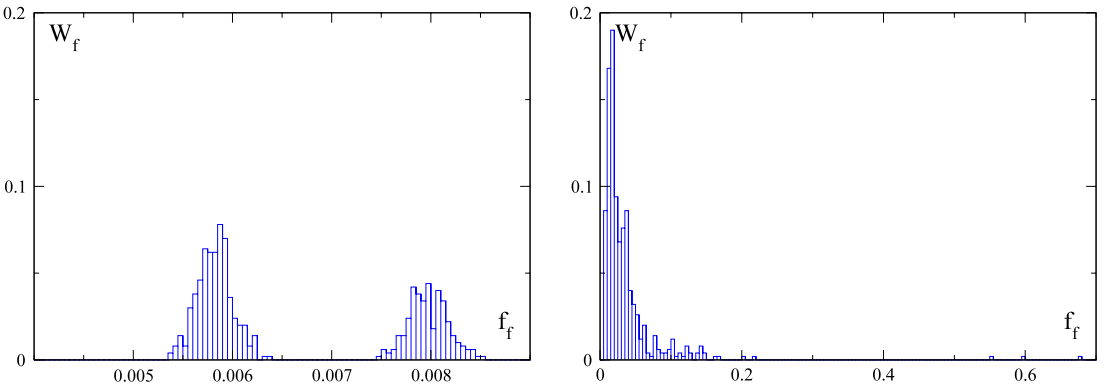


Fig. 15. (Color online) Data for the LiveJournal network: probability distribution W_f of final opinion f_f for a fixed initial opinion $f_i = 0.4$ and $a = 0.5$ in the PROF model. *Left panel:* usual linear weight $P(K)$ in Eq. (1). *Right panel:* a quadratic weight $P^2(K)$ in Eq. (1). Histograms are obtained with $N_r = 500$ initial random realizations; the normalization is fixed by the condition that the sum of W_f over all histogram bins is equal to unity.

own opinion, being different from the main opinion of the majority. In this way, the proposed PROF–Sznajd model shows that totalitarian opinions can be escaped from by small subcommunities. We find that the properties of opinion formation are rather similar for the two university networks of Cambridge and Oxford. However, the results obtained for networks of LiveJournal and Twitter show that the range of bistability practically disappears for these networks. Our data indicate that this is related to a slower algebraic decay of PageRank in these cases compared to the university networks. However, the deep reasons for such a difference require a more detailed analysis. Indeed, the LiveJournal and Twitter networks demonstrate rather different behavior for the P^2-weighted function of opinion formation. The studies performed for regular networks [10] show the existence of stable or bistable fixed points for opinion formation models that have certain similarities with the opinion formation properties found in our studies. At the same time the results obtained in Ref. [38] show that three-body spin coupling can generate a chaotic renormalization dynamics. Some of our results (Fig. 15, right panel) give indications of the possible existence of such a chaotic phase in social networks.

The enormous development of social networks in the last few years [2–5] definitely shows that the analysis of opinion formation on such networks requires further investigations. This research can find also various other applications. One of them could be a neuronal network of a brain which represents itself as a directed scale-free network [40]. The applications of network science to brain networks is now under rapid development (see, e.g., Ref. [41]), and Google matrix methods can find useful applications in this field [42].

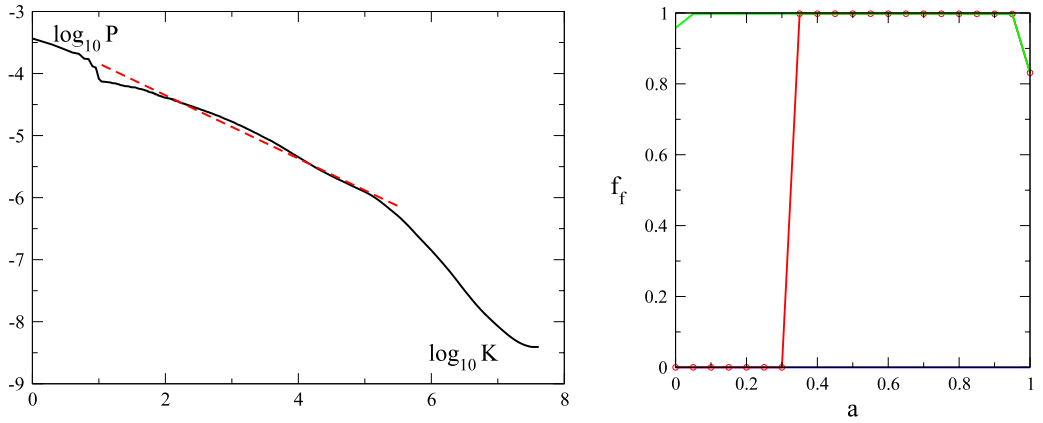


Fig. 16. (Color online) Data for the Twitter network. *Left panel:* PageRank probability decay with PageRank index K (full curve); the fitted algebraic dependence is shown by the dashed line $y = b - \beta x$ (for $1 \leq \log_{10} K \leq 5.5$) with the exponent $\beta = 0.511 \pm 0.0021$ and $b = -3.33 \pm 0.0069$ (for the range $5.5 \leq \log_{10} K \leq 7$ we find $\beta = 1.23$). *Right panel:* dependence of the final fraction of red nodes f_f on the tenacious parameter a (or conformist parameter $b = 1 - a$) in the PROF model for initial red nodes in N_{top} values of PageRank index ($1 \leq K \leq N_{top}$; cf. Fig. 4, Fig. 14). Here, $N_{top} = 1200$ (blue line at $f_f = 0$); 1250 (red curve with circles); and 1300 (top green line); $T = 0$.

Acknowledgments

This work is supported in part by the EC FET Open project "New tools and algorithms for directed network analysis" (NADINE No. 288956). We thank A. Benczúr and S. Vigna for providing us with friendly access to the LiveJournal database [17] and the Twitter dataset [18]. We also thank the France–Armenia collaboration grant CNRS/SCS No 24943 (IE-017) on "Classical and quantum chaos".

References

[1] J.R. Zaller, The Nature and Origins of Mass Opinion, Cambridge University Press, Cambridge, UK, 1999.
[2] Wikipedia, LiveJournal, March 9, 2012. http://en.wikipedia.org/wiki/LiveJournal.
[3] Wikipedia, Facebook, March 9, 2012. http://en.wikipedia.org/wiki/Facebook.
[4] Wikipedia, Twitter, March 9, 2012. http://en.wikipedia.org/wiki/Twitter.
[5] Wikipedia, VK (Social network), March 9, 2012. http://en.wikipedia.org/wiki/VK_(social_network).
[6] S.N. Dorogovtsev, J.F.F. Mendes, Evolution of Networks, Oxford Univ. Press, 2003.
[7] G. Caldarelli, Scale-Free Networks, Oxford Univ. Press, 2007.
[8] S. Galam, J. Math. Psych. 30 (1986) 426.
[9] T.M. Liggett, Stochastic Interacting Systems: Contact, Voter and Exclusion Processes, Springer, Berlin, 1999.
[10] S. Galam, Europhys. Lett. 70 (2005) 705.
[11] D.J. Watts, P.S. Dodds, J. Consumer Research 34 (4) (2007) 441.
[12] S. Galam, Int. J. Mod. Phys. C 19 (2008) 409.
[13] C. Castellano, S. Fortunato, V. Loreto, Rev. Mod. Phys. 81 (2009) 591.
[14] P.L. Krapivsky, S. Redner, E. Ben-Naim, A Kinetic View of Statistical Physics, Cambridge University Press, Cambridge, UK, 2010.
[15] B. Schmittmann, A. Mukhopadhyay, Phys. Rev. E 82 (2010) 066104.
[16] Academic Web Link Database Project. http://cybermetrics.wlv.ac.uk/database/.
[17] M. Kurucz, A.A. Benczúr, A. Pereszlenyi, Large-scale principal component analysis on livejournal friends network, in: Proc. Workshop on Social Network Mining and Analysis Held in Conjunction with 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008, Las Vegas NV, August 24–27, 2008. http://dms.sztaki.hu/en/letoltes/livejournal-data.
[18] H. Kwak, C. Lee, H. Park, S. Moon, What is Twitter, a social network or a news media? in: Proc. 19th Int. Conf. WWW2010, ACM, New York, NY, 2010, p. 591. the web data are downloaded from the web site maintained by S. Vigna. http://law.dsi.unimi.it/webdata/twitter-2010/.
[19] S. Brin, L. Page, Comput. Netw. ISDN Syst. 30 (1998) 107.
[20] A.M. Langville, C.D. Meyer, Google's PageRank and Beyond: The Science of Search Engine Rankings, Princeton University Press, Princeton, 2006.
[21] S. Redner, Phys. Today 58 (6) (2005) 49.
[22] F. Radicchi, S. Fortunato, B. Markines, A. Vespignani, Phys. Rev. E 80 (2009) 056103.
[23] J.D. West, T.C. Bergstrom, C.T. Bergstrom, Coll. Res. Lib. 71 (2010) 236. http://www.eigenfactor.org/.
[24] F. Radicchi, PLoS ONE 6 (2011) e17249.
[25] A.O. Zhirov, O.V. Zhirov, D.L. Shepelyansky, Eur. Phys. J. B 77 (2010) 523.
[26] L. Ermann, D.L. Shepelyansky, Acta Phys. Pol. A 120 (6A) (2011) A158.
[27] T. Preis, D. Reith, H.E. Stanley, Phil. Trans. R. Soc. A 368 (2010) 5707.
[28] T. Preis, H.S. Moat, H.E. Stanley, S.R. Bishop, Sci. Reports 2 (2012) 350.
[29] S. Saavedra, J. Duch, B. Uzzi, PLoS ONE 6 (2011) e26705.
[30] J. Ginsberg, M.H. Mohebbi, R.S. Patel, L. Brammer, M.S. Smolinski, L. Brilliant, Nature 457 (2009) 1012.
[31] A. Vespignani, Science 325 (2009) 425.
[32] K. Sznajd-Weron, J. Sznajd, Int. J. Mod. Phys. C 11 (2000) 1157.
[33] K.M. Frahm, B. Georgeot, D.L. Shepelyansky, J. Phys, A: Math. Theor. 44 (2011) 465101.
[34] L. Ermann, A.D. Chepelianskii, D.L. Shepelyansky, J. Phys. A: Math. Theor. 45 (2012) 275101.
[35] S. Galam, B. Walliser, Physica A 389 (2010) 481.
[36] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, J. Chem. Phys. 21 (1953) 1087.

[37] V. Sood, S. Redner, Phys. Rev. Lett. 94 (2005) 178701.
[38] N.S. Ananikian, S.K. Dallakian, Physica D 107 (1997) 75.
[39] G.K. Zipf, Human Behavior and the Principle of Least Effort, Addison-Wesley, Boston, 1949.
[40] V.M. Eguiluz, D.R. Chialvo, G.A. Cecchi, M. Baliki, A.V. Apkarian, Phys. Rev. Lett. 94 (2005) 018102.
[41] X.-N. Zuo, R. Ehmke, M. Mennes, D. Imperati, F.X. Castellanos, O. Sporns, M.P. Milham, Cereb. Cortex, 2011. http://dx.doi.org/10.1093/cercor/bhr269.
[42] D.L. Shepelyansky, O.V. Zhirov, Phys. Lett. A 374 (2010) 3206.

EPJ B
Condensed Matter and Complex Systems
EPJ.org
your physics journal

Eur. Phys. J. B (2013) 86: 193 DOI: 10.1140/epjb/e2013-31090-8

Spectral properties of Google matrix of Wikipedia and other networks

Leonardo Ermann, Klaus M. Frahm and Dima L. Shepelyansky

edp sciences
Società Italiana di Fisica
Springer

Eur. Phys. J. B (2013) 86: 193
DOI: 10.1140/epjb/e2013-31090-8

THE EUROPEAN PHYSICAL JOURNAL B

Regular Article

Spectral properties of Google matrix of Wikipedia and other networks

Leonardo Ermann[1,2], Klaus M. Frahm[2], and Dima L. Shepelyansky[2,a]

[1] Departamento de Física Teórica, GIyA, Comisión Nacional de Energía Atómica, 1429 Buenos Aires, Argentina
[2] Laboratoire de Physique Théorique du CNRS, IRSAMC, Université de Toulouse, UPS, 31062 Toulouse, France

Received 5 December 2012
Published online 29 April 2013 –

Abstract. We study the properties of eigenvalues and eigenvectors of the Google matrix of the Wikipedia articles hyperlink network and other real networks. With the help of the Arnoldi method, we analyze the distribution of eigenvalues in the complex plane and show that eigenstates with significant eigenvalue modulus are located on well defined network communities. We also show that the correlator between PageRank and CheiRank vectors distinguishes different organizations of information flow on BBC and Le Monde web sites.

1 Introduction

With the appearance of the world wide web (WWW) [1] the modern society created huge directed networks where the information retrieval and ranking of network nodes becomes a formidable challenge. The mathematical grounds of ranking of nodes are based one the concept of Markov chains [2] and related class of Perron-Frobenius operators naturally appearing in dynamical systems (see, e.g., [3]). A concrete implementation of these mathematical concepts to the ranking of WWW nodes was started by Brin and Page in 1998 [4]. It is significantly based on the PageRank algorithm (PRA) which became a fundamental element of the Google search engine broadly used by internet users [5].

Already in 1998, Brin and Page pointed out that *"despite the importance of large-scale search engines on the web, very little academic research has been done on them"* [4]. Since that time the academic studies have been concentrated mainly on the properties of the PageRank vector determined by the PRA (see, e.g., [5–8]). Of course, the PageRank vector is at the basis of ranking of network nodes but the whole description of a directed network is given by the Google matrix G. Thus, it is important to understand the properties of the whole spectrum of eigenvalues of Google matrix and to analyze the meaning and significance of its eigenstates. Certain spectral properties of G matrix have been analyzed in references [9–15]. Here, we concentrate our spectral analysis on the Wikipedia articles network studied in reference [16]. The advantage of this network is due to a clear meaning of nodes, determined by the titles of Wikipedia articles thus simplifying the understanding of information flow in this network. In addition to that, we analyze the statistical properties of eigenvalues and eigenstates of G for WWW networks of Cambridge University, Python, BBC and Le Monde crawled in March 2011.

The Google matrix elements of a directed network are defined as [4,5,17]:

$$G_{ij} = \alpha S_{ij} + (1-\alpha)/N, \tag{1}$$

where the matrix S_{ij} is obtained from an adjacency matrix A_{ij} by normalizing all nonzero columns to one ($\sum_i S_{ij} = 1$) and replacing columns with only zero elements by $1/N$ (*dangling nodes*) with N being the matrix size. For the WWW an element A_{ij} of the adjacency matrix is equal to unity if a node j points to the node i and zero otherwise. The damping parameter α in the WWW context describes the probability $(1-\alpha)$ to jump to any node for a random surfer. For WWW, the Google search engine uses $\alpha \approx 0.85$ [5]. The matrix G belongs to the class of Perron-Frobenius operators [5], its largest eigenvalue is $\lambda = 1$ and other eigenvalues have $|\lambda| \leq \alpha$. The right eigenvector at $\lambda = 1$, which is called the PageRank, has real nonnegative elements $P(i)$ and gives a probability $P(i)$ to find a random surfer at site i. Due to the gap $1-\alpha \approx 0.15$ between the largest eigenvalue and the other eigenvalues the PRA permits an efficient and simple determination of the PageRank by the power iteration method. Note that at $\alpha = 1$ the largest eigenvalue $\lambda = 1$ is typically highly degenerate due to many invariant subspaces which define many independent Perron-Frobenius operators which provide (at least) one eigenvalue $\lambda = 1$. This point and also a numerical method to determine the PageRank for the case $1-\alpha \ll 1$ are described in detail in reference [13].

Once the PageRank (at $\alpha = 0.85$) is found, all nodes can be sorted by decreasing probabilities $P(i)$. The node

[a] e-mail: dima@irsamc.ups-tlse.fr

Table 1. Parameters of all networks considered in the paper.

	N	N_ℓ	n_A
Wikipedia	3282257	71012307	3000
Cam. 2011	893176	15106706	4000
Python	541545	9031262	5000
BBC	319637	7278258	4000
Le Monde	134196	10621445	5000

rank is then given by index $K(i)$ which reflects the relevance of the node i. The top PageRank nodes are located at small values of $K(i) = 1, 2, \ldots$

In addition to a given directed network A_{ij}, it is useful to analyze an inverse network with inverted direction of links with elements of adjacency matrix $A_{ij} \to A_{ji}$. The Google matrix G^* of the inverse network is then constructed via corresponding matrix S^* according to the relations (1) using the same value of α as for the G matrix. The right eigenvector of G^* at eigenvalue $\lambda = 1$ is called CheiRank giving a complementary rank index $K^*(i)$ of network nodes [15,16,18–20]. It is known that the PageRank probability is proportional to the number of ingoing links characterizing how popular or known a given node is while the CheiRank probability is proportional to the number of outgoing links highlighting the node communicativity (see, e.g., [5–8,16,19]). The statistical properties of the node distribution on the PageRank-CheiRank plane are described in reference [19] for various directed networks.

The paper is composed as following: the spectrum of the Google matrix of various networks is analyzed in Section 2, statistical properties of eigenstates are discussed in Section 3, the communities related to Wikipedia eigenstates are examined in Section 4, the distribution of nodes in the PageRank-CheiRank plane is studied in Section 5, the link distribution over PageRank index is considered in Section 6, discussion of results is given in Section 7. An Appendix gives all parameters of the five directed networks considered here and describes in detail certain eigenvalues and eigenvectors.

2 Google matrix spectrum

We study the spectrum of eigenvalues of the Google matrix of five directed networks. For each network the number of nodes N and the number of links N_ℓ are given in Table 1 (see Appendix). The spectrum is obtained numerically using the powerful Arnoldi method described in [21–23]. The idea of the method is to construct a set of orthonormal vectors by applying the matrix (G, S, G^*, S^* or any other matrix of which we want to determine the largest eigenvalues) on some suitable normalized initial vector and orthonormalizing the result to the initial vector. Then the matrix is applied to the second vector and the result is orthonormalized to the first two vectors and so on. The used scalar products and normalization factors during the Gram-Schmidt process provide the matrix representation of the initial big matrix on the set of orthonormal vectors (which span a *Krylov space*) in a form of a Hessenberg matrix whose eigenvalues converge typically quite well versus the largest eigenvalues of the initial matrix even if the chosen number of orthonormal vectors, the Arnold dimension n_A, is quite modest (3000–5000 in this work) as compared to the initial matrix size.

Table 2. G and G^* eigespectrum parameters for all networks.

	N_s	N_d	$d_{\max}$	$N_{\rm circ.}$	N_1
Wikipedia	515	255	11	381	255
Wikipedia*	21198	5355	717	8968	5365
Cam. 2011	808	329	74	343	332
Cam. 2011*	186062	2039	5144	2044	2041
Python	198	23	72	26	23
Python*	1589	25	951	35	31
BBC	50	19	28	19	19
BBC*	39	28	6	28	28
Le Monde	83	64	18	64	64
Le Monde*	789	354	15	373	361

In this work, we are interested in the spectrum of the matrix $S = G(\alpha = 1)$ (or S^*) since the spectrum of $G(\alpha)$ (or $G^*(\alpha)$) is simply obtained by rescaling the complex eigenvalues with the factor α (apart from "one" largest eigenvalue $\lambda = 1$ which does not change).

The direct dionalization of the Google matrix G faces a number of numerical challenges. Thus, the highly degenerate unit eigenvalue $\lambda = 1$ of S creates convergence problems for the Arnoldi method. To resolve this numerical problem, we follow the approach developed in references [13,15] and follow the description given there. We first find the invariant isolated subsets. These subsets are invariant with respect to applications of S. We merge all subspaces with common members, and obtain a sequence of disjoint subspaces V_j of dimension d_j invariant by applications of S. The remaining part of nodes forms the wholly connected *core space*. Such a classification scheme can be efficiently implemented in a computer program and it provides a subdivision of network nodes in N_c core space nodes and N_s subspace nodes belonging to at least one of the invariant subspaces V_j inducing the block triangular structure of matrix S:

$$S = \begin{pmatrix} S_{ss} & S_{sc} \\ 0 & S_{cc} \end{pmatrix}, \qquad (2)$$

where S_{ss} is itself composed of many small diagonal blocks for each invariant subspace and whose eigenvalues can be efficiently obtained by direct ("exact") numerical diagonalization.

The total subspace size N_S, the number of independent subspaces N_d, the maximal subspace dimension $d_{\max}$ and the number N_1 of S eigenvalues with $\lambda = 1$ are given in Table 2. The spectrum and eigenstates of the core space S_{cc} are determined by the Arnoldi method with Arnoldi dimension n_A giving the eigenvalues λ_i of S_{cc} with largest modulus and the corresponding eigenvectors ψ_j ($G\psi_i = \lambda_i\psi_i$). The values of n_A we used for the different networks are given in Table 1. According to Table 2, we have the average number of links per node $\zeta_\ell \approx 21.63$ (Wikipedia),

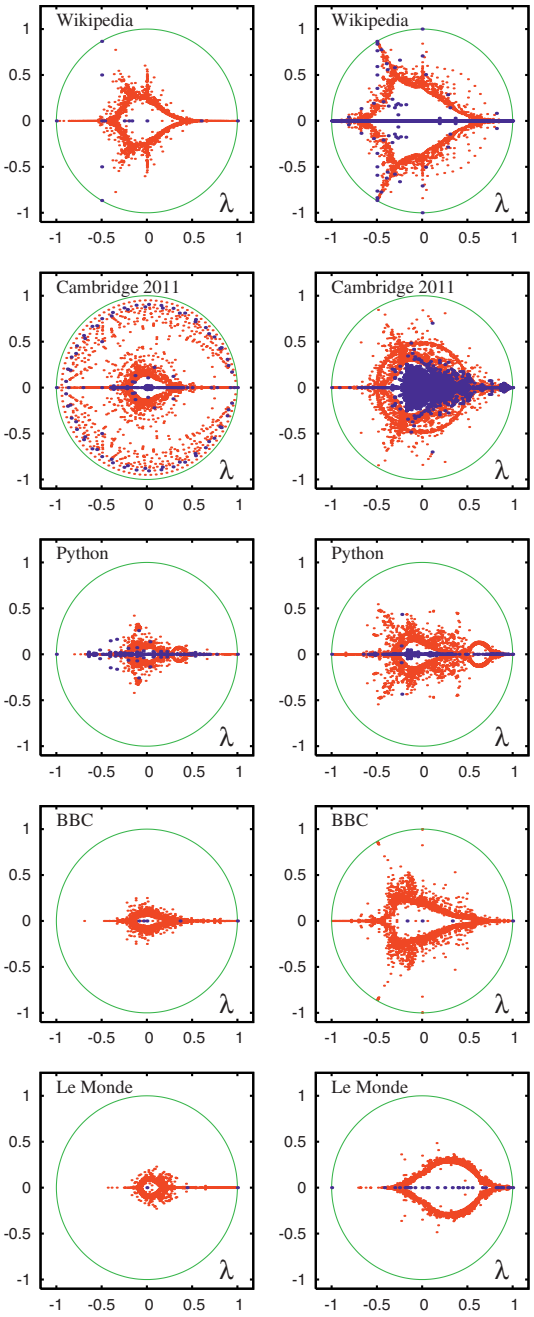


Fig. 1. Spectrum of eigenvalues λ the Google matrices G (left column) and G^* (right column) for Wikipedia, Cambridge 2011, Python, BBC and Le Monde ($\alpha = 1$). Red dots are core space eigenvalues, blue dots are subspace eigenvalues and the full green curve shows the unit circle. The core space eigenvalues were calculated by the projected Arnoldi method with Arnoldi dimensions n_A as given in Table 1.

16.91 (Cambridge 2011), 16.67 (Python), 22.77 (BBC), 79.14 (Le Monde).

The distributions of subspaces eigenvalues and largest n_A eigenvalues of the core space are shown in Figure 1 in the complex plane λ for all five networks. The blue points show the eigenvalues of isolated subspaces. We note that their number is relatively small compared to those of

Table 3. Eigenvalues of eigenvectors shown in Figures 1 and 2 by corresponding colors. Index m of λ_m numbers eigenvalues in the decreasing order of $|\lambda|$ in the core space.

	Color	Eigenvalue
Wikipedia	red	$\lambda_1 = 0.999987$
	green	$\lambda_2 = 0.977237$
	blue	$\lambda_{52} = -0.35003 + i\,0.77374$
	pink	$\lambda_{864} = -0.34293 + i\,0.43145$
Wikipedia*	red	$\lambda_1 = 0.999982$
	green	$\lambda_2 = 0.999902$
	blue	$\lambda_{662} = 0.0000000 + i\,0.84090$
	pink	$\lambda_{38} = -0.49626 + i\,0.85653$
Cam. 2011	red	$\lambda_1 = 0.999749$
	green	$\lambda_2 = 0.999270$
	blue	$\lambda_{350} = 0.41779 + i\,0.77856$
	pink	$\lambda_{144} = -0.52909 + i\,0.78693$
Cam. 2011*	red	$\lambda_1 = 0.999998$
	green	$\lambda_2 = 0.999994$
	blue	$\lambda_{765} = 0.24846 + i\,0.80915$
	pink	$\lambda_{249} = -0.48736 + i\,0.84568$
Python	red	$\lambda_1 = 0.999975$
	green	$\lambda_2 = 0.998864$
	blue	$\lambda_{3315} = 0.14484 + i\,0.19215$
	pink	$\lambda_{1337} = -0.14427 + i\,0.42051$
Python*	red	$\lambda_1 = 0.999995$
	green	$\lambda_2 = 0.999991$
	blue	$\lambda_{2559} = 0.37694 + i\,0.45231$
	pink	$\lambda_{3076} = 0.12214 + i\,0.47416$
BBC	red	$\lambda_1 = 0.99883$
	green	$\lambda_2 = 0.99251$
	blue	$\lambda_{1276} = -0.12414 + i\,0.24795$
	pink	$\lambda_{1148} = -0.22459 + i\,0.20024$
BBC*	red	$\lambda_1 = 0.999999$
	green	$\lambda_2 = 0.999994$
	blue	$\lambda_{16} = -0.00067 + i\,0.99930$
	pink	$\lambda_{90} = -0.49635 + i\,0.85848$
Le Monde	red	$\lambda_1 = 0.998837$
	green	$\lambda_2 = 0.983123$
	blue	$\lambda_{926} = 0.10295 + i\,0.22890$
	pink	$\lambda_{1118} = 0.08023 + i\,0.20595$
Le Monde*	red	$\lambda_1 = 0.999999$
	green	$\lambda_2 = 0.999959$
	blue	$\lambda_{2093} = 0.15987 + i\,0.48502$
	pink	$\lambda_{2474} = 0.17637 + i\,0.40917$

British University networks [24] (up to year 2006) analyzed in reference [13]. We attribute this to a larger number of ζ_ℓ links per node that reduces an effective size of isolated parts of network. Between 2006 and 2011, especially for Cambridge, it seems that the increased use of PHP and similar web software tends to considerably increase the value of ζ_ℓ. Indeed, we have $\zeta_\ell \approx 10$ for university networks up to 2006 [13] which used less this kind of PHP software. In Figure 1 the red points show n_A eigenvalues of the core space with largest $|\lambda|$. Due to finite n_A value there is an empty white space around $\lambda = 0$. There is no significant gap for core eigenvalues since λ_1 is rather close to 1 (see Tab. 3).

In global, we can say that the structure of the Wikipedia spectrum of S and S^* is somewhat similar to

those of Cambridge 2006 (see Fig. 2 in Ref. [13]). For Cambridge 2011, the spectrum of S is drastically changed compared to the year 2006 but for S^* certain features remain common both for 2006 and 2011 (e.g., a circle $|\lambda| \approx 0.5$, triplet-star). For Python, BBC and Le Monde the imaginary parts $\mathrm{Im}(\lambda)$ of eigenvalues of S are relatively small compared to the networks of Wikipedia and Cambridge. We suppose that there are less symmetric links in the later cases. It is interesting that for S^* of Python, BBC and Le Monde the imaginary parts $\mathrm{Im}(\lambda)$ are significantly larger than for S.

The origin of nontrivial structures of the spectrum of G and G^* for directed networks discussed here and in references [11–13,15] still require detailed analysis. We note that well visible triplet and cross structures (see, e.g., Wikipedia spectrum in Fig. 1 and Fig. 2 of [13]) naturally appear in the spectra of random unistochastic matrices of size $N = 3$ and 4, which have been analyzed analytically and numerically in reference [25]. In view of this similarity, we suppose that networks with such structures have some triplet or quartet subgroup of nodes weakly coupled to the rest of the network. However, a detailed understanding of the spectrum requires a deeper analysis. In the next section, we turn to a study of eigenstate properties.

3 Statistical properties of eigenstates

The dependence of PageRank P and CheiRank P^* vectors on their indexes K and K^* at $\alpha = 0.85; 1 - 10^{-8}$ are shown in Figure 2. At $\alpha = 0.85$, we have an approximate algebraic decay of probability according to the Zipf law $P \sim 1/K^{\beta}, P^* \sim 1/K^{*\beta}$ (see, e.g., [14] and references therein). We find the following values β for PageRank (CheiRank): $0.96 \pm 0.002\,(0.73 \pm 0.003)$ Wikipedia; $0.81 \pm 0.007\,(0.90 \pm 0.004)$ Cambridge 2011; $1.12 \pm 0.01\,(1.17 \pm 0.006)$ Python; $1.20 \pm 0.006\,(0.96 \pm 0.004)$ BBC; $1.08 \pm 0.009\,(0.55 \pm 0.002)$ Le Monde. Formally, the statistical errors in β are relatively small but in some cases there are variations of slope in the decay of PageRank (CheiRank) probability that gives a dependence of β on a fitting range (e.g., that is why β here is a bit different from its values for Wikipedia given in Ref. [16]). We note that the value $\beta \approx 1$ for the PageRank remains relatively stable to all networks corresponding to the usual exponent $\mu \approx 2.1$ of algebraic decay of the ingoing link distribution leading to $\beta = 1/(\mu - 1) \approx 0.9$ (see, e.g., [6,7,14–16]).

For CheiRank the variations of β from one network to another are more significant being in agreement with the fact that for outgoing links the exponent $\mu \approx 2.7$ varies in a more significant manner.

For $\alpha = 1 - 10^{-8}$, we find that the main probability of PageRank and CheiRank eigenvectors is located on isolated subspaces with N_s nodes; after that value there is a significant drop of probability for $K, K^* > N_s$. This effect was already found and explained in detail in reference [13] and our new data confirm that it is indeed rather generic.

The modulus of four eigenfuctions $|\psi_i(j)|$ from the core space are shown in Figure 2 by color curves as a function of their own index K_i which order $|\psi_i(j)|$ in a monotonic decreasing order. For Python, BBC and Le Monde the decay of $|\psi_i(j)|$ with K_i is similar to the decay

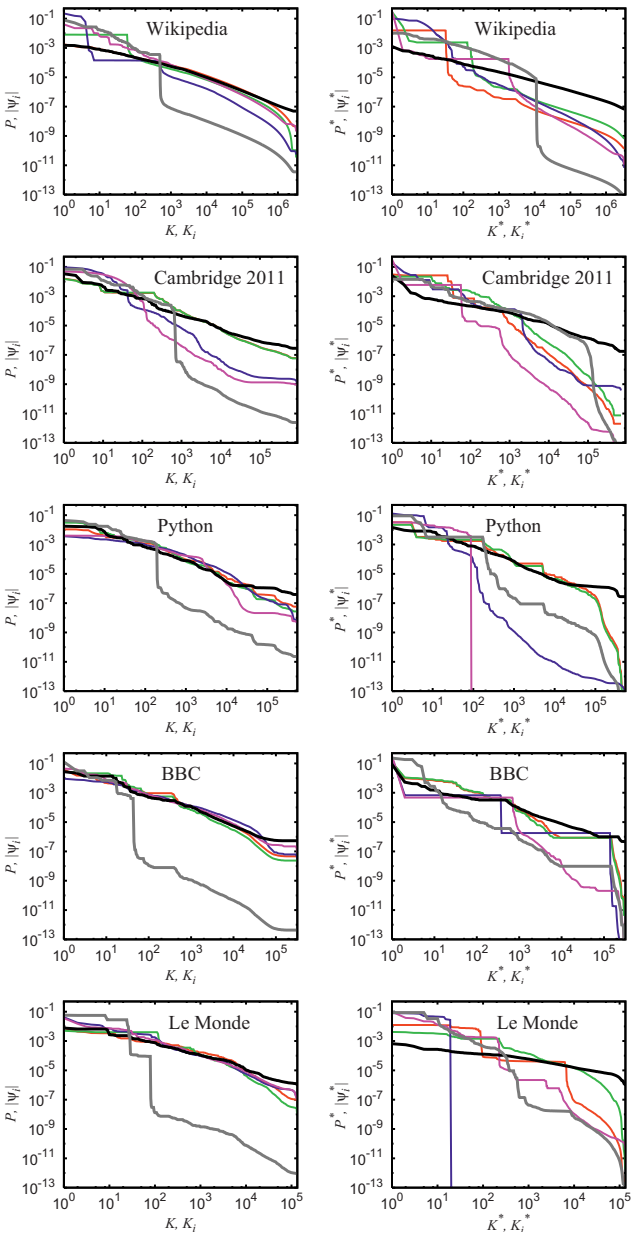


Fig. 2. PageRank P (left column) and CheiRank P^* (right column) vectors are shown as a function of the corresponding rank indexes K or K^* for the Google matrices of Wikipedia, Cambridge 2011, Python, BBC and Le Monde at the damping parameter $\alpha = 0.85$ (thick black curve) and $\alpha = 1 - 10^{-8}$ (thick gray curve). The thin color curves show for each panel the modulus of four core space eigenvectors $|\psi_i|$ of S (left column) and $|\psi_i^*|$ of S^* (right column) versus their ranking indexes K_i or K_i^*. Red and green curves are the eigenvectors corresponding to the two largest core space eigenvalues (in modulus) which are real and close to 1; blue and pink curves are the eigenvectors corresponding to two complex eigenvalues with large imaginary part. The chosen eigenvalues and other relevant quantities for each case are listed in Tables 1–3.

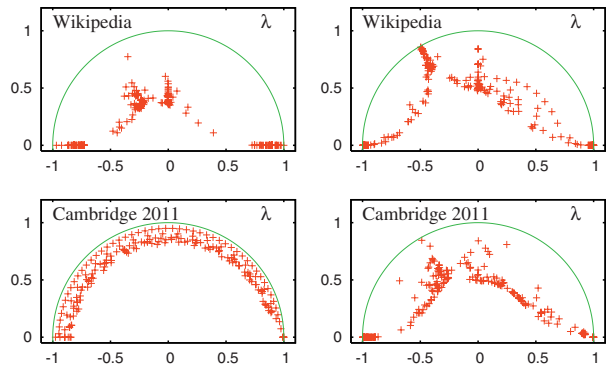


Fig. 3. A selection of 200 complex core space eigenvalues closest to the unit circle for the matrices S (left column) and S^* (right column) of Wikipedia and Cambridge 2011 networks. The characteristics of corresponding eigenvectors are shown in Figures 4 and 5.

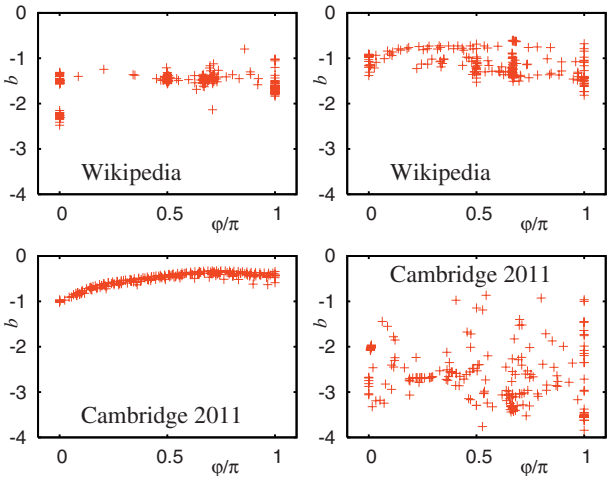


Fig. 4. Left column: algebraic exponent b obtained from a power law fit $|\psi_i(K_i)| \sim K_i^b$ for $K_i \geq 10^4$ shown as a function of the phase $\varphi = \arg(\lambda_i)$ of the complex eigenvalue λ_i associated to the eigenvector ψ_i of S. The shown data points correspond to the eigenvalue selection of Figure 3 for networks of Wikipedia and Cambridge 2011. Right column: the same as in the left column for the eigenvectors of S^*.

of PageRank probability with K. For Wikipedia and Cambridge 2011 we see that eigenvectors $|\psi_i(j)|$ are more localized. The eigenstates of S^* have a significantly more irregular decay compared to the eigenstates of S.

To analyze the properties of core eigenstates of Wikipedia and Cambridge 2011 in a better way, we select 200 core space eigenvalues of S and S^* being closest to the unitary circle $|\lambda| = 1$. These eigenvalues are shown in Figure 3. For these eigenvalues, we compute the corresponding eigenvectors $\psi_i(j)$ and by fitting a power law dependence $|\psi_i(K_i)| \sim K_i^b$ at $K_i \geq 10^4$ we determine the dependence of the exponent b on the phase of the eigenvalue $\varphi = \arg(\lambda_i)$. For Wikipedia, we have values of $|b|$ distributed mainly in the range (1–2) for S and in the range (0.5–1.5) for S^*. For Cambridge 2011, we have a

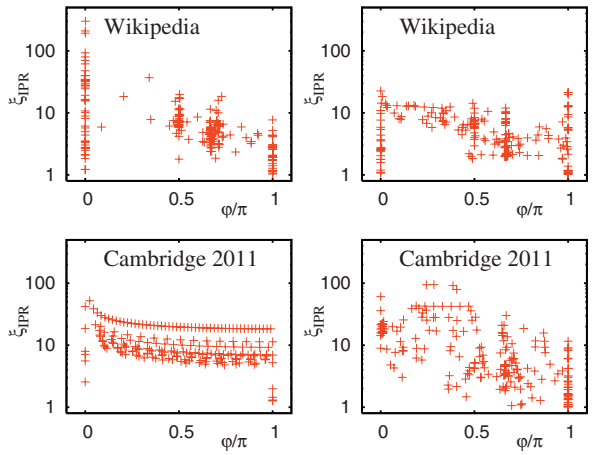


Fig. 5. Left column: inverse participation ratio $\xi_{\rm IPR} = (\sum_j |\psi_i(j)|^2)^2 / \sum_j |\psi_i(j)|^4$ shown as a function of the phase $\varphi = \arg(\lambda_i)$ of the complex eigenvalue λ_i associated to the eigenvector ψ_i of S. The data points correspond to the eigenvalue selection of Figure 3 for networks of Wikipedia and Cambridge 2011. Right column: the same as in the left column for the eigenvectors of S^*.

more compact range (0.5–1) for S while for S^* there is a very broad variation of $|b|$ values in the range (1–4).

The above approximate power law description of the eigenstate decay characterizes their behavior at large K values. The behavior at low K values can be characterized by the inverse participation ratio (IPR) $\xi_{\rm IPR} = (\sum_j |\psi_i(j)|^2)^2 / \sum_j |\psi_i(j)|^4$, which gives an approximate number of nodes on which the main probability of an eigenstate $\psi_i(j)$ is located. We note that such a characteristic is broadly used in disordered mesoscopic systems allowing to detect the Anderson transition from localized phase with finite ξ to delocalized phase with ξ value comparable with the system size [26]. The IPR data are presented in Figure 5 for eigenvalues selection of Figure 3. We find that $\xi_{\rm IPR}$ values are by a factor 10^4 to 10^5 smaller than the network size N. This means that these eigenstates are well localized on a restricted number of nodes. We try to analyze what are these nodes in next section for the example of Wikipedia where the meaning of a node is clearly defined by the title of the corresponding Wikipedia article.

4 Communities of Wikipedia eigenstates

To understand the meaning of other eigenstates in the core space we order selected eigenstates by their decreasing value $|\psi_i(j)|$ and apply a frequency analysis on the first 1000 articles with $K_i \leq 1000$. The mostly frequent word of a given eigenvector is used to label the eigenvector name. These labels with corresponding eigenvalues are shown in Figure 6 in λ-plane. We identify four main categories for the selected eigenvectors shown by different colors in Figure 6: countries (red), biology and medicine (orange), mathematics (blue) and others (green). The category of others contains rather diverse articles

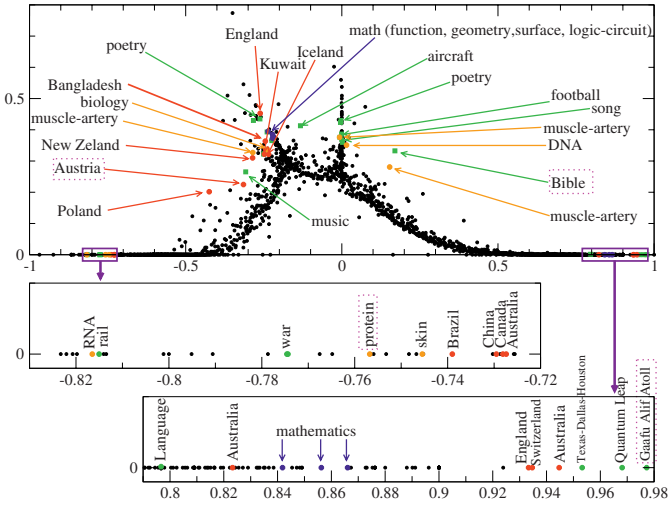


Fig. 6. Complex eigenvalue spectrum of the matrices S for Wikipedia. Highlighted eigenvalues represent different communities of Wikipedia and are labeled by the most repeated and important words following word counting of first 1000 nodes. Color are used in the following way: red for countries, orange for biology, blue for mathematics and green for others. Top panel shows complex plane for positive imaginary part of eigenvalues, while middle and bottom panels focus in the negative and positive real parts. Top 20 nodes with largest values of eigenstates $|\psi_i|$ and their eigenvalues λ_i are given in Tables 4–7 (4 names marked by dotted boxes in figure panels).

about poetry, Bible, football, music, American TV series (e.g., Quantum Leap), small geographical places (e.g., Gaafru Alif Atoll). Clearly these eigenstates select certain specific communities which are relatively weakly coupled with the main bulk part of Wikipedia that generates relatively large modulus of $|\lambda_i|$. The top 20 articles of eigenstate PageRank index K_i are listed in Tables 4–7.

The eigenvector of Table 4 has a positive real λ and is linked to the main article *Gaafu Alif Atoll* which in its turn is linked mainly to atolls in this region. Clearly this case represents well localized community of articles mainly linked between themselves that gives slow relaxation rate of this eigenmode with $\lambda = 0.9772$ being rather close to unity.

In Table 5, we have an eigenvector with real negative eigenvalue $\lambda = -0.8165$ with the top node *Photoactivatable fluorescent protein*. This node is linked to *Kaede (protein)* and *Eos (protein)* with the later being isolated from coral. Its picture is listed in *Portal:Berkshire/Selected picture* which has pictures of *St Paul's Cathedral* and *Legoland Windsor* that generates appearance of these, on a first glance unrelated articles, to be present in this eigenvector. Thus, this eigenvector also highlights a specific community which is somewhat stronger coupled to the global Wikipedia core, due to a link to selected pictures, with a smaller modulus of λ compared to the case of Table 4.

The eigenvector of Table 6 has a complex eigenvalue with $|\lambda| = 0.3733$ and the top article *Portal:Bible*. The top three articles of this eigenvector have very close values of $|\psi_i(j)|$ that seems to be the reason why we have

Table 4. Node rank for decreasing modulus of eigenstate $|\psi_i|$ corresponding to the eigenvalue $\lambda_2 = 0.97724$ (see Fig. 6).

	$\lambda_2 = 0.9772$ ("Gaafu Alif Atol")	$\lvert\psi_i\rvert$
1	Gaafu Alif Atoll	0.00816
2	Kureddhoo (Gaafu Alif Atoll)	0.00812
3	Hithaadhoo (Gaafu Alif Atoll)	0.00808
4	Dhigurah (Gaafu Alif Atoll)	0.00806
5	Maarandhoo (Gaafu Alif Atoll)	0.00806
6	Hulhimendhoo (Gaafu Alif Atoll)	0.00805
7	Araigaiththaa	0.00798
8	Baavandhoo	0.00798
9	Baberaahuttaa	0.00798
10	Bakeiththaa	0.00798
11	Beyruhuttaa	0.00798
12	Beyrumaddoo	0.00798
13	Boaddoo	0.00798
14	Budhiyahuttaa	0.00798
15	Dhevvalaabadhoo	0.00798
16	Dhevvamaagalaa	0.00798
17	Dhigudhoo	0.00798
18	Dhonhuseenahuttaa	0.00798
19	Falhumaafushi	0.00798
20	Falhuverrehaa	0.00798

Table 5. Node rank for decreasing modulus of eigenstate $|\psi_i|$ corresponding to the eigenvalue $\lambda_{80} = -0.8165$ (see Fig. 6).

	$\lambda_{80} = -0.8165$ ("protein")	$\lvert\psi_i\rvert$
1	Photoactivatable fluorescent protein	0.22767
2	Kaede (protein)	0.13942
3	Eos (protein)	0.13942
4	Fusion protein	0.05946
5	Green fluorescent protein	0.05723
6	Portal:Berkshire/Selected picture	0.01019
7	Persistent tunica vasculosa lentis	0.00552
8	Portal:Berkshire/Selected picture/Layout	0.00416
9	Portal:Berkshire/Selected picture/1	0.00416
10	Portal:Berkshire/Nominate/ Selected picture	0.00416
11	Persistent hyperplastic primary vitreous	0.00338
12	Tunica vasculosa lentis	0.00338
13	Tpr-met fusion protein	0.00319
14	St Paul's Cathedral	0.00256
15	Legoland Windsor	0.00255
16	Complementary DNA	0.00252
17	Gené	0.00221
18	Gene	0.00215
19	Gag-onc fusion protein	0.00181
20	Protein	0.00177

$\varphi = \arg(\lambda_i) = \pi \times 0.3496$ being very close to $\pi/3$. The Bible is strongly linked to various aspects of human society that leads to a relatively small modulus value of this well defined community.

In Table 7, we have an eigenvector which starts from the article *Lower Austria* with the eigenvalue modulus $|\lambda| = 0.3869$. This article is linked to such articles as *Austria* and *Upper Austria* with historical links to *Styria*. It also links to its city capital *Krems an der Donau*. The articles *World War II* and *Jew* appear due to a sentence

Table 6. Node rank for decreasing modulus of eigenstate $|\psi_i|$ corresponding to the eigenvalue $\lambda_{1481} = 0.1699 + i0.3325$ (see Fig. 6).

	$\lambda_{1481} = 0.1699 + i0.3325$ ("Bible")	$\lvert\psi_i\rvert$
1	Portal:Bible	0.02311
2	Portal:Bible/Featured chapter/archives	0.02201
3	Portal:Bible/Featured article	0.02063
4	Bible	0.01684
5	Portal:Bible/Featured chapter	0.01644
6	Books of Samuel	0.00852
7	Books of Kings	0.00849
8	Books of Chronicles	0.00840
9	Book of Leviticus	0.00426
10	Book of Ezra	0.00425
11	Book of Ruth	0.00420
12	Book of Deuteronomy	0.00417
13	Book of Joshua	0.00400
14	Book of Exodus	0.00397
15	Book of Judges	0.00395
16	Book of Genesis	0.00394
17	Book of Numbers	0.00389
18	Portal:Bible/Featured chapter/1 Kings	0.00347
19	Portal:Bible/Featured chapter/Numbers	0.00347
20	Portal:Bible/Featured chapter/2 Samuel	0.00347

Table 7. Node rank for decreasing modulus of eigenstate $|\psi_i|$ corresponding to the eigenvalue $\lambda_{1395} = -0.3149 + i0.2248$ (see Fig. 6).

	$\lambda_{1395} = -0.3149 + i0.2248$ ("Austria")	$\lvert\psi_i\rvert$
1	Lower Austria	0.04284
2	Austria	0.03112
3	Upper Austria	0.00817
4	Styria	0.00781
5	Burgenland	0.00307
6	World War II	0.00304
7	Krems an der Donau	0.00282
8	Jew	0.00272
9	Slovakia	0.00268
10	Bruck an der Leitha (district)	0.00265
11	History of Austria	0.00263
12	Wiener Neustadt	0.00260
13	Mostviertel	0.00251
14	States of Austria	0.00250
15	Waidhofen an der Ybbs	0.00249
16	MELK	0.00246
17	Melk	0.00246
18	Bundesland (Austria)	0.00239
19	Wachau	0.00233
20	Waldviertel	0.00226

"Before World War II, Lower Austria had the largest number of Jews in Austria". Due to links with very popular nodes the eigenvector of this community has a relative small modulus of λ.

Let us make here a few additional remarks about other eigenvectors. For example, we analyzed the meaning of eigenvector with $\lambda = -0.3500 + i0.7737 = |\lambda| \exp(i\theta)$ (located slightly above the word *England* in Fig. 6). Its top five amplitude modulus are *Screen Producers Association of Australia*, *Screen Producers Association of Australia (SPAA)*, *SPAA Conference*, *SPAA Fringe*, *Sydney*. This clearly shows that this vector selects a certain community of Australian Screen Producers. It is interesting to note that we have here $\theta = 114^\circ$ being close to the angle $2\pi/3$ corresponding to $1/3$ resonance rotations mainly between first three top nodes.

In fact, there are other eigenvalues which have θ being close to resonance values with $\theta/2\pi = 1/3, 1/4 \ldots$ Thus, the eigenvector *England* has $\lambda = -0.2613 + i0.4527$ with $\theta = 120^\circ$ corresponding to the resonance rotation between three nodes. Indeed, the top amplitudes of this eigenvector have titles *Charles William Hempel*, *Charles Frederick Hempel*, *Carl Frederick Hempel* with strong links between these titles leading to $1/3$ rotation (this vector is marked as *England* since this word is the most frequent among top 1000 titles).

There are other eigenvalues close to $1/3$ resonance rotation. Thus, we have $\lambda = -0.2621 + i0.4346$ with $\theta = 121^\circ$ marked as *poetry* in Figure 6. This eigenvector has top amplitude modulus: *Poetry* (0.0622), *Portal:Poetry/poem archive* (0.03339), *Portal:Poetry/poem archive/2006 archive* (0.03289), *Portal:Poetry* (0.03180), *Walter Raleigh* (0.0064). We think that the top nodes 2, 3, 4 have practically the same amplitudes thus corresponding to the resonance $1/3$ rotation between these three nodes.

There is also another eigenvector marked *poetry* in Figure 6 with $\lambda = -0.0026 + i0.4297$ and $\theta \approx 90^\circ$. In fact this article speaks about *1000s in poetry* with approximately equal 6 amplitudes about poetry in various years that corresponds to a resonance $1/6$ rotation generating $\theta \approx 90^\circ$. There are also other vectors with resonance values $1/2, 1/4, 1/6$ that produce eigenvalues with a dominant imaginary part. We also note that there are other resonance eigenvalues among those given in Table 3 (e.g., λ_{38} with $\theta = 120.1^\circ$). We think that such resonance θ values have close similarity with those of random matrix models of small size $N = 3, 4, 5, 6$ analyzed in reference [25] corresponding to the main part of information exchange between a small number of nodes.

The above analysis shows that the eigenvectors of the Google matrix of Wikipedia clearly identify certain communities which are relatively weakly connected with the Wikipedia core when the modulus of corresponding eigenvalue is close to unity. For moderate values of $|\lambda|$, we still have well defined communities which however have stronger links with some popular articles (e.g., countries) that lead to a more rapid decay of such eigenmodes.

The above results show that the analysis of eigenvectors highlights interesting features of communities and network structure. However, a priori it is not evident what is a correspondence between the numerically obtained eigenvectors and the specific community features in which someone has a specific interest. It is possible that for a well defined community it can be useful to construct a personalized Google matrix (see, e.g., [5]) and to perform analysis of its eigenstates.

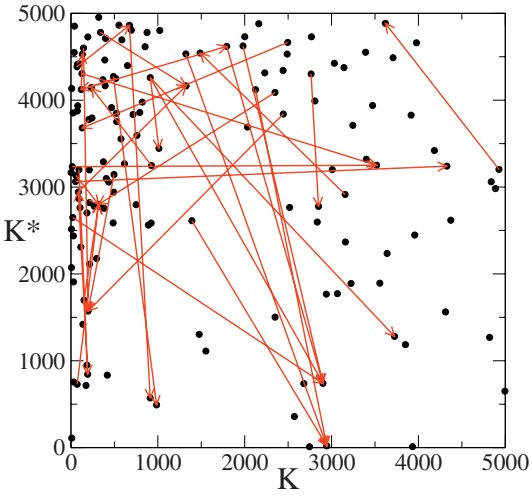


Fig. 7. Top 5000 values in PageRank-CheiRank plane (K, K^*) of Wikipedia. All nodes and all links in this region are shown by black circles and red arrows, respectively.

5 CheiRank versus PageRank plane

As it is discussed in references [15,16,18,19], it is useful to look on the distribution of network nodes on PageRank-CheiRank plane (K, K^*). For Wikipedia a large scale distribution is analyzed in references [16,19] and the networks of British Universities, Linux Kernel and Twitter are considered in references [15,19].

In Figure 7, we show for Wikipedia the distribution of nodes in (K, K^*) plane for a relatively small range of top 5000 values of K, K^*. All directed links in this region are also shown. In fact the number of such links and number of nodes in this region are relatively small. Indeed, a large scale density of nodes (see Fig. 3 in Ref. [16]) shows that the density of nodes is not very high at the top corner of PageRank-CheiRank plane. This happens due to the fact that top nodes of PageRank, whose components are proportional to the number of ingoing links, are usually not those of CheiRank, whose components are proportional to the number to outgoing links.

The correlation between PageRank and CheiRank vectors can be characterized by their correlator [18,19]:

$$\kappa = N \sum_{i=1}^{N} P(K(i)) P^*(K^*(i)) - 1. \qquad (3)$$

For our networks we find its values to be $\kappa = 4.08$ (Wikipedia), 41.5 (Cambridge 2011), 12.9 (Python), 140.2 (BBC), 0.85 (Le Monde). Except for the case of Le Monde, these values are relatively high showing that there is a significant correlation between PageRank and CheiRank probabilities on corresponding networks. We remind that for Linux Kernel networks the values of κ are close to zero corresponding to absence of correlations there [18,19].

The strong difference between κ values for BBC and Le Monde shows that the structure of these two web sites is very different. To analyze this difference in a better way we show the density of nodes for these two networks on small and large scales in Figure 8. For small scale, shown by top panels, it is clear that the density of nodes is significantly larger for BBC network. However, this difference becomes even more drastic on the large logarithmic scale of the whole network shown in bottom panels. Indeed, on a logarithmic scale we see that BBC network has a square like distribution region with a certain probability maximum around the diagonal $K \approx K^*$ while Le Monde network has a triangular type distribution which is typical for networks without correlations between PageRank and CheiRank vectors, like it is the case for the Linux Kernel networks (see Fig. 4 in Ref. [19]). Indeed, a random procedure of node generation on (K, K^*) plane gives such a triangular distribution without correlations between PageRank and CheiRank nodes (see procedure description and right panel of Fig. 4 in Ref. [16]). This analysis shows that BBC and Le Monde agencies handle information flows on their web sites in a drastically different manner. Thus for the BBC web site the most popular articles are at the same time also the most communicative ones while in contrast to that for the Le Monde web site the most popular and most communicative articles are very different.

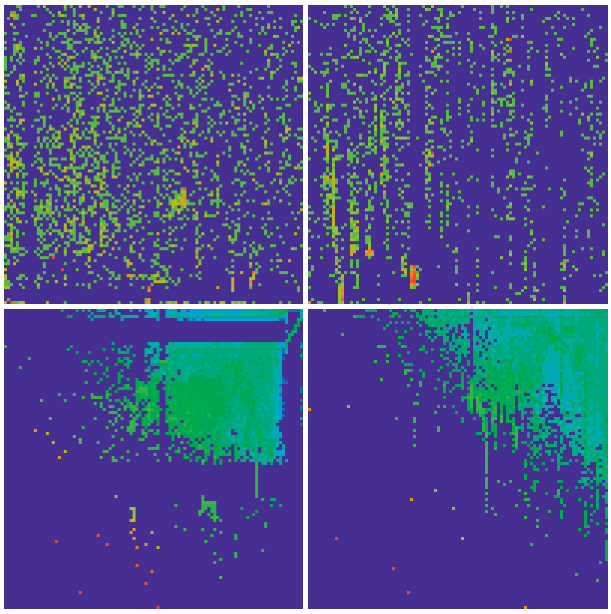


Fig. 8. Density of nodes $W(K, K^*)$ on PageRank-CheiRank plane (K, K^*) for the networks of BBC (left panels) and Le Monde (right panels). Top panels show density in the range $1 \leq K, K^* \leq 10^4$ with averaging over cells of size 100×100; bottom panels show density averaged over 100×100 logarithmically equidistant grids for $0 \leq \ln K, \ln K^* \leq \ln N$, the density is averaged over all nodes inside each cell of the grid, the normalization condition is $\sum_{K,K^*} W(K, K^*) = 1$. Color varies from blue at zero value to red at maximal density value. At each panel the x-axis corresponds to K (or $\ln K$ for the bottom panels) and the y-axis to K^* (or $\ln K^*$ for the bottom panels).

6 Links distribution over PageRank nodes

To understand the properties of directional flow on a network it is also useful to analyze the distribution of links

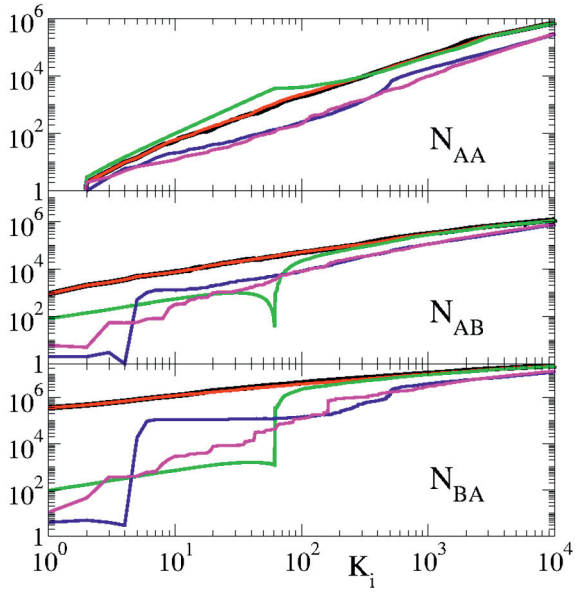


Fig. 9. Number of links between or inside sets A and B defined by the index K_i ordered by decreasing absolute value of Wikipedia eigenstates. The number of links starting and pointing to nodes inside the set A (N_{AA}) is shown in top panel as a function of K_i. The cases of links from set A to set B (N_{AB}) and from B to A (N_{BA}) are shown in middle and bottom panel, respectively. Note that the total number of links is conserved and the quantity N_{BB} can be obtained as $N_{BB} = N_\ell - N_{AA} - N_{AB} - N_{BA}$. The case of PageRank vector with damping parameter $\alpha = 0.85$ is shown by a black curve versus K index. The color curves show the cases of four core space eigenvectors $|\psi_i|$ of S versus their ranking indexes K_i. Red and green curves are the eigenvectors corresponding to the two largest core space eigenvalues (in modulus) being $\lambda_1 = 0.99998702$ and $\lambda_2 = 0.97723699$, respectively; blue and pink curves are the eigenvectors corresponding to two complex eigenvalues with large imaginary part being $\lambda_{52} = -0.35003316 + i0.77373677$ and $\lambda_{864} = -0.34293502 + i0.43144930$, respectively.

over PageRank nodes. We illustrate this approach for the Wikipedia network. Suppose that all nodes are ordered in a decreasing order of modulus of a given eigenvector. For the PageRank vector all nodes are numbered by the PageRank index K, while for a given eigenstate $\psi_i(j)$ all nodes are numbered by a local corresponding index K_i. We now divide all nodes on two parts A and B with $1, \ldots, K_i$ nodes for A and $K_i + 1, \ldots, N$ nodes for B. Then we determine the number of links N_{AA} starting and ending in part A, the number of links N_{AB} pointing from part A to part B and the number of links N_{BA} pointing from part B to part A. The number of links inside part B is then $N_{BB} = N_\ell - N_{AA} - N_{AB} - N_{BA}$. For the PageRank vector, the dependence of N_{AA} on K was analyzed for different networks in reference [15]. Here we generalize this concept to consider links between two parts A, B for various eigenvectors of the Google matrix.

According to the data of Figure 9, we find that for all eigenvectors $N_{AA} \propto K_i^{1.5}$ grows approximately in an algebraic way with the exponent being close to 1.5 being similar to the PageRank case considered in reference [15]. However, the dependence of N_{AB} and N_{BA} on K_i is rather different for different eigenstates. For the PageRank and the λ_1 eigenvector, we find practically the same behavior linked to the fact that at $\alpha = 0.85$, the PageRank vector is rather close to the first core space eigenvector (see discussion in Ref. [13]). Here, the interesting point is that at small values of K_i we have N_{BA} being larger than N_{AB} almost by a factor 100. This is due to the fact that low rank nodes at large K_i point preferentially to high rank nodes at low K_i. For other three eigenvectors with λ_2, λ_{52}, λ_{864}, we find well pronounced step-like behavior of N_{AB}, N_{BA} on K_i. We argue that the step size in K_i is given by the size of a community which has preferential links mainly inside the community. Indeed, for the eigenvector of λ_2 (see Tab. 3) we see that the community size is approximately $N_{\rm cs} \approx 1/|\psi_1| \approx 100$ that corresponds to the step size in $K_i \approx 70$ for this case.

These results show that the analysis of the link distribution over the PageRank index provides interesting and useful information about characteristics and properties of directed networks.

7 Discussion

In this work, we performed a spectral analysis of eigenvalues and eigenstates of the Google matrix of Wikipedia and other networks. Our study shows that the spectrum of the core space component has eigenvalues in a close vicinity of $\lambda = 1$ and that there are isolated subspaces which give a degeneracy of the eigenvalue $\lambda = 1$. The eigenvalues and eigenstates with relatively large values of $|\lambda|$ can be efficiently determined by the powerful Arnoldi method. These eigenstates are mainly located on well defined network communities. We also find that the spectrum changes drastically from one network to another even if the distribution of links and decay of PageRank is rather similar for the networks considered. This means that the properties of directed networks strongly depend on the internal network structure. We show that the correlation between PageRank and CheiRank vectors highlights specific properties of information flow on directed network. For example, this correlation demonstrates a drastic difference between web sites of BBC and Le Monde. The distribution of links between PageRank nodes also provides an interesting information about the network structure. On the basis of our studies, we argue that the developed spectral analysis of Google matrix brings a deeper understanding of information flow on real directed networks.

We thank A.D. Chepelianskii for making to us available network data collected by him for networks of Cambridge University, Python, BBC, Le Monde in March 2011. Our research presented here is supported in part by the EC FET Open project "New tools and algorithms for directed network analysis" (NADINE No. 288956). This work was granted access to the HPC resources of CALMIP (Toulouse) under the allocation 2012-P0110.

Appendix

The tables are given in the text of the paper. The notations used in the tables are: N is network size, N_ℓ is the number of links, n_A is the Arnoldi dimension used for the Arnoldi method for the core space eigenvalues, N_d is the number of invariant subspaces, $d_{\max}$ gives a maximal subspace dimension, $N_{\rm circ.}$ notes number of eigenvalues on the unit circle with $|\lambda_i| = 1$, N_1 notes number of unit eigenvalues with $\lambda_i = 1$. We remark that $N_s \geq N_{\rm circ.} \geq N_1 \geq N_d$ and $N_s \geq d_{\max}$ and the average subspace dimension is given by: $\langle d \rangle = N_s/N_d$. We note that the values of N, N_ℓ for network of Cambridge 2011 are slightly different from those given in [19] due to a slightly different procedure of cleaning of row data collection (e.g., count of pdf and other type nodes). Eigenvalues for eigenvectors are shown in Figure 1 with the colors red, green, blue or pink corresponding to colors of Table 3. The index m of λ_m in Tables 3–7 counts the order number of core eigenvalues in a decreasing order of $|\lambda_m|$.

References

1. Wikipedia, World Wide Web, `http://en.wikipedia.org/wiki/World_Wide_Web`
2. A.A. Markov, Izvestiya Fiziko-matematicheskogo obschestva pri Kazanskom universitete **15**, 135 (1906) (in Russian)
3. M. Brin, G. Stuck, *Introduction to dynamical systems* (Cambridge University Press, Cambridge, 2002)
4. S. Brin, L. Page, Computer Networks and ISDN Systems **30**, 107 (1998)
5. A.M. Langville, C.D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings* (Princeton University Press, Princeton, 2006)
6. D. Donato, L. Laura, S. Leonardi, S. Millozzi, Eur. Phys. J. B **38**, 239 (2004)
7. G. Pandurangan, P. Raghavan, E. Upfal, Internet Math. **3**, 1 (2005)
8. N. Litvak, W.R.W. Scheinhardt, Y. Volkovich, Lect. Notes Comput. Sci. **4936**, 72 (2008)
9. S. Serra-Capizzano, SIAM J. Matrix Anal. Appl. **27**, 305 (2005)
10. O. Giraud, B. Georgeot, D.L. Shepelyansky, Phys. Rev. E **80**, 026107 (2009)
11. B. Georgeot, O. Giraud, D.L. Shepelyansky, Phys. Rev. E **81**, 056109 (2010)
12. L. Ermann, A.D. Chepelianskii, D.L. Shepelyansky, Eur. Phys. J. B **79**, 115 (2011)
13. K.M. Frahm, B. Georgeot, D.L. Shepelyansky, J. Phys, A **44**, 465101 (2011)
14. L. Ermann, D.L. Shepelyansky, Acta Phys. Polonica A **120**, A158 (2011), `www.quantware.ups-tlse.fr/QWLIB/tradecheirank/`
15. K.M. Frahm, D.L. Shepelyansky, Eur. Phys. J. B **85**, 355 (2012), `www.quantware.ups-tlse.fr/QWLIB/twittermatrix/`
16. A.O. Zhirov, O.V. Zhirov, D.L. Shepelyansky, Eur. Phys. J. B **77**, 523 (2010), `www.quantware.ups-tlse.fr/QWLIB/2drankwikipedia/`
17. Wikipedia, Google matrix, `http://en.wikipedia.org/wiki/Google_matrix`
18. A.D. Chepelianskii, *Towards physical laws for software architecture*, `arXiv:1003.5455[cs.SE]` (2010), `www.quantware.ups-tlse.fr/QWLIB/linuxnetwork/`
19. L. Ermann, A.D. Chepelianskii, D.L. Shepelyansky, J. Phys. A **45**, 275101 (2012), `www.quantware.ups-tlse.fr/QWLIB/dvvadi/`
20. Wikipedia, CheiRank, `http://en.wikipedia.org/wiki/CheiRank`
21. G.W. Stewart, *Matrix Algorithms Volume II: Eigensystems* (SIAM, Philadelphia, 2001)
22. G.H. Golub, C. Greif, BIT Num. Math. **46**, 759 (2006)
23. K.M. Frahm, D.L. Shepelyansky, Eur. Phys. J. B **76**, 57 (2010)
24. Academic Web Link Database Project `http://cybermetrics.wlv.ac.uk/database/`
25. K. Zyczkowski, M. Kus, W. Slomczynski, H.-J. Sommers, J. Phys. A **36**, 3425 (2003)
26. F. Evers, A.D. Mirlin, Rev. Mod. Phys. **80**, 1355 (2008)

PageRank model of opinion formation on Ulam networks

L. Chakhmakhchyan[a,b,c], D. Shepelyansky[d]

[a] *A.I. Alikhanyan National Science Laboratory, 0036 Yerevan, Armenia*
[b] *Laboratoire Interdisciplinaire Carnot de Bourgogne, UMR CNRS 6303, Université de Bourgogne, 21078 Dijon Cedex, France*
[c] *Institute for Physical Research, 0203 Ashtarak-2, Armenia*
[d] *Laboratoire de Physique Théorique du CNRS, IRSAMC, Université de Toulouse, UPS, F-31062 Toulouse, France*

Abstract

We consider a PageRank model of opinion formation on Ulam networks, generated by the intermittency map and the typical Chirikov map. The Ulam networks generated by these maps have certain similarities with such scale-free networks as the World Wide Web (WWW), showing an algebraic decay of the PageRank probability. We find that the opinion formation process on Ulam networks have certain similarities but also distinct features comparing to the WWW. We attribute these distinctions to internal differences in network structure of the Ulam and WWW networks. We also analyze the process of opinion formation in the frame of generalized Sznajd model which protects opinion of small communities.

Keywords:
PageRank, Ulam networks, opinion formation

1. Introduction

The understanding of mechanisms of opinion formation in the modern society is at the heart of a newly emerged research field, known as sociophysics [1]. A number of voter models has been developed during the last few decades for understanding of nontrivial features of opinion formation in a society (see Refs. [2–6] for details). However, these models are generally considered on abstract regular lattices, which are very different from a scale-free structure of modern social networks with hundreds of millions of users. In particular, such social networks as LiveJournal [7], Facebook [8] or Twitter [9] allow to have a rapid information exchange over a large fraction of network users and to share social events, making an essential contribution to the mass opinion formation. These social networks have a growing influence on the social and political life.

A straightforward way of taking into account the main features of such networks was recently proposed in Ref. [10]: the opinion on each given node of a scale-free network is assumed to be formed by opinions of its linked neighbors, weighted with their PageRank probability. The latter quantity is interpreted as a probability of finding a random surfer on a given node [11, 12]. Obviously, this approach introduces the notion of importance of a node, naturally reproducing the real society, where each person has its degree of authority. Mathematically the PageRank is defined as the right eigenvector with unit eigenvalue of Google matrix of a given network [12]. Although the PageRank algorithm was initially proposed for an efficient ranking of web pages [11], it turned out to be useful for the analysis of broad

Email addresses: levonc@rambler.ru (L. Chakhmakhchyan), dima@irsamc.ups-tlse.fr (D. Shepelyansky)
URL: http://www.quantware.ups-tlse.fr/dima (D. Shepelyansky)

class of real networks including e.g. scientific journal rating, neuronal and world trade networks, etc. [13–16]. The rules of Google matrix construction for a given directed network are described in [11, 12, 15].

In the present work we study the PageRank Opinion Formation (PROF) model, proposed in [10], on another family of directed networks, known as Ulam networks. The Ulam method, introduced in Ref. [17], was initially proposed for constructing a matrix approximant for a Perron-Frobenius operator of dynamical systems (we note that the Google matrix also falls in the same class of operators). The Ulam conjecture [17] was shown to be true for various types of generic fully chaotic maps on an interval [18–21]. Recent studies have shown that this method naturally generates a class of directed networks, which properties have certain similarities with the WWW directed networks [22, 23]. Thus the Ulam networks demonstrate a sensitivity to the damping parameter α of the corresponding Google matrix and a power law decay of its PageRank. Here we are interested in two particular examples: the typical Chirikov map with dissipation and the one dimensional intermittency map. The first one, introduced in Ref. [24] for a description of continuous chaotic flows, has been studied in [22, 25]. The second one is generated from intermittency maps, studied in systems exhibiting intermittency phenomenon, featuring anomalous diffusion and transport [26–30].

In this work we analyze the properties of PROF model on the Ulam networks and study the influence of network elite on opinion formation process. We also consider the Sznajd model [31], generalized for scale-free networks following [10]. This model incorporates the effect of groups, consisting of voters of the same opinion following the trade union slogan *united we stand, divided we fall*.

In the rest, the paper is organized as follows: in the next section we give a brief description of the Ulam method and PROF model and present our numerical results. In Section 3 we combine the PROF and Sznajd models and analyze their properties on Ulam networks. The discussion of the results is given in Section 4.

2. The PROF model and Ulam networks

We start with a brief outline of the Ulam method for dynamical maps following the description given in [22, 23]. As the first model we use the one-dimensional (1d) intermittency map described in [23]:

$$\bar{x} = f(x) = \begin{cases} x + (2x)^{z_1}/2, \text{ for } 0 \le x < 1/2 \\ (2x - 1 - (1-x)^{z_2} + 1/2^{z_2})/ \\ (1 + 1/2^{z_2}), \text{ for } 1/2 \le x < 1 \end{cases} \quad (1)$$

where $\bar{x}$ notes the new value of variable x. The Ulam network generated by this map is constructed in the following way: the whole interval $0 < x < 1$ is divided to N equal cells and N_c trajectories (randomly distributed inside a cell) are iterated on one map iteration from cell j, to obtain matrix elements for transitions to cell i: $S_{ij} = N_i(j)/N_c$, where $N_i(j)$ is the number of trajectories arrived from cell j to cell i. From the matrix S_{ij}, one constructs the Google matrix $\mathbf{G}$, defined as:

$$\mathbf{G} = \alpha \mathbf{S} + (1 - \alpha)\mathbf{E}/N, \quad (2)$$

where $E_{ij} = 1$ and α is the damping factor. We use a probability normalization of the eigenstate $|\psi_1\rangle$ (with a unit eigenvalue) of the matrix (2), which results in the PageRank P_j of the network (see [23] for a detailed description of its properties). We also arrange all N nodes in monotonic decreasing order of the PageRank probability. In what follows we set the damping factor of the Google matrix of the intermittency map (1) to $\alpha = 1$. We also fix the parameters of (1) to $z_1 = 2$ and $z_2 = 0.2$. This choice gives a power law decay of the PageRank (sorted in descending order): $P_j \propto 1/j$ [23].

We construct the PROF model for the Google matrix of the intermittency map (1) in the following way. We associate each node of the network with a spin variable σ_i, taking values $+1$ (red color) or -1 (blue color). Afterwards, we compute the quantity Σ_i over all directly linked

neighbors j of a node i:

$$\Sigma_i = a\sum_j P^+_{j,in} + b\sum_j P^+_{j,out} - a\sum_j P^-_{j,in} - b\sum_j P^-_{j,out}, \quad (3)$$

where $P_{j,in}$ and $P_{j,out}$ denote the PageRank probability P_j of a node j pointing to node i (incoming link) and a node j to which node i points to (outgoing link). The two parameters a and b are used to tune the importance of incoming and outgoing links with the imposed relation $a + b = 1$ $(0 < a, b < 1)$. The values P^+ and P^- correspond to red and blue nodes respectively. On one iteration the value of a spin σ_i is fixed to $+1$ (red) for $\Sigma_i > 0$ or -1 (blue) for $\Sigma_i < 0$. We note that the a and b parameters define the type of a society: for a large value a a person takes mainly the opinion of those electors who point to him/her (a tenacious society) and the opposite for large values of b (a conformist society).

In Fig. 1 we present the evolution of the fraction of red nodes $f(t)$ ($f(t) = N_{red}/N$) versus the iteration time t. We distinguish two important cases, namely, when initially opinions are randomly distributed over the network, and when the first N_{top} nodes of the highest PageRank probability are of the same opinion, e.g. of a red color. For a random distribution the system converges to its final state after $t_c \approx 25$ iterations for $a = b = 0.5$. Iterations are defined as in [10].

In Fig. 1 we show the time evolution of opinion for the initial state where the society elite, corresponding to the top nodes N_{top} of highest PageRank probability, has the same opinion (dotted curves). In this case the elite can impose its opinion to a faction of society which is by a factor $2-3$ larger than the initial fraction. However, in comparison with the social or university networks considered in [10] this increase is less significant that is due to a smaller number of linked nodes for the Ulam network of intermittency map.

For a comprehensive analyzes of the dependence of the final fraction of red nodes f_f on the initial state f_i, we consider below the evolution of $f(t)$ for a large number of N_r initial (random) distributions of red nodes (Fig. 2). We find that

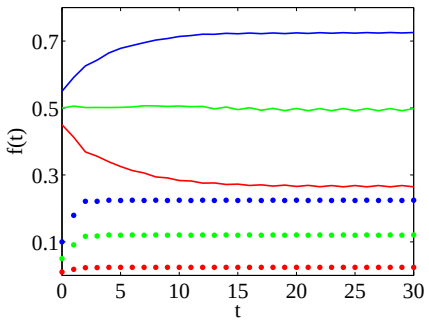


Figure 1: Time evolution of the opinion, given by a fraction of red nodes $f(t)$, as a function of number of time iteration t ($a = b = 0.5$). Full curves correspond to different initial fractions $f_i = f(0)$ at a random realization: $f_i = 0.45$ (red); 0.5 (green); 0.55 (blue). The dotted curves stand for the initial state with the first N_{top} nodes of the highest PageRank probability being red: $N_{top} = 100$ (red); $N_{top} = 500$ (green); $N_{top} = 1000$ (blue). The total matrix size is $N = 10^4$; $\alpha = 1$.

there is a certain critical value f_c such, that initial fractions f_i of red nodes completely die out if $f_i < f_c$, or become dominant for $f_i > 1 - f_c$. For $a = 0.2$ the value of f_c is $f_c \approx 0.45$, while for $a = 0.65$ we have $f_c \approx 0.35$. In contrast to results obtained in [10] we find that the system has no bistability for $a < 0.7$: the final state is fixed for a concrete homogeneous initial distribution of opinions. However, for a dominating tenacious society at $a > 0.7$ there is a small probability that a small initial fraction of red nodes leads to a complete domination of red color for values of $f_i > f_c$ (see Fig. 2 left bottom panel). For the case of $a = 0.8$, we have $f_c \approx 0.3$. Obviously, the results are symmetric with respect to a change of red and blue colors.

We also analyze how the final state depends on the number of the elite members N_{top} with the highest PageRank of the same opinion (Fig. 3). We see that for any type of a society (any a) there exists a value of N^c_{top} such that the elite can convince the whole society, if $N_{top} > N^c_{top}$. Note that the value of N^c_{top} depends on the tenacious parameter a. The larger the tenacious parameter is, the smaller number of the elite members of a same opinion can bring the system to unanimity.

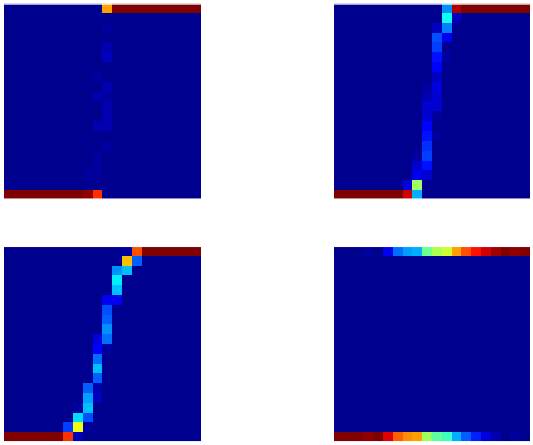


Figure 2: Density plot of probability W_f to find a final red fraction f_f, shown in y-axis, in dependence on an initial red fraction f_i, shown in x-axis; data are shown inside the unit square $0 < f_i, f_f < 1$. The values of W_f are defined as a relative number of realizations found inside each of 20×20 cells, which cover the whole unit square. Here $N_r = 10^3$ realizations of randomly distributed red and blue colors are used to obtain W_f values (with convergence time up to $t = 150$). Here $a = 0.2$ (left top panel), 0.5 (left bottom panel), 0.65 (right top panel), 0.8 (right bottom panel); $N = 10^4$. The probability W_f is proportional to color changing from zero (blue) to unity (brown).

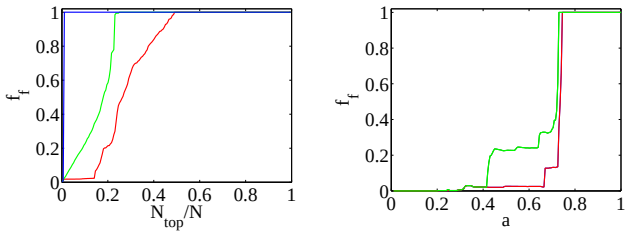


Figure 3: Left panel: final fraction of red nodes f_f versus N_{top}/N, for $a = 0.4$ (red), 0.6 (green), 0.8 (blue). Right panel: dependence of the final fraction of red nodes f_f on the parameter a, for initial state with different number of the first N_{top} nodes of the highest PageRank being red: $N_{top} = 100$ (red); 1000 (green). Here $N = 10^4$.

3. The generalized PROF-Sznajd model

In this section we consider the properties of the combination of PROF and Sznajd models [31]. The Sznajd model features the idea of groups of a society and thus incorporates a well-known principle "United we stand, divided we fall". A thorough analyzes of the problem on regular lattice networks can be found in Ref. [32]. The present generalization (which results in the PROF-Sznajd model) is applicable to scale-free and Ulam networks. We define the notion of group of nodes at each discrete time step τ following Ref. [10]:

1. we pick randomly a node i in the network and consider the state of the $N_g - 1$ highest PageRank nodes pointing to it;
2. if the node i and all other $N_g - 1$ nodes have the same color (same spin orientation), these N_g nodes form a group, whose effective PageRank value is the sum of all the member values $P_g = \sum_j^{N_g} P_j$. If it is not the case, we leave the nodes unchanged and perform the next time step;
3. consider all the nodes pointing to any member of the group and check all these nodes n directly linked to the group: if an individual node PageRank value P_n is less than the defined above P_g, the node joins the group by taking the same color (polarization) as the group nodes and increase P_g by the value of P_n; if it is not the case, a node is left unchanged.

In Fig. 4 we present a typical behavior of the PROF-Sznajd model on Ulam network generated by the intermittency map. Firstly, we find that the convergence time is longer than that of the PROF model, which is the generic feature of the Sznajd model. The system converges to its final state after a time τ_c of the order of $\tau_c \sim 10N$. Note that there are still some fluctuations in the steady state regime, which were absent in the conventional PROF model. Another observation concerns the group size N_g: we find that the size of the group does not affect much the properties of the model: there is a small decrease in the resistivity of minorities with the group size increase (of around 2% with a change from $N_g = 3$ to $N_g = 4$). Furthermore, the network practically does not have nodes with more than four incoming links, hence, we find that considering a group size with $N_g > 5$ loses its sense.

The right panel of Fig. 4 shows a density plot of probability W_f, constructed in a similar to Fig. 2

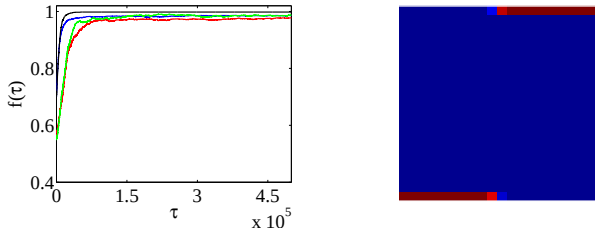


Figure 4: Left panel: time evolution of the fraction of red nodes $f(\tau)$ of the PROF-Sznajd model, with different initial fractions of red nodes and the group size N_g (at one random realization each): $f_i = 0.55$, $N_g = 3$ (red); $f_i = 0.55$, $N_g = 4$ (green); $f_i = 0.7$, $N_g = 3$ (blue); $f_i = 0.7$, $N_g = 4$ (black). Right panel: the same as in Fig. 2, but for the PROF-Sznajd model with group size $N_g = 3$, with convergence time up to $\tau = 5 \cdot 10^5$; colors are as in Fig. 2. Here $N = 10^4$.

way. We see, that the rate of surviving of small fractions of (red) nodes is drastically small (we address this result to the poor incoming link structure of the Ulam network). The initial states are suppressed if $f_i \lesssim 0.45$. But for $0.45 < f_i < 0.5$ ($0.5 < f_i < 0.55$) there is a small probability of approximately 8% that the fraction will become dominant (be suppressed). Outside of this small range of f_i we don't find any regions of bistability: the final state of the system is fixed.

For the PROF-Sznajd model we are additionally interested in the Ulam network, generated by another dynamical map, the typical Chirikov map with dissipation:

$$\begin{cases} y_{t+1} = \eta y_t + k \sin(x_t + \theta_t), \\ x_{t+1} = x_t + y_{t+1}. \end{cases} \qquad (4)$$

Here the dynamical variables x and y are taken at integer moments of time t. Also x has a meaning of phase variable and y is a conjugated momentum or action. For a detailed description of this dynamical system, see Ref. [22]. The map region is $0 \le x < 2\pi$ and $-\pi \le y < \pi$, with 2π-periodic boundary conditions. The phases $\theta_t = \theta_{t+T}$ are T random phases periodically repeated along time t. Here we consider the $T10$ case with $T = 10$, analyzed in Ref. [22]. The values of parameters are set to $\eta = 0.99$, $k = 0.22$. The list of 10 values of θ_t phases can be found in the Appendix of Ref. [22]. For the construction of the Ulam network we divide the phase space to $n_x \times n_y$ cells ($n_x = n_y = 100$). Afterwards, N_c trajectories are propagated from each given cell j during T map iterations to obtain elements of the adjacency matrix S_{ij} for transitions to cell i (in the same manner as for the mapping (1)). The total matrix size is $N = 10^4$.

For this network we find a higher strength of resistivity of minorities, since it has a richer link structure. On Fig. 5 we plot the average of the final fraction of red nodes f_f versus the initial fraction f_i. We see here that minor opinions die out if $f_i \lesssim 0.3$. The damping factor of the Google matrix here is set to $\alpha = 0.95$, which gives a power law decay of the PageRank with a slope of 0.48 (see Ref. [22]). We also looked at the f_f versus f_i behavior for other values of the damping factor. As mentioned above, the Google matrix properties of Ulam networks are sensitive to the values of α. Nevertheless, our calculations showed, that for $0.95 < \alpha < 1$, qualitative behavior of the PROF-Sznajd model remains similar to that of Fig. 5. On the other hand, as pointed out in Ref. [10], the increase of the slope of the power law decay of the PageRank should result in a bistable behavior of the PROF and PROF-Sznajd models on social and university networks. However, this argument does not hold true for Ulam networks: although the slope of the PageRank increases with growth of α (e.g. for $\alpha = 0.98$ we have $P_j \propto 1/j^{0.7}$, while for $\alpha = 0.99$ we have $P_j \propto 1/j^{0.9}$), bistability does not emerge. Thus we conclude that a presence of bistability behavior is associated not only with the slope of the PageRank decay, but also with the intrinsic structure of the network itself.

For the PROF-Sznajd T10 model we find that the elite of the society cannot convince any elector, if its fraction is initially relatively small. In particularly, the first N_{top} nodes of the highest PageRank with the same opinion are suppressed for $N_{top}/N \lesssim 0.2$. For $N_{top}/N \gtrsim 0.2$, the elite becomes capable to influence the opinion of other electors, but the convergence process as well as the final state starts exhibiting fluctuations of a significant amplitude. These fluctuations become smaller for higher values of N_{top} and almost disappear for $N_{top}/N \gtrsim 0.7$ where the society comes

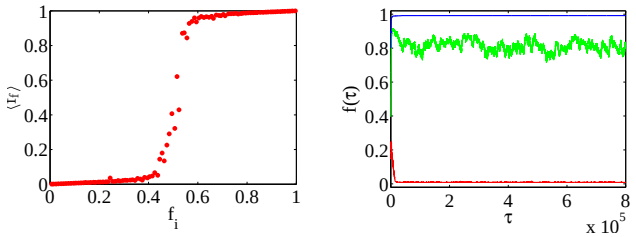


Figure 5: Left panel: The average of the final fraction of red nodes $\langle f_f \rangle$ versus the initial fraction f_i for the PROF-Sznajd model of T10 model of the typical Chirikov map ($\alpha = 0.95$, $n_x = n_y = 100$, $N = 10^4$). Here, $N_r = 10^3$ realizations with a convergence time up to $\tau = 3 \cdot 10^5$ are used to obtain the average $\langle f_f \rangle$ (the group size is $N_g = 3$). Right panel: time evolution of the fraction of red nodes $f(\tau)$ for the same model, for the initial state with the first N_{top} nodes of the highest PageRank being red: $N_{top} = 1500$ (red); $N_{top} = 4000$ (green); $N_{top} = 8000$ (blue).

to unanimity.

Finally, we shortly describe the initial and final distributions of red nodes in the coordinate space. It is of interest to consider the case of initial state with N_{top} red nodes with the highest PageRank, since for random distributions the final and initial states are homogeneously distributed over phase plane. Figure 6 shows the initial and final distributions for $N_{top} = 2200$. We find that the top elite nodes first tend to convince other members of the elite corresponding to the denser regions on the right panel of Fig. 6 with high values of the PageRank probability.

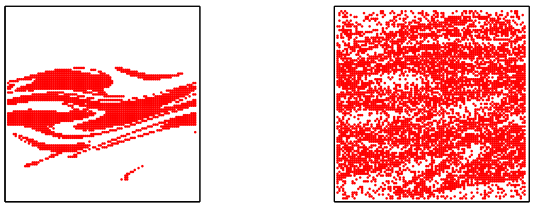


Figure 6: Coordinate distribution of red nodes in (y, x) phase space of the PROF-Sznajd $T10$ model ($\alpha = 0.95$, $n_x = n_y = 100$, $N = 10^4$); the phase plane is shown in $2\pi \times 2\pi$ square. Left panel shows the initial state with $N_{top} = 2200$ nodes of the highest PageRank being red and $f_i = 0.22$; right panel corresponds to the final state with $f_f = 0.5758$.

4. Discussion

In this work we analyzed the features of a recently proposed PageRank opinion formation model on two examples of Ulam networks. The Ulam networks generated by the discussed above one dimensional intermittency and typical Chrikov maps exhibit some intrinsic properties similar to the WWW. This fact makes the analyzes relevant to the opinion formation process in real societies. We pointed out that the elite of a society does not have a considerable influence on the decision making process of the electors for an equal mixture of conformist and tenacious society. However, the influence of the elite becomes tangible for a dominating tenacious society. In contrast to the university networks analyzed in [10] we find practically no regions of bistability behaviour for a random distribution of initial opinions. Only a dominating tenacious society shows some signs of bistability.

We also considered a generalization of the Sznajd model for Ulam networks (PROF-Sznajd model). We found here that the system still practically does not feature bistable regimes. On the basis of our studies we conclude that the PageRank decay exponent does not influence the bistability for the Ulam networks considered in this work. We argue that the chaotic maps considered generate strong stretching of small regions of phase space but do not generate significant number of loop returns. We think that this feature is different from university networks which are characterized by a significant number of loops. We presume that this internal feature of the Ulam networks is at the origin of significant difference in opinion formation on these two types of scale-free networks. The presented results can be useful for analysis of opinion formation on other types of scale-free directed networks.

Acknowledgments

We thank N.Ananikyan for useful discussions. This work was supported by the France-Armenia collaboration grant CNRS/SCS No. 24943 (IE-017) on "Classical and quantum chaos" and EC FET Open project NADINE N288956. L.C.

gratefully acknowledges the funding by the Conseil Régional de Bourgogne and FP7/2007-2013 grant No. 205025-IPERA.

References

[1] S. Galam, Int. J. Mod. Phys. C 19 (2008) 409.
[2] S. Galam, J. Math. Psych. 30 (1986) 426.
[3] T.M. Liggett, Stochastic Interacting Systems: Contact, Voter and Exclusion Processes, Springer, Berlin, 1999.
[4] S. Galam, Europhys. Lett. 70 (2005) 705.
[5] P.L. Krapivsky, S. Redner, E. Ben-Naim, A Kinetic View of Statistical Physics, Cambridge University Press, Cambridge, UK, 2010.
[6] C. Castellano, S. Fortunato, V. Loreto, Rev. Mod. Phys. 81 (2009) 591.
[7] Wikipedia, LiveJournal. http://en.wikipedia.org/wiki/LiveJournal.
[8] Wikipedia, Facebook. http://en.wikipedia.org/wiki/Facebook.
[9] Wikipedia, Twitter. http://en.wikipedia.org/wiki/Twitter.
[10] V. Kandiah, D. L. Shepelyansky, Physica A 391 (2012) 5779.
[11] S. Brin, L. Page, Comput. Netw. ISDN Syst., 30 (1998) 107.
[12] A.M. Langville, C.D. Meyer, Googles PageRank and Beyond: The Science of Search Engine Rankings, Princeton University Press, Princeton (2006).
[13] F. Radicchi, S. Fortunato, B. Markines, A. Vespignani, Phys. Rev. E 80 (2009) 056103.
[14] J.D. West, T.C. Bergstrom, C.T. Bergstrom, Coll. Res. Lib. 71 (2010) 236.
[15] D.L.Shepelyansky, O.V.Zhirov, Phys. Lett. A 374 (2010) 3206.
[16] L. Ermann, D.L. Shepelyansky, Acta Phys. Pol. A 120 (6A) (2011) A158.
[17] S.M. Ulam, A Collection of mathematical problems, Vol. 8 of Interscience tracs in pure and applied mathematics, Interscience, New York, (1960) p. 73.
[18] Z. Kovacs and T. Tel, Phys. Rev. A 4641 (1989) 40.
[19] Z. Kaufmann, H. Lustfeld, and J. Bene, Phys. Rev. E 1416 (1996) 53.
[20] G. Froyland, R. Murray, and D. Terhesiu, Phys. Rev. E 76 (2007) 036702.
[21] D. Terhesiu and G. Froyland, Nonlinearity 21 (2008) 1953.
[22] D. L. Shepelyansky, O. V. Zhirov, Phys. Rev. E 81 (2010) 036213 .
[23] L. Ermann, D. L. Shepelyansky, Phys. Rev E. 81 (2010) 03622.
[24] B. V. Chirikov, Research Concerning the Theory of Nonlinear Resonance and Stochasticity: Preprint No. 267 (Institute of Nuclear Physics, Novosibirsk, 1969) ([English translation: CERN Trans. 71-40, Geneva (1971)]).
[25] K.M. Frahm and D.L. Shepelyansky, Phys. Rev. E 80 (2009) 016210.
[26] Y. Pomeau and P. Manneville, Commun. Math. Phys. 74 (1980) 189.
[27] T. Geisel, J. Nierwetberg, and A. Zacherl, Phys. Rev. Lett. 54 (1985) 616.
[28] A. S. Pikovsky, Phys. Rev. A 43 (1991) 3146.
[29] R. Artuso and C. Manchein, Phys. Rev. E 80 (2009) 036210.
[30] R. Artuso, G. Cristadoro, Phys. Rev. Lett. 90 (2003) 244101.
[31] K. Sznajd-Weron, J. Sznajd, Int. J. Mod. Phys. C 11 (2000) 1157.
[32] C. Castellano, S. Fortunato, V. Loreto, Rev. Mod. Phys. 81 (2009) 591.

Google matrix of the citation network of Physical Review

Klaus M. Frahm,[1] Young-Ho Eom,[1] and Dima L. Shepelyansky[1]

[1] *Laboratoire de Physique Théorique du CNRS, IRSAMC, Université de Toulouse, UPS, 31062 Toulouse, France*

(Dated: October 21, 2013)

We study the statistical properties of spectrum and eigenstates of the Google matrix of the citation network of Physical Review for the period 1893 - 2009. The main fraction of complex eigenvalues with largest modulus is determined numerically by different methods based on high precision computations with up to $p = 16384$ binary digits that allows to resolve hard numerical problems for small eigenvalues. The nearly nilpotent matrix structure allows to obtain a semi-analytical computation of eigenvalues. We find that the spectrum is characterized by the fractal Weyl law with a fractal dimension $d_f \approx 1$. It is found that the majority of eigenvectors are located in a localized phase. The statistical distribution of articles in the PageRank-CheiRank plane is established providing a better understanding of information flows on the network. The concept of ImpactRank is proposed to determine an influence domain of a given article. We also discuss the properties of random matrix models of Perron-Frobenius operators.

PACS numbers: 89.75.Hc, 89.20.Hh, 89.75.Fb

I. INTRODUCTION

The development of Internet led to emergence of various types of complex directed networks created by modern society. The size of such networks grows rapidly going beyond ten billions in last two decades for the World Wide Web (WWW). Thus the development of mathematical tools for the statistical analysis of such networks becomes of primary importance. In 1998, Brin and Page proposed the analysis of WWW on the basis of PageRank vector of the associated Google matrix constructed for a directed network [1]. The mathematical foundations of this analysis are based on Markov chains [2] and Perron-Frobenius operators [3]. The PageRank algorithm allows to compute the ranking of network nodes and is known to be at the heart of modern search engines [4]. However, in many respects the statement of Brin and Page that *"Despite the importance of large-scale search engines on the web, very little academic research has been done on them"* [1] still remains valid at present. In our opinion, this is related to the fact that the Google matrix G belongs to a new class of operators which had been rarely studied in physical systems. Indeed, the physical systems are usually described by Hermitian or unitary matrices for which the Random Matrix Theory [5] captures many universal properties. In contrast, the Perron-Frobenium operators and Google matrix have eigenvalues distributed in the complex plane belonging to another class of operators.

The Google matrix is constructed from the adjacency matrix A_{ij} which has unit elements if there is a link pointing from node j to node i and zero otherwise. Then the matrix of Markov transitions is constructed by normalizing elements of each column to unity ($S_{ij} = A_{ij}/\sum_i A_{ij}$, $\sum_j S_{ij} = 1$) and replacing columns with only zero elements (*dangling nodes*) by $1/N$, with N being the matrix size. After that the Google matrix of the network takes the form [1, 4]:

$$G_{ij} = \alpha S_{ij} + (1-\alpha)/N \ . \qquad (1)$$

The damping parameter α in the WWW context describes the probability $(1-\alpha)$ to jump to any node for a random surfer. For WWW the Google search engine uses $\alpha \approx 0.85$ [4]. The PageRank vector P_i is the right eigenvector of G at $\lambda = 1$ ($\alpha < 1$). According to the Perron-Frobenius theorem [3], P_i components are positive and represent the probability to find a random surfer on a given node i (in the stationary limit) [4]. All nodes can be ordered in a decreasing order of probability $P(K_i)$ with highest probability at top values of PageRank index $K_i = 1, 2, ...$

The distribution of eigenvalues of G can be rather nontrivial with appearance of the fractal Weyl law and other unusual properties (see e.g. [6, 7]). For example, a matrix G with random positive matrix elements, normalized to unity in each column, has $N-1$ eigenvalues λ concentrated in a small radius $|\lambda| < 1/\sqrt{3N}$ and one eigenvalue $\lambda = 1$ (see below in section VII). Such a distribution is drastically different from the eigenvalue distributions found for directed networks with algebraic distribution of links [8] or those found numerically for other directed networks including WWW of universities [9, 10], Linux Kernel and Twitter networks [11, 12], Wikipedia networks [13, 14]. In fact even the Albert-Barabási model of preferential attachment [16] still generates the complex spectrum of λ with a large gap ($|\lambda| < 1/2$) [8] being very different from the gapless and strongly degenerate G spectrum of WWW of British universities [10] and Wikipedia [13, 14]. Thus it is useful to get a deeper understanding of the spectral properties of directed networks and to develop more advanced models of complex networks which have a spectrum similar to such networks as British universities and Wikipedia.

With the aim to understand the spectral properties of Google matrix of directed networks we study here the Citation Network of Physical Review (CNPR) for the whole period up to 2009 [15]. This network has $N = 463348$ nodes (articles) and $N_\ell = 4691015$ links. Its network structure is very similar to the tree network since the

citations are time ordered (with only a few exceptions of mutual citations of simultaneously published articles). As a result we succeed to develop powerful tools which allowed us to obtain the spectrum of G in semi-analytical way. These results are compared with the spectrum obtained numerically with the help of the powerful Arnoldi method (see its description in [17, 18]). Thus we are able to get a better understanding of the spectral properties of this network. Due to time ordering of article citations there are strong similarities between the CNPR and the network of integers studied recently in [19].

We note that the PageRank analysis of the CNPR had been performed in [20, 21],[22] showing its efficiency in determining the influential articles of Physical Review. The citation networks are rather generic (see e.g. [23]) and hence the extension of PageRank analysis of such networks is an interesting and important task. Here we put the main accent on the spectrum and eigenstates properties of the Google matrix of the CNPR but we also discuss the properties of two-dimensional (2D) ranking on PageRank-CheiRank plane developed recently in [24, 25],[26]. We also analyze the properties of ImpactRank which shows a domain of influence of a given article.

In addition to the whole CNPR we also consider the CNPR without Rev. Mod. Phys. articles which has $N = 460422$, $N_\ell = 4497707$. If in the whole CNPR we eliminate future citations (see description below) then this triangular CNPR has $N = 463348$, $N_\ell = 4684496$. Thus on average we have approximately 10 links per node. The network includes all articles of Physical Review from its foundation in 1893 till the end of 2009.

The paper is composed as follows: in Section II we present a detailed analysis of the Google matrix spectrum of CNPR, the fractal Weyl law is discussed in Section III, properties of eigenstates are discussed in Section IV, CheiRank versus PageRank distributions are considered in Section V, properties of impact propagation through the network are studied in Section VI, certain random matrix models of Google matrix are studied in Section VII, the discussion of the results is given in Section VIII.

II. EIGENVALUE SPECTRUM

The Google matrix of CNPR is constructed on the basis of Eq.(1) using citation links from one article to another (see also [22]). The matrix structure for different order representations of articles is shown in Fig. 1. In the top left panel all articles are ordered by time that generates almost perfect triangular structure corresponding to time ordering of citations. Still there are a few cases with joint citations of articles which appear almost at the same time. This breaks the triangular structure but the weight of such cases is small and we will see that with a good approximation one can neglect such links in a first approximation. The triangular matrix structure is also well visible in the middle left panel where articles are time ordered within each Phys. Rev. journal. The left bottom panel shows the matrix elements for each Phys

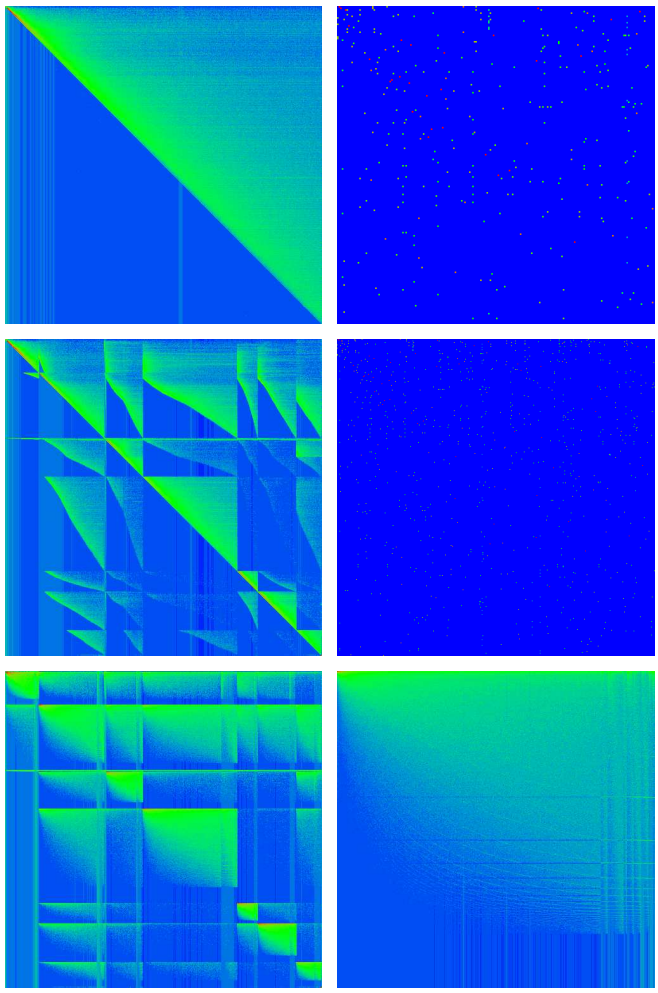


FIG. 1: (Color online) Different order representations of the Google matrix of the CNPR ($\alpha = 1$). *Left column:* The top panel shows the density of matrix elements $G_{tt'}$ in the basis of the publication time index t (and t'). The middle panel shows the density of matrix elements in the basis of journal ordering according to: Phys. Rev. Series I, Phys. Rev., Phys. Rev. Lett., Rev. Mod. Phys., Phys. Rev. A, B, C, D, E, Phys. Rev. STAB and Phys. Rev. STPER with time ordering inside each journal. The bottom panel shows the same as middle panel but with PageRank index ordering inside each journal. Note that the journals Phys. Rev. Series I, Phys. Rev. STAB and Phys. Rev. STPER are not clearly visible due to a small number of published papers. Also Rev. Mod. Phys. appears only as a thin line with 2-3 pixels (out of 500) due to a limited number of published papers. The three left panels and the bottom right panel show the coarse-grained density of matrix elements done on 500×500 square cells for the entire network. *Right column:* Matrix elements $G_{KK'}$ are shown in the basis of PageRank index K (and K') with the range $1 \le K, K' \le 200$ (top panel); $1 \le K, K' \le 400$ (middle panel); $1 \le K, K' \le N$ (bottom panel). Color shows the amplitude (or density) of matrix elements G changing from blue for zero value to red at maximum value. The PageRank index K is determined from the PageRank vector at $\alpha = 0.85$.

Rev journal when inside each journal the articles are ordered by their PageRank index K. The right panels show the matrix elements of G on different scales, when all articles are ordered by the PageRank index K.

The dependence of number of no-zero links N_G, between nodes with PageRank index being less than K, on K is shown in Fig. 2 (left panel). We see that compared to the other networks of universities, Wikipedia and Twitter studied in [13] we have for CNPR the lowest values of N_G/K practically for all available K values. This reflects weak links between top PageRank articles of CNPR being in contrast with Twitter which has very high interconnection between top PageRank nodes. Since the matrix elements $G_{KK'}$ are inversely proportional to the number of links we have very strong average matrix elements for CNPR at top K values (see Fig. 2 (right panel)).

In the following we present the results of numerical and analytical analysis of the spectrum of the CNPR matrix G.

A. Nearly nilpotent matrix structure

The triangular structure of the $CNPR$ Google matrix in time index (see Fig. 1) has important consequences for the eigenvalue spectrum λ defined by the equation for the eigenstates $\psi_i(j)$:

$$\sum_{j'} G_{jj'}\psi_i(j') = \lambda_i \psi_i(j) \; . \qquad (2)$$

The spectrum of G at $\alpha = 1$, or the spectrum of S, obtained by the Arnoldi method [17, 18] with the Arnoldi dimension $n_A = 8000$, is shown in Fig. 3. For comparison we also show the case of reduced CNPR without Rev. Mod. Phys.. We see that the spectrum of the reduced case is rather similar to the spectrum of the full CNPR.

The matrix S can be decomposed on invariant subspaces S_{ss}, the core space S_{cc} with fully connected nodes, and the coupling block S_{sc}, thus being presented in the form [10]:

$$S = \begin{pmatrix} S_{ss} & S_{sc} \\ 0 & S_{cc} \end{pmatrix} \; . \qquad (3)$$

The subspace-subspace block S_{ss} is actually composed of many diagonal blocks for each of the invariant subspaces. Each of these blocks corresponds to a column sum normalized matrix of the same type as G and has therefore at least one unit eigenvalue thus explaining the high degeneracy of S eigenvalue $\lambda = 1$. This structure is discussed in detail in [10].

A network with a similar triangular structure, constructed from factor decompositions of integer numbers, was previously studied in [19]. There it was analytically shown that the corresponding G has only a small number of non-vanishing eigenvalues and that the numerical diagonalization methods, including the Arnoldi method, are facing subtle difficulties of numerical stability due to large Jordan blocks associated to the highly degenerate zero eigenvalue. The numerical diagonalization of these Jordan blocks is highly sensitive to numerical round-off errors. For example a perturbed Jordan block of dimension D associated to the eigenvalue zero and with a perturbation ε in the opposite corner has eigenvalues on a complex circle of radius $\varepsilon^{1/D}$ [19] which may became very large for sufficient large D even for $\varepsilon \sim 10^{-15}$. Therefore in presence of many such Jordan blocks the numerical diagonalization methods create rather big "artificial clouds" of incorrect eigenvalues.

In the examples studied in [19] these clouds extended up to eigenvalues $|\lambda| \approx 0.01$. The spectrum for the Physical Review network shown in Fig. 3 shows also a sudden increase of the density of eigenvalues below $|\lambda| \approx 0.3-0.4$ and one needs to be concerned if these eigenvalues are "real" or only an artifact of the same type of numerical instability. Actually, we find that the eigenvalues of Fig. 3 below $|\lambda| \approx 0.3 - 0.4$ are changed completely in a random way if we apply to the network or the numerical algorithm certain transformations or modifications which are *mathematically neutral* but which have a different effect on the numerical round-off errors (e.g. a permutation of the network nodes, keeping the same network-link structure, or simply changing the evaluation order of the sums used for the scalar products between vectors in the Gram-Schmidt orthogonalization for the Arnoldi method). This clearly indicates that these eigenvalues are not reliable due to problems in the numerical evaluation.

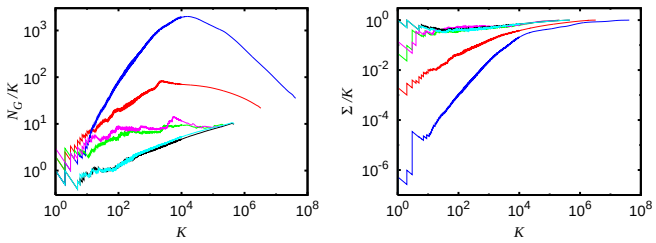


FIG. 2: (Color online) *Left panel:* dependence of the linear density N_G/K of nonzero elements of the adjacency matrix among top PageRank nodes on the PageRank index K for the networks of Twitter (blue curve), Wikipedia (red curve), Oxford University 2006 (magenta curve), Cambridge University 2006 (green curve), with data taken from Ref. [12], and Physical Review all journals (cyan curve) and Physical Review without Rep. Mod. Phys. (black curve) (curves from top to bottom at $K = 100$). *Right panel:* dependence of the quantity Σ/K on the PageRank index K with $\Sigma = \sum_{K_1<K,\, K_2<K} G_{K_1,K_2}$ being the weight of the Google matrix elements inside the $K \times K$ square of top PageRank indexes. The curves correspond to the same networks as in the left panel: Physical Review without Rep. Mod. Phys. (black curve), Physical Review all journals (cyan curve), Oxford University 2006 (magenta curve), Cambridge University 2006 (green curve), Wikipedia (red curve), and Twitter (blue curve) (curves from top to bottom at $K = 1$).

The theory of [19] is based on the exact triangular structure of the matrix S_0 which appears in the representation of $S = S_0 + ed^T/N$ (see also below Eq. 4). In fact the matrix S_0 is obtained from the adjacency matrix by normalizing the sum of the elements in non-vanishing columns to unity and simply keeping at zero vanishing columns. For the network of integers [19] this matrix is nilpotent with $S_0^l = 0$ for a certain modest value of l being much smaller than the network size $l \ll N$. However, for CNPR the matrix S_0 is not exactly nilpotent despite the overall triangular matrix structure visible in Fig. 1. Even though most of the non-vanishing matrix elements $(S_0)_{tt'}$ (whose total number is equal to the number of links $N_\ell = 4691015$) are in the upper triangle $t < t'$ there are a few non-vanishing elements in the lower triangle $t > t'$ (whose number is 12126 corresponding to 0.26 % of the total number of links [27]). The reason is that in most cases papers cite other papers published earlier but in certain situations for papers with close publication date the citation order does not always coincide with the publication order. In some cases two papers even mutually cite each other. In the following we will call these cases "future citations". The rare non-vanishing matrix elements due to future citations are not visible in the coarse grained matrix representation of Fig. 1 but they are responsible for the fact that S_0 of CMPR is not nilpotent and that there are also a few invariant subspaces. On a purely triangular network one can easily show the absence of invariant subspaces (smaller than the full network size) when taking into account the extra columns due to the dangling nodes.

However, despite the effect of the future citations the matrix S_0 is still partly nilpotent. This can be seen by multiplying a uniform initial vector e (with all components being 1) by the matrix S_0 and counting after each iteration the number N_i of non-vanishing entries [28] in the resulting vector $S_0^i e$. For a nilpotent matrix S_0 with $S_0^l = 0$ the number N_i becomes obviously zero for $i \geq l$. On the other hand, since the components of e and the non-vanishing matrix elements of S_0 are positive, one can easily verify that the condition $S_0^l e = 0$ for some value l also implies $S_0^l \psi = 0$ for an arbitrary initial (even complex) vector ψ which shows that S_0 must be nilpotent with $S_0^l = 0$.

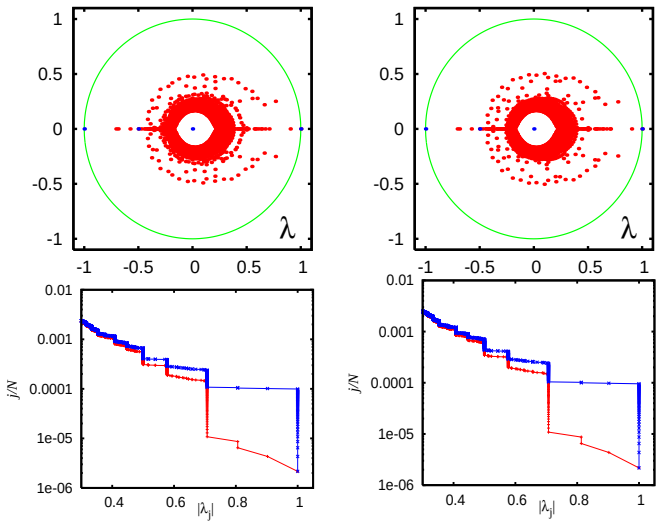


FIG. 3: (Color online) Spectrum of S for CNPR (reduced CNPR without Rev. Mod. Phys.) shown on left panels (right panels). *Top panels:* Subspace eigenvalues (blue dots) and core space eigenvalues (red dots) in λ-plane (green curve shows unit circle); there are 27 (26) invariant subspaces, with maximal dimension 6 (6) and the sum of all subspace dimensions is $N_s = 71$ (75). The core space eigenvalues are obtained from the Arnoldi method applied to the core space subblock S_{cc} of S with Arnoldi dimension $n_A = 8000$ as explained in Ref. [10] and using standard double-precision arithmetic. *Bottom panels:* Fraction j/N of eigenvalues, shown in a logarithmic scale, with $|\lambda| > |\lambda_j|$ for the core space eigenvalues (red bottom curve) and all eigenvalues (blue top curve) from raw data of top panels. The number of eigenvalues with $|\lambda_j| = 1$ is 45 (43) of which 27 (26) are at $\lambda_j = 1$; this number is identical to the number of invariant subspaces which have each one unit eigenvalue.

In Fig. 4 we see that for the CNPR the value of N_i saturates at a value $N_{sat} = 273490$ for $i \geq 27$ which is 59% of the total number of nodes $N = 463348$ in the network. On one hand the (small) number of future citations ensures that the saturation value of N_i is not zero but on the other hand it is smaller than the total number of nodes by a macroscopic factor. Mathematically the first iteration $e \to S_0 e$ removes the nodes corresponding to empty (vanishing) lines of the matrix S_0 and the next iterations remove the nodes whose lines in S_0 have become empty after having removed from the network the non-occupied nodes due to previous iterations. For each node removed during this iteration process one can construct a vector belonging to the Jordan subspace of S_0 associated to the eigenvalue 0. In the following we call this subspace *generalized kernel*. It contains all eigenvectors of S_0^j associated to the eigenvalue 0 where the integer j is the size of the largest 0-eigenvalue Jordan block. Obviously the dimension of this generalized kernel of S_0 is larger or equal than $N - N_{sat} = 189857$ but we will see later that its actual dimension is even larger and quite close to N. We will argue below that most (but not all) of the vectors in the generalized kernel of S_0 also belong

to the generalized kernel of S which differs from S_0 by the extra contributions due to the dangling nodes. The high dimension of the generalized kernel containing many large 0-eigenvalue Jordan subspaces explains very clearly the numerical problem due to which the eigenvalues obtained by the double-precision Arnoldi method are not reliable for $|\lambda| < 0.3 - 0.4$.

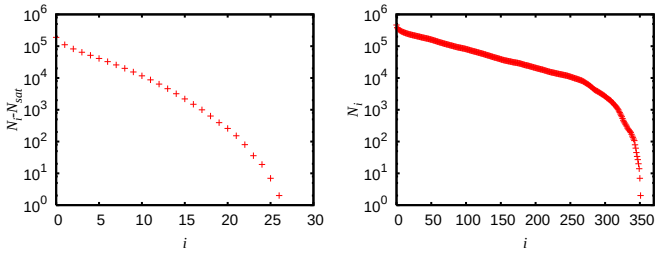


FIG. 4: (Color online) Number of occupied nodes N_i (i.e. positive elements) in the vector $S_0^i\, e$ versus iteration number i (red crosses) for the CNPR (left panel) and the triangular CNPR (right panel). In both cases the initial value is the network size $N_0 = N = 463348$. For the CNPR N_i saturates at $N_i = N_{sat} = 273490 \approx 0.590N$ for $i \geq 27$ while for the triangular CNPR N_i saturates at $N_i = 0$ for $i \geq 352$ confirming the nilpotent structure of S_0. In the left panel the quantity $N_i - N_{sat}$ is shown in order to increase visibility in the logarithmic scale.

B. Spectrum for the triangular CNPR

In order to extend the theory for the triangular matrices developed in [19] we consider the triangular CNPR obtained by removing all future citation links $t' \to t$ with $t \geq t'$ from the original CNPR. The resulting matrix S_0 of this reduced network is now indeed nilpotent with $S_0^{l-1} \neq 0$, $S_0^l = 0$ and $l = 352$ which is much smaller than the network size. This is clearly seen from Fig. 4 showing that N_i, calculated from the triangular CNPR, indeed saturates at $N_i = 0$ for $i \geq 352$. According to the arguments of [19], and additional demonstrations given below, there are at most only $l = 352$ non-zero eigenvalues of the Google matrix at $\alpha = 1$. This matrix has the form

$$S = S_0 + (1/N)\, e\, d^T \qquad (4)$$

where d and e are two vectors with $e(n) = 1$ for all nodes $n = 1, \ldots, N$ and $d(n) = 1$ for dangling nodes n (corresponding to vanishing columns in S_0) and $d(n) = 0$ for the other nodes. In the following we call d the dangling vector. The extra contribution $e\, d^T/N$ just replaces the empty columns (of S_0) with $1/N$ entries at each element and d^T is the line vector obtained as the transpose of the column vector d.

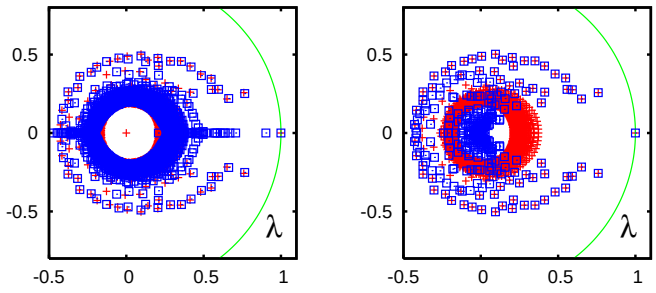


FIG. 5: (Color online) *Left Panel:* Comparison of the core space eigenvalue spectrum of S for CNPR (blue squares) and triangular CNPR (red crosses). Both spectra are calculated by the Arnoldi method with $n_A = 4000$ and standard double-precision. *Right Panel:* Comparison of the numerically determined non-vanishing 352 eigenvalues obtained from the representation matrix (12) (blue squares) with the spectrum of triangular CNPR (red crosses) already shown in the left panel. Numerics is done with standard double-precision.

In the left panel of Fig. 5 we compare the core space spectrum of S for CNPR and triangular CNPR (data are obtained by the Arnoldi method with $n_A = 4000$ and standard double-precision). We see that the largest complex eigenvalues are rather close for both cases but in the full network we have a lot of eigenvalues on the real axis (with $\lambda < -0.3$ or $\lambda > 0.4$) which are absent for the triangular CNPR. Furthermore, both cases suffer from the same problem of numerical instability due to large Jordan blocks.

Let us briefly remind the analytical theory of [19] for pure triangular networks with a nilpotent matrix S_0 such that $S_0^l = 0$. For this we define the coefficients:

$$c_j = d^T S_0^j\, e/N \quad , \quad b_j = e^T S_0^j\, e/N \qquad (5)$$

which are non-zero only for $j = 0, 1, \ldots, l-1$. The fact that the non-vanishing columns of S_0 are sum normalized and that the other columns (corresponding to dangling nodes) are zero can be written as: $e^T S_0 = e^T - d^T$ implying $d^T = e^T(\mathbb{1} - S_0)$. Using this identify and the fact that $S_0^k = 0$ for $k \geq l$ we find:

$$\sum_{k=j}^{l-1} c_k = d^T (\mathbb{1} - S_0)^{-1} S_0^j\, e/N = e^T S_0^j\, e/N = b_j \qquad (6)$$

and in particular for $j = 0$ we obtain the sum rule $\sum_{k=0}^{l-1} c_k = 1$ and for $j = l-1$ the identity $b_{l-1} = c_{l-1}$.

Consider now a right eigenvector ψ of S with eigenvalue λ. If $d^T\psi = 0$ we find from (4) that ψ is also an eigenvector of S_0 and since S_0 is nilpotent the eigenvalue must be $\lambda = 0$. Therefore for $\lambda \neq 0$ we have necessarily $d^T\psi \neq 0$ and with the appropriate normalization of ψ we have $d^T\psi = 1$ that implies together with the eigenvalue equation: $\psi = (\lambda\mathbb{1} - S_0)^{-1}\, e/N$ where the matrix inverse is well defined for $\lambda \neq 0$. The eigenvalue is determined

by the condition:

$$0 = \lambda^l(1 - d^T\psi) = \lambda^l \left(1 - d^T \frac{\mathbb{1}}{\lambda\mathbb{1} - S_0} e/N\right) . \quad (7)$$

Since S_0 is nilpotent we may expand the matrix inverse in a finite series and therefore the eigenvalue λ is the zero of the reduced polynomial of degree l:

$$\mathcal{P}_r(\lambda) = \lambda^l - \sum_{j=0}^{l-1} \lambda^{l-1-j}\, c_j \quad (8)$$

where the coefficients c_j are given by (5). Using $d^T = e^T(\mathbb{1} - S_0)$ we may rewrite (7) in the form:

$$0 = \lambda^l \left(1 - e^T \frac{\mathbb{1} - S_0}{\lambda\mathbb{1} - S_0} e/N\right) = (\lambda - 1)\lambda^l\, e^T \frac{\mathbb{1}}{\lambda\mathbb{1} - S_0} e/N \quad (9)$$

which gives another expression for the reduced polynomial:

$$\mathcal{P}_r(\lambda) = (\lambda - 1) \sum_{j=0}^{l-1} \lambda^{l-1-j}\, b_j \quad (10)$$

using the coefficients b_j and confirming explicitly that $\lambda = 1$ is indeed an eigenvalue of S. The expression (10) can also be obtained by a direct calculation from (6) and (8).

Since the reduced polynomial has at most l zeros λ_j ($\neq 0$ since $c_{l-1} = b_{l-1} \neq 0$) we find that there are at most l non-vanishing eigenvalues of S given by these zeros. They can also be obtained as the eigenvalues of a "small" $l \times l$ matrix. To see this let us define the following set of vectors v_j for $j = 1, \ldots, l$ by $v_j = c_{j-1}^{-1}\, S_0^{j-1}\, e/N$ where we have chosen to apply the prefactor c_{j-1}^{-1} to the vector $S_0^{j-1}\, e/N$ [29]. From (4) and (5) one finds that Sv_j can be expanded in the other vectors v_k as

$$Sv_j = \frac{c_j}{c_{j-1}}\, v_{j+1} + c_0\, v_1 = \sum_{k=1}^{l} \bar{S}_{kj}\, v_k \quad (11)$$

where $\bar{S}_{kj}$ are the matrix elements of the $l \times l$ representation matrix

$$\bar{S} = \begin{pmatrix} c_0 & c_0 & \cdots & c_0 & c_0 \\ c_1/c_0 & 0 & \cdots & 0 & 0 \\ 0 & c_2/c_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & c_{l-1}/c_{l-2} & 0 \end{pmatrix} . \quad (12)$$

Note that for the last vector v_l we have $Sv_l = c_0\, v_1$ since $c_l = 0$ and therefore the matrix $\bar{S}$ provides a closed and mathematically exact representation of S on the l-dimensional subspace generated by $v_1, \ldots, v_l$. Furthermore one can easily verify (by a recursive calculation in l) that the characteristic polynomial of $\bar{S}$ coincides with the reduced polynomial (8). Therefore numerical diagonalization of $\bar{S}$ provides an alternative method to compute the non-vanishing eigenvalues of S. In principle one can also determine directly the zeros of the reduced polynomial by the Newton-Maehly method and in [19] this was indeed done for cases with very modest values of $l \leq 29$. However, here for the triangular CNPR we have $l = 352$ and the coefficients c_j become very small, especially: $c_{l-1} \approx 3.6 \times 10^{-352}$ a number which is (due to the exponent) outside the range of 64 bit standard double-precision numbers (IEEE 754) with 52 bits for the mantissa, 10 bits for the exponent (with respect to 2) and two bits for the signs of mantissa and exponent. This exponent range problem is not really serious and can for example be circumvented by a smart reformulation of the algorithm to evaluate the ratio $\mathcal{P}_r(\lambda)/\mathcal{P}_r'(\lambda)$ using only ratios c_j/c_{j-1} which do not have this exponent range problem. However, it turns out that in this approach the convergence of the Newton-Maehly method using double-precision arithmetic is very bad for many zeros and does not provide reliable results. Below we show how this problem can be solved using high precision calculations but for the moment we mention that one may also try another approach by diagonalizing numerically the representation matrix $\bar{S}$ given in (12) which also depends on the ratios c_j/c_{j-1}.

In the right panel of Fig. 5 we compare the numerical double-precision spectra of $\bar{S}$ with the results of the Arnoldi method with double-precision and the uniform initial vector e as start vector for the Arnoldi iterations. We remind that the Arnoldi method determines an orthonormal set of vectors $\zeta_1, \zeta_2, \zeta_3, \ldots, \zeta_{n_A}$ where the first vector ζ_1 is obtained by normalizing a given initial vector and ζ_{j+1} is obtained by orthonormalizing $S\zeta_j$ to the previous vectors determined so far. It is obvious due to (11) that for the initial uniform vector e each ζ_j is given by a linear combination of the vectors v_k with $k = 1, \ldots, j$. Since the subspace of v_k for $k = 1, \ldots, l$ is closed with respect to applications of S the Arnoldi method should, in theory, break off at $n_A = l$ with a zero coupling element. The latter is given as the norm of $S\zeta_l$ othogonalized to $\zeta_1, \ldots, \zeta_l$ and if this norm vanishes the vector ζ_{l+1} cannot be constructed and the Arnoldi method has completely explored an S-invariant subspace of dimension l.

However, due to a strong effect of round-off errors and the fact that the vectors v_j are numerically "nearly" linearly dependent the last coupling element does not vanish numerically (when using double-precision) and the Arnoldi method produces a cloud of numerically incorrect eigenvalues due to the Jordan blocks which are mathematically outside the representation space (defined by the vectors v_j) but which are still explored due to round-off errors and clearly visible in Fig. 5. The double-precision spectrum of $\bar{S}$ seems to provide well defined eigenvalues in the range where the Arnoldi method produces the "Jordan block cloud" but outside this cloud both spectra coincide only partly, mainly for the eigenvalues with

largest modulus and positive real part. For the eigenvalues with negative real part there are considerable deviations. As we will see later the eigenvalues produced by the Arnoldi method at double-precision are reliable provided that they are well *outside* the Jordan block cloud of incorrect eigenvalues. Therefore the deviations outside the Jordan block cloud show that the numerical double-precision diagonalization of the representation matrix $\bar{S}$ is not reliable as well but here the effect of numerical errors is quite different as for the Arnoldi method as it is explained below.

We have tried to determine the zeros of the reduced polynomial using higher precision numbers with 80 or even 128 bits (quadruple precision) which helps to solve the (minor) exponent range problem because these formats use more bits for the exponent. However, there are indeed two other serious numerical problems. First it turns out that in a certain range of the complex plane around $\mathrm{Re}(\lambda) \approx -0.1$ to -0.2 and $\mathrm{Im}(\lambda) \leq 0.1$ the numerical evaluation of the polynomial suffers in a severe way from an alternate sign problem with a strong loss of significance. Second the zeros of the polynomial depend in a very sensitive way on the precision of the coefficients c_j (see below). We have found that even 128 bit numbers are not sufficient to obtain all zeros with a reasonable graphical precision.

Therefore we use the very efficient GNU Multiple Precision Arithmetic Library (GMP library) [30]. With this library one has 31 bits for the exponent and one may chose an arbitrary number of bits for the mantissa. We find that using 256 bits (binary digits) for the mantissa the complex zeros of the reduced polynomial can be determined with a precision of 10^{-18}. In this case the convergence of the Newton-Maehly method is very nice and we obtain that the sum (and product) of the complex zeros coincide with a high precision with the theoretical values c_0 (respectively: $(-1)^{l-1}c_{l-1}$) due to (8). We have also tested different ways to evaluate the polynomial, such as Horner scheme versus direct evaluation of the sum and for both methods using both expressions (8) and (10). It turns out that with 256 binary digits during the calculation the zeros obtained by the different variants of the method coincide very well within the required precision of 10^{-18}. Of course the coefficients c_j or b_j given by (5) need also to be evaluated with the precision of 256 binary digits but there is no problem of using high precision vectors since the non-vanishing matrix elements of S_0 are rational numbers that allow to perform the evaluation of the vectors $S_0^j e/N$ with arbitrary precision. We also tested a random modification of c_j according to $c_j \to c_j(1+10^{-16}X)$ where X is a random number in the interval $]-0.5, 0.5[$. This modification gives significant differences of the order of 10^{-2} to 10^{-1} for some of the complex zeros and which are well visible in the graphical representation of the spectra. Therefore, the spectrum depends in a very sensitive way on these coefficients and it is now quite clear that numerical double-precision diagonalization of $\bar{S}$, which depends according to (12) on the values c_j, cannot provide accurate eigenvalues simply because the double-precision round-off errors of c_j imply a sensitive change of eigenvalues. In particular some of the numerical eigenvalues of $\bar{S}$ differ quite strongly from the high precision zeros of the reduced polynomial.

In order to study more precisely the effect of the numerical instability of the Arnoldi method due to the Jordan blocks we also use the GMP library to increase the numerical precision of the Arnoldi method. To be precise we implement the first part of this method, the *Arnold iteration* in which the $n_A \times n_A$ Arnoldi representation matrix is determined by the Gram-Schmidt orthogonalization procedure, using high precision numbers while for the second step, the numerical diagonalization of this representation matrix, we keep the standard double-precision. It turns that only the first step is numerically critical. Once the Arnoldi representation matrix is obtained in a careful and precise way, it is numerically well conditioned and its numerical diagonalization works well with only double-precision.

In Fig. 6 we compare the exact spectrum obtained by the high precision determination of the zeros of the reduced polynomial (using 256 bits) with the spectra of the Arnold method for 52 bits (corresponding to the mantissa of double-precision numbers), 256 bits, 512 bits and 1280 bits. Here we use for the Arnoldi method a uniform initial vector and the Arnold dimension $n_A = l = 352$. In this case, as explained above, in theory the Arnoldi method should provide the exact $l = 352$ non-vanishing eigenvalues (in absence of round-off errors).

However, with the precision of 52 bits we have a considerable number of eigenvalues on a circle of radius ≈ 0.3 centered at 0.05 indicating a strong influence of round-off errors due to the Jordan blocks. Increasing the precision to 256 (or 512) bits implies that the number of correct eigenvalue increases and the radius of this circle decreases to 0.13 (or 0.1) and in particular it does not extend to all angles. We have to increase the precision of the Arnoldi method to 1280 bits to have a perfect numerical confirmation that the Arnoldi method explores the exact invariant subspace of dimension $l = 352$ and generated by the vectors v_j. In this case the eigenvalues obtained from the Arnoldi method and the high-precision zeros of the reduced polynomial coincide with an error below 10^{-14} and in particular the Arnoldi method provides a nearly vanishing coupling matrix element at the last iteration confirming that there is indeed an exact decoupling of the Arnoldi matrix and an invariant closed subspace of dimension 352.

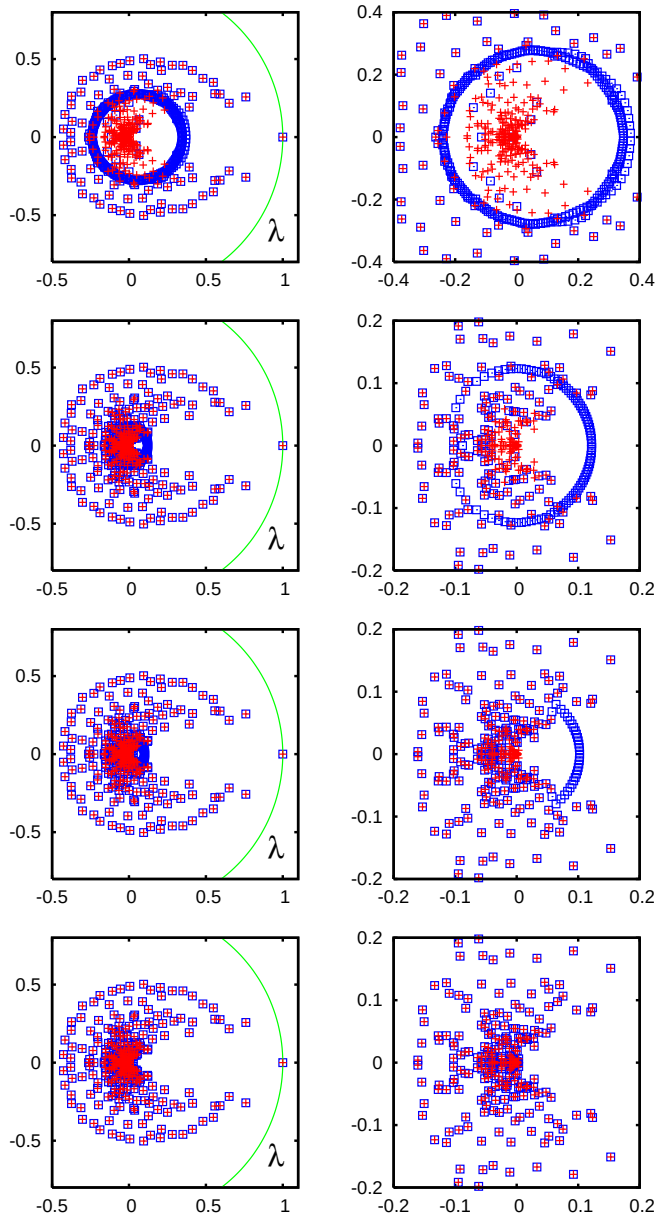


FIG. 6: (Color online) Comparison of the numerically accurate 352 non-vanishing eigenvalues of S matrix of triangular CNPR, determined by the Newton-Maehly method applied to the reduced polynomial (8) with a high-precision calculation of 256 binary digits (red crosses, all panels), with eigenvalues obtained by the Arnoldi method at different numerical precisions (for the determination of the Arnoldi matrix) for triangular CNPR and Arnoldi dimension $n_A = 352$ (blue squares, all panels). The first row corresponds to the numerical precision of 52 binary digits for standard double-precision arithmetic. The second (third, fourth) row corresponds to the precision of 256 (512, 1280) binary digits. All high precision calculations are done with the library GMP [30]. The panels in the left column show the complete spectra and the panels in the right columns show the spectra in a zoomed range: $-0.4 \leq \mathrm{Re}(\lambda), \mathrm{Im}(\lambda) \leq< 0.4$ for the first row or $-0.2 \leq \mathrm{Re}(\lambda), \mathrm{Im}(\lambda) \leq 0.2$ for the second, third and fourth rows.

The results shown in Fig.6 clearly confirm the above theory and the scenario of the strong influence of Jordan blocks on the round-off errors. In particular, we find that in order to increase the numerical precision it is only necessary to implement the first step of the method, the Arnoldi iteration, using high precision numbers while the numerical diagonalization of the Arnoldi representation matrix can still be done using standard double-precision arithmetic. We also observe, that even for the case with lowest precision of 52 bits the eigenvalues obtained by the Arnoldi method are numerically accurate provided that there are well outside the circle (or cloud) of numerically incorrect eigenvalues.

C. High precision spectrum of the whole CNPR

Based on the observation that a high precision implementation of the Arnoldi method is useful for the triangular CNPR, we now apply the high precision Arnoldi method with 256, 512 and 756 bits and $n_A = 2000$ to the original CNPR. The results for the core space eigenvalues are shown in Fig. 7 where we compare the spectrum of the highest precision of 756 bits with lower precision spectra of 52, 256 and 512 bits. As in Fig. 6 for the triangular CNPR, for CNPR we also observe that the radius and angular extension of the cloud or circle of incorrect Jordan block eigenvalues decreases with increasing precision. Despite the lower number of $n_A = 2000$ as compared to $n_A = 8000$ of Fig. 3 the number of accurate eigenvalues with 756 bit precision is certainly considerably higher.

The higher precision Arnoldi method certainly improves the quality of the smaller eigenvalues, e.g. for $|\lambda| < 0.3 - 0.4$, but it also implies a strange shortcoming as far as the degeneracies of certain particular eigenvalues are concerned. This can be seen in Fig. 8 which shows the core space eigenvalues $|\lambda_j|$ versus the level number j for various values of the Arnoldi dimension and the precision. In these curves we observe flat plateaux at certain values $|\lambda_j| = 1/\sqrt{n}$ with $n = 2, 3, 4, 5, \ldots$ corresponding to degenerate eigenvalues which turn out to be real but with positive or negative values: $\lambda_j = \pm 1/\sqrt{n}$. For fixed standard double-precision arithmetic with 52 binary digits the degeneracies increase with increasing Arnoldi dimension and seem to saturate for $n_A \geq 4000$. However at the given value of $n_A = 2000$ the degeneracies *decrease* with increasing precision of the Arnoldi method. Apparently the higher precision Arnoldi method is less able to determine the correct degeneracy of a degenerate eigenvalue.

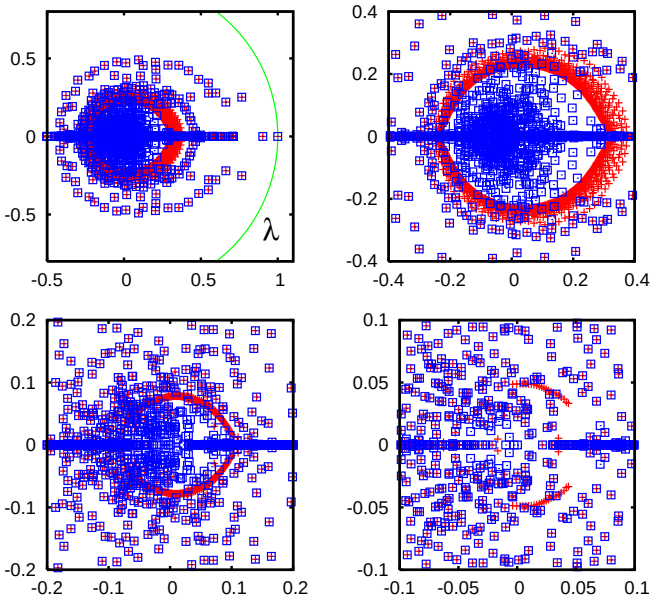


FIG. 7: (Color online) Comparison of the core space eigenvalue spectrum of S of CNPR, obtained by the high precision Arnoldi method using 768 binary digits (blue squares, all panels), with lower precision data of the Arnoldi method (red crosses). In both top panels the red crosses correspond to double-precision with 52 binary digits (extended range in left top panel and zoomed range in right top panel). In the bottom left (right) panel red crosses correspond to the numerical precision of 256 (512) binary digits. In these two cases only a zoomed range is shown. The eigenvalues outside the zoomed ranges coincide for both data sets up to graphical precision. In all cases the Arnoldi dimension is $n_A = 2000$. High precision calculations are done with the library GMP [30].

This point can be understood as follows. In theory, assuming perfect precision, the simple version of Arnoldi method used here (in contrast to more complicated block Arnoldi methods) can only determine one eigenvector for a degenerate eigenvalue. The reason is that for a degenerate eigenvalue we have a particular linear combination of the eigenvectors for this eigenvalue which contribute in any initial vector (in other words "one particular" eigenvector for this eigenvalue) and during the Arnoldi iteration this particular eigenvector will be perfectly conserved and the generated Krylov space will only contain this and no other eigenvector for this eigenvalue. However, due to round-off errors we obtain at each step new random contributions from other eigenvectors of the same eigenvalue and it is only due to these round-off errors that we can see the flat plateaux in Fig. 8. Obviously, increasing the precision reduces this round-off error effect and the flat plateaux are indeed considerably smaller for higher precisions.

The question arises about the origin of the degenerate eigenvalues in the core space spectrum. In other examples, such as the WWW for certain university networks [10], the degeneracies, especially of the leading eigenvalue 1, could be treated by separating and diagonalizing the exact subspaces and the remaining core space spectrum contained much less or nearly no degenerate eigenvalues. However, here for the CNPR we have "only" 27 subspaces with maximal dimension of 6 containing 71 nodes in total. The eigenvalues due to these subspaces are 1, -1, -0.5, 0 with degeneracies 27, 18, 4, 22 (see blue dots in the upper panels of Fig. 3). These exact subspaces exist only due to the modest number of future citation links. Even when we take care that in all cases the Arnoldi method is applied to the core space without these 71 subspace nodes, there are still remain a lot of degenerate eigenvalues in the core space spectrum.

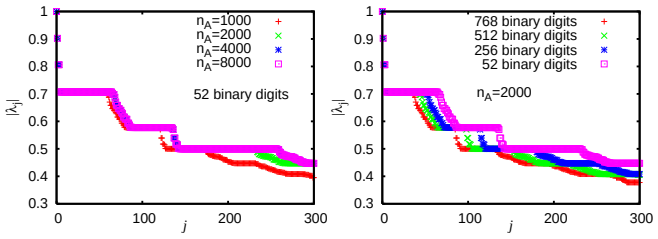


FIG. 8: (Color online) Modulus $|\lambda_j|$ of the core space eigenvalues of S of CNPR, obtained by the Arnoldi method, shown versus level number j. *Left panel:* data for standard double-precision with 52 binary digits with different Arnoldi dimensions $1000 \leq n_A \leq 8000$. *Right panel:* data for Arnoldi dimension $n_A = 2000$ with different numerical precisions between 52 and 768 binary digits.

In order to understand the mechanism of these degenerate core space eigenvalues we extend the argumentation of the last subsection for triangular CNPR to the case of nearly triangular networks. Consider again the matrix S given by Eq. (4) but now S_0 is not nilpotent. There are two groups of eigenvectors ψ of S with eigenvalue λ. The first group is characterized by the orthogonality $d^T\psi = 0$ of the eigenvector ψ with respect to the dangling vector d and the second group is characterized by the non-orthogonality $d^T\psi \neq 0$. In the following, we describe efficient methods to determine all eigenvalues of the first group and a considerable number of eigenvalues of the second group. We note that for the case of a purely triangular network the first group contains only eigenvectors for the eigenvalue 0 and the second group contains the eigenvectors for the l non-vanishing eigenvalues as discussed in the last subsection. In principle there are also complications due to generalized eigenvectors (associated to non-trivial Jordan blocks) but they appear mainly for zero eigenvalue and we for the moment do not discuss these complications.

First we note that the subspace eigenvectors of S belong to the first group because the nodes of the subspaces of S cannot contain dangling nodes which are by construction of S are linked to any other node and therefore belong to the core space. Since any subspace eigenvector ψ has non-vanishing values only for subspace nodes

being different from dangling nodes we have obviously $d^T\psi = 0$. We also note that an eigenvector of S of the first group with $d^T\psi = 0$ is due to (4) also an eigenvector of S_0 with the same eigenvalue.

For the remaining eigenvectors in the first group one might try to diagonalize the matrix S_0 and check for each eigenvector of S_0 if the identity $d^T\psi = 0$ holds in which case we would obtain an eigenvector of S of the first group but generically, and apart from the subspace eigenvectors, there is no reason that eigenvectors of S_0 with isolated non-degenerate eigenvalues obey this identity. However, if we have an eigenvalue of S_0 with a degeneracy $m \geq 2$ we may construct by suitable linear combinations $m-1$ linearly independent eigenvectors of S_0 which also obey $d^T\psi = 0$ and therefore this eigenvalue with degeneracy m of S_0 is also an eigenvalue with degeneracy $m-1$ of S. In order to determine the degenerate eigenvalues of S_0 it is useful to determine the subspaces of S_0 which (in contrast to the subspaces of S) may contain dangling nodes. Actually, each dangling node is a trivial subspace of dimension 1 with a network matrix of size 1×1 and being zero. Explicitly we have implemented the following procedure: first we determine the subspaces of S (with 71 nodes in total) and remove these nodes from the network. Then we determine all subspaces of S_0 whose dimension is below 10. Each time such a subspace is found its nodes are immediately removed from the network. When we have tested in a first run all nodes as potential subspace nodes the procedure is repeated until no new subspaces of maximal dimension 10 are found since removal of former subspaces may have created new subspaces. Then the limit size of 10 is doubled to 20, 40, 80 etc. to ensure that we do not miss large subspaces. However, for the CNPR it turns out that the limit size of 10 allows to find all subspaces. In our procedure a subsequently found subspace may potentially have links to a former subspace leading to a block-triangular (and not block-diagonal structure as it was done in ref. [10]). This method to determine "relative" subspaces of a network already reduced by former subspaces is more convenient for the CNPR which is nearly triangular and it allows also to determine correctly all subspace eigenvalues by diagonalizing each relative subspace network. The removal of subspace nodes of S and S_0 reduces the network size from $N = 463348$ to 404959. In the next step we remove in the same way the subspaces of the transpose S_0^T of S_0 (since the eigenvalues of S_0^T and S_0 are the same) which reduces the network size furthermore to 90965. In total this procedure provides a block triangular structure of S_0 as:

$$S_0 = \begin{pmatrix} S_1 & * & \cdots & & & \cdots & * \\ 0 & S_2 & * & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & \vdots \\ & & 0 & B & * & & \\ \vdots & & & 0 & T_1 & * & \vdots \\ \vdots & & & & 0 & T_2 & * \\ 0 & \cdots & & & \cdots & \ddots & \ddots \end{pmatrix} \tag{13}$$

where S_1, S_2, ... represent the diagonal subblocks associated to the subspaces of S and S_0 while T_1, T_2, ... represent the diagonal subblocks associated to the subspaces of S_0^T and B is the "bulk" part for the remaining network of 90965 nodes. The stars represent potential non-vanishing entries whose values do not influence the eigenvalues of S_0. The subspace blocks S_1, S_2, ... and T_1, T_2, ... which are individually of maximal dimension 10 can be directly diagonalized and it turns that out of 372382 eigenvalues in these blocks only about 4000 eigenvalues (counting degeneracies) or 950 eigenvalues (non-counting degeneracies) are different from zero. Most of these eigenvalues are not degenerate and are therefore not eigenvalues of S but there are still quite many degenerate eigenvalues at $\lambda = \pm 1/\sqrt{n}$ with $n \geq 2$ taking small integer values and who are also eigenvalues of S with a degeneracy reduced by one.

Concerning the bulk block B we can write it in the form $B = B_0 + f_1\, e_1^T$ where f_1 is the first column vector of B and $e_1^T = (1, 0, \ldots, 0)$. The matrix B_0 is obtained from B by replacing its first column to zero. We can apply the above argumentation between S and S_0 in the same way to B and B_0, i.e. the degenerate eigenvalues of B_0 with degeneracy m are also eigenvalues of B with degeneracy $m-1$ (with eigenvectors obeying $e_1^T\psi = 0$) and therefore eigenvalues of S with degeneracy $m-2$. The matrix B_0 is decomposed in a similar way as in (13) with subspace blocks, which can be diagonalized numerically, and a new bulk block $\tilde{B}$ of dimension 63559 and which may be treated in the same way by taking out its first column. This procedure provides a recursive scheme which after 9 iterations stops with a final bulk block of zero size. At each iteration we keep only subspace eigenvalues with degeneracies $m \geq 2$ and which are joined with reduced degeneracies $m-1$ to the subspace spectrum of the previous iteration. For this joined spectrum we keep again only eigenvalues with degeneracies $m \geq 2$ which are joined with the subspace spectrum of the next higher level etc.

In this way we have determined all eigenvalues of S_0 with a degeneracy $m \geq 2$ which belong to the eigenvalues of S of the first group. Including the direct subspace of S there are 4999 non-vanishing eigenvalues (counting degeneracies) or 442 non-vanishing eigenvalues (non-counting degeneracies). The degeneracy of the zero

λ	degeneracy
1	27
-1	18
$\pm 1/\sqrt{2}$	27
$\pm 1/\sqrt{3}$	20
$1/2$	58
$-1/2$	52
$\pm 1/\sqrt{5}$	20
$\pm 1/\sqrt{6}$	52
$\pm 1/\sqrt{7}$	6
$\pm 1/\sqrt{8}$	44
$1/3$	47
$-1/3$	39
$\pm 1/\sqrt{10}$	33
$\pm 1/\sqrt{11}$	1
$\pm 1/\sqrt{12}$	85
$\pm 1/\sqrt{14}$	15
$\pm 1/\sqrt{15}$	46
$1/4$	52
$-1/4$	42
$\pm 1/\sqrt{18}$	29
$\pm 1/\sqrt{20}$	60
$\pm 1/\sqrt{21}$	30
$\pm 1/\sqrt{22}$	3
$\pm 1/\sqrt{24}$	69
$1/5$	20
$-1/5$	11

TABLE I: Degeneracies of the eigenvalues with largest modulus for the whole CNPR whose eigenvectors ψ belong to the first group and obey the orthogonality $d^T\psi = 0$ with the dangling vector d.

eigenvalue (or the dimension of the generalized kernel) is found by this procedure to be 455789 but this would only be correct assuming that there are no general eigenvectors of higher order (representation vectors of non-trivial Jordan blocks) which is clearly not the case. The Jordan subspace structure of the zero eigenvalue complicates the argumentation. Here at each iteration step the degeneracy has to be reduced from m to $m-D$ where $D > 1$ is the dimension of the maximal Jordan block since each generalized eigenvector at a given order has to be treated as an independent vector when constructing vectors obeying the orthogonality with respect to the dangling vector d. Therefore the degeneracy of the zero eigenvalue cannot be determined exactly but we may estimate its degeneracy of about ~ 455000 out of 463348 nodes in total. This implies that the number of non-vanishing eigenvalues is about $\sim 8000 - 9000$ which is considerably larger than the value of 352 for the triangular CNPR but still much smaller than the total network size.

In Table I we provide the degeneracies for some of the eigenvalues $\pm 1/\sqrt{n}$ for integer n in the range $1 \le n \le 25$. The degeneracies for $+1/\sqrt{n}$ and $-1/\sqrt{n}$ are identical for non-square numbers n (with non-integer $\sqrt{n}$) and different for square numbers (with integer $\sqrt{n}$). Apparently for non-square numbers the eigenvalues are only generated from effective 2×2 blocks:

$$\begin{pmatrix} 0 & 1/n_1 \\ 1/n_2 & 0 \end{pmatrix} \quad \Rightarrow \quad \lambda = \pm \frac{1}{\sqrt{n_1\, n_2}} \tag{14}$$

with positive integers n_1 and n_2 such that $n = n_1\, n_2$ while for square numbers $n = m^2$ they may be generated by such blocks or by simple 1×1 blocks containing $1/m$ such that the degeneracy for $+1/\sqrt{n} = +1/m$ is larger than the degeneracy for $-1/\sqrt{n} = -1/m$. Furthermore, statistically the degeneracy is smaller for prime numbers n or numbers with less factorization possibilities and larger for numbers with more factorization possibilities. The Arnoldi method (with 52 bits for double-precision arithmetic and $n_A = 8000$) provides according to the sizes of the plateaux visible in Fig. 8 the overall approximate degeneracies ~ 60 for $|\lambda| = 1/\sqrt{2}$ (i.e. $\pm 1/\sqrt{2}$ counted together), ~ 50 for $|\lambda| = 1/\sqrt{3}$ and ~ 115 for $|\lambda| = 1/2$. These values are coherent with (but slightly larger than) the values 54, 40 and 110 taken from Table I. Actually, as we will see below, the slight differences between the degeneracies obtained from Fig. 8 and from Table I are indeed relevant and correspond to some eigenvalues of the second group which are close but not identical to $\pm 1/\sqrt{2}$, $\pm 1/\sqrt{3}$ or $\pm 1/2$ and do not contribute in Table I.

We now consider the eigenvalues λ of S for the eigenvectors of the second group with non-orthogonality $d^T\psi \neq 1$ or $d^T\psi = 1$ after proper renormalization of ψ. Now ψ cannot be an eigenvector of S_0 and λ is not an eigenvalue of S_0. As in the last subsection the eigenvalue equation $S\psi = \lambda\psi$, the condition $d^T\psi = 1$ and (4) imply that the eigenvalue λ of S is a zero of the rational function

$$\mathcal{R}(\lambda) = 1 - d^T \frac{\mathbb{1}}{\lambda\mathbb{1} - S_0} e/N = 1 - \sum_{j,q} \frac{C_{jq}}{(\lambda - \rho_j)^q} \tag{15}$$

where we have formally expanded the vector e/N in eigenvectors of S_0 and with ρ_j being the eigenvalues of S_0 and q is the *order* of the eigenvector of ρ_j used in this expansion, i.e. $q = 1$ for simple eigenvectors and $q > 1$ for generalized eigenvectors of higher order due to Jordan blocks. Note that even the largest possible value of q for a given eigenvalue may be (much) smaller than its multiplicity m. Furthermore the case of simple repeating eigenvalues (with simple eigenvectors) with higher multiplicity $m > 1$ leads only to several identical terms $\sim (\lambda - \rho_j)^{-1}$ for any eigenvector of this eigenvalue thus all contributing to the coefficients C_{jq} and whose precise values we do not need to know in the following. For us the important point is that the second identity in (15) establishes that $\mathcal{R}(\lambda)$ is indeed a rational function whose

denominator and numerator polynomials have the same degree and whose poles are (some of) the eigenvalues of S_0.

We mention that one can also show by a simple determinant calculation (similar to a calculation shown in [19] for triangular networks with nilpotent S_0) that:

$$P_S(\lambda) = P_{S_0}(\lambda)\,\mathcal{R}(\lambda) \tag{16}$$

where $P_S(\lambda)$ [or $P_{S_0}(\lambda)$] is the characteristic polynomial of S (S_0). Therefore those zeros of $\mathcal{R}(\lambda)$ which are not zeros of $P_{S_0}(\lambda)$ (i.e. not eigenvalues of S_0) are indeed zeros of $P_S(\lambda)$ (i.e. eigenvalues of S) since there are not poles of $\mathcal{R}(\lambda)$. Furthermore, generically the *simple* zeros $P_{S_0}(\lambda)$ also appear as poles in $\mathcal{R}(\lambda)$ and are therefore not eigenvalues of S. However, for a zero of $P_{S_0}(\lambda)$ (eigenvalue of S_0) with *higher multiplicity* $m > 1$ (and unless m is equal to the maximal Jordan block order q associated to this eigenvalue of S_0) the corresponding pole in $\mathcal{R}(\lambda)$ only reduces the multiplicity to $m-1$ (or $m-q$ in case of higher order generalized eigenvectors) and we have also a zero of $P_S(\lambda)$ (eigenvalue of S). Some of the eigenvalues of S_0, whose eigenvectors ψ are orthogonal to the dangling vector ($d^T\psi = 0$) and do not contribute in the expansion in (15), are not poles of $\mathcal{R}(\lambda)$ and therefore also eigenvalues of S. This concerns essentially the direct subspace eigenvalues of S which are also direct subspace eigenvalues of S_0 as already mentioned above. In total the identity (16) confirms exactly the above picture that there are two groups of eigenvalues and with the special role of direct subspace eigenvalues belonging to the first group.

Our aim is to determine numerically the zeros of the rational function $\mathcal{R}(\lambda)$. In order to evaluate this function we expand the first identity in (15) in a matrix geometric series and we obtain

$$\mathcal{R}(\lambda) = 1 - \sum_{j=0}^{\infty} c_j \lambda^{-1-j} \tag{17}$$

with the coefficients c_j defined in (5) and provided that this series converges. In the last subsection, where we discussed the case of a nilpotent matrix S_0 with $S_0^l = 0$, the series was finite and for this particular case we had $\mathcal{R}(\lambda) = \lambda^{-l}\mathcal{P}_r(\lambda)$ where $\mathcal{P}_r(\lambda)$ was the reduced polynomial defined in (8) and whose zeros provided the l nonvanishing eigenvalues of S for nilpotent S_0.

However, for the CNPR the series are infinite since all c_j are different from zero. One may first try a crude approximation and simply replace the series by a finite sum for $j < l$ and using some rather large cutoff value for l and determine the zeros in the same way as for the nilpotent case (high precision calculation of the zeros of the reduced polynomial of degree l). It turns that in this way we obtain correctly the largest core space eigenvalue of S as $\lambda_1 = 0.999751822283878$ which is also obtained by (any variant of) the Arnoldi method. However, the other zeros obtained by this approximation lie all on a circle of radius ≈ 0.9 in the complex plane and do not obviously represent any valid eigenvalues. Increasing the cutoff value l does not help either and it increases only the density of zeros on this circle. To understand this behavior we note that in the limit $j \to \infty$ the coefficients c_j behave as $c_j \propto \rho_1^j$ where $\rho_1 = 0.902448280519224$ is the largest eigenvalue of the matrix S_0 with an eigenvector non-orthogonal to d. Note that the matrix S_0 has also some degenerate eigenvalues at $+1$ and -1 but these eigenvalues are obtained from the direct subspace eigenvectors of S (which are also direct subspace eigenvectors of S_0) and which are orthogonal to the dangling vector d and do not contribute in the rational function (15). It turns actually out that the eigenvalue ρ_1 is also the largest *subspace space* eigenvalue of S_0 (after having removed the direct subspace nodes of S). By analyzing explicitly the small-dimensional subspace related to this eigenvalue one can show that ρ_1 is given as the largest solution of the polynomial equation $x^3 - \frac{2}{3}x - \frac{2}{15} = 0$ and can therefore be expressed as $\rho_1 = 2\,\mathrm{Re}\,[(9 + i\sqrt{119})^{1/3}]/(135)^{1/3}$. The asymptotic behavior $c_j \propto \rho_1^j$ is also confirmed by the direct numerical evaluation of c_j. Therefore the series (17) converges only for $|\lambda| > \rho_1$ and a simple (even very large) cutoff in the sum implies that only eigenvalues $|\lambda_j| > \rho_1$ can be determined as a zero of the finite sum. The only eigenvalue respecting this condition is the largest core space eigenvalue λ_1 given above.

One may try to improve this by a "better" approximation which consists of evaluating the sum exactly up to some value l and than to replace the remaining sum as a geometric series with the approximation: $c_j \approx c_l \rho_1^{j-l}$ for $j \geq l$ and with ρ_1 determined as the ratio $\rho_1 = c_l/c_{l-1}$ (which provides a sufficient approximation) or taken as its exact (high precision) value. This improved approximation results in $\mathcal{R}(\lambda) \approx \lambda^{-l}(\lambda - \rho_1)^{-1}\mathcal{P}(\lambda)$ with a polynomial $\mathcal{P}(\lambda)$ whose zeros provide in total four correct eigenvalues. Apart from λ_1 it also gives $\lambda_2 = 0.902445536212661$ (note that this eigenvalue of S is very close but different to the eigenvalue ρ_1 of S_0) and $\lambda_{3,4} = 0.765857950563684 \pm i\,0.251337495625571$ such that $|\lambda_{3,4}| = 0.806045245100386$. All these four core space eigenvalues coincide very well with the first four eigenvalues obtained from the Arnoldi method. However, the other zeros of the Polynomial $\mathcal{P}(\lambda)$ lie again on a circle, now with a reduced radius ≈ 0.7, and do not coincide with eigenvalues of S. This can be understood by the fact that the coefficients c_j obey for $j \to \infty$ the more precise asymptotic expression $c_j \approx C_1\rho_1^j + C_2\rho_2^j + C_2\rho_3^j + \ldots$ with the next eigenvalues $\rho_2 = 1/\sqrt{2} \approx 0.707$ and $\rho_3 = -\rho_2$. Here the first term $C_1\rho_1^j$ is dealt with analytically by the replacement of the geometric series but the other terms create a new convergence problem. Therefore the improved approximation allows only to determine the four core space eigenvalues with $|\lambda_j| > |\rho_{2,3}| = 1/\sqrt{2}$. To obtain more valid eigenvalues it seems to be necessary to sum up by geometric series many of the next terms, not only the next two terms due to ρ_2 and ρ_3, but also the

following terms of smaller eigenvalues ρ_j of S_0. In other words the exact pole structure of the rational function $\mathcal{R}(\lambda)$ has be kept as best as possible.

Therefore due to the rational structure of the function $\mathcal{R}(\lambda)$ with many eigenvalues ρ_j of S_0 that determine its precise pole structure we suggest the following numerical approach using high precision arithmetic. For a given number p of binary digits, e.g. $p = 1024$, we determine the coefficients c_j for $j < l$ where the cutoff value

$$l \approx \frac{\ln(1-\rho_1) - p\ln(2)}{\ln(\rho_1)} \approx 6.753\,p + \text{const.} \qquad (18)$$

is sufficiently large to evaluate the sum (17) accurately in the given precision of p binary digits (error below 2^{-p}) for *all complex values* λ *on the unit circle*, i.e. $|\lambda| = 1$, where the series converges well. Furthermore we choose a number n_R of "eigenvalues" we want to calculate, e.g. $n_R = 300$, and evaluate the rational function $\mathcal{R}(z)$ at $n_S = 2n_R + 1$ support points $z_j = \exp(2\pi i\, j/n_S)$ ($j = 0, \ldots, n_S - 1$) uniformly distributed on the unit circle using the series (17). Then we calculate the rational function $R_I(z)$ which interpolates $\mathcal{R}(z)$ at the n_S support points z_j, $R_I(z_j) = \mathcal{R}(z_j)$, using Thiele's interpolation formula. Then the numerator and denominator polynomials of $R_I(z)$ are both of degree n_R. Thiele's interpolation formula expresses $R_I(z)$ in terms of a continued fraction expansion using inverse differences. This method is quite standard and well described in the literature of numerical mathematics, see for example [31]. After having evaluated a table of n_S inverse differences (with $n_S^2/2$ operations) one can evaluate arbitrary values of $R_I(z)$ using the continued fraction expansion (with n_S operations). It is not very difficult to derive from the continued fraction expansion a recursive scheme to evaluate the values of the numerator and denominator polynomials separately as well as their derivatives. Using this scheme we determine the n_R complex zeros of the numerator polynomial using the (high precision variant of the) Newton-Maehly method. These zeros correspond to the zeros of the rational functional $\mathcal{R}(z)$ and are taken as approximate eigenvalues of the matrix S of the second group. The main idea of this approach is to evaluate these zeros from the analytical continuation of $\mathcal{R}(z)$ using values for $|z| = 1$ to determine its zeros well inside the unit circle.

We also consider a second variant of the method where the number of support points $n_S = 2n_R + 2$ is even (instead of $n_S = 2n_R + 1$ being odd as for the first variant). In this case the numerator polynomial is of degree $n_R + 1$ (instead of n_R) while the denominator polynomial is of degree n_R and we choose to interpolate the inverse of the rational function $1/\mathcal{R}(z)$ (instead of $\mathcal{R}(z)$ itself) by $R_I(z)$ such that the zeros of $\mathcal{R}(z)$ are given by the n_R zeros of the denominator (instead of the numerator) polynomial of $R_I(z)$.

The number n_R must not be too small in order to well approximate the second identity in (15) by the fit function. On the other hand for a given precision of p binary digits the number of n_R must not be too large as well because the coefficients c_j, which may be written as the expansion $c_j = \sum_\nu C_\nu\, \rho_\nu^j$, do not contain enough information to resolve its structure for the smaller eigenvalues ρ_j of S_0. Therefore for too large values of n_R (for a given precision), we obtain additional artificial zeros of the numerator polynomial (or of the denominator polynomial for the second variant) of $R_I(z)$, mostly close to the unit circle, somehow as additional nodes around the support points.

It turns out that for the proper combination of p and n_R values the method provides highly accurate eigenvalues and works astonishingly well. In particular for values of n_R below a certain threshold (depending on the precision p) both variants of the method with odd or even number of support points provide numerically identical zeros (with final results rounded to 52 binary digits) which indeed coincide very accurately (for most of them) with the eigenvalues of S we want to determine.

For example, as can be seen in Fig. 9, for $p = 1024$ we obtain $n_R = 300$ eigenvalues for which the big majority coincides numerically (error $\sim 10^{-14}$) with the eigenvalues obtained from the high precision Arnoldi method for 768 binary digits and furthermore both variants of the rational interpolation method provide identical spectra.

However for $n_R = 340$ some of the zeros do not coincide with eigenvalues of S and most of these deviating zeros lie close to the unit circle. We can even somehow distinguish between "good" zeros (associated to eigenvalues of S) being identical for both variants of the method and "bad" artificial zeros which are completely different for both variants (see Fig. 9). We note that for the case of too large n_R values the artificial zeros are extremely sensitive to numerical round-off errors (in the high precision variables) and that they change strongly, when slightly modifying the support points (e.g. a random modification $\sim 10^{-18}$ or simply changing their order in the interpolation scheme) or when changing the precise numerical algorithm (e.g. between direct sum or Horner scheme for the evaluation of the series of the rational function). Furthermore, they do not respect the symmetry that the zeros should come in pairs of complex conjugate numbers in case of complex zeros. This is because Thiele's rational interpolation scheme breaks the symmetry due to complex conjugation once round-off errors become relevant.

However, we have carefully verified that for the proper values of n_R not being too large (e.g. $n_R = 300$ for $p = 1024$) the obtained zeros are numerically identical (with 52 binary digits in the final result) with respect to small changes of the support points (or their order) or with respect to different numerical algorithms and that they respect perfectly the symmetry due to complex conjugation.

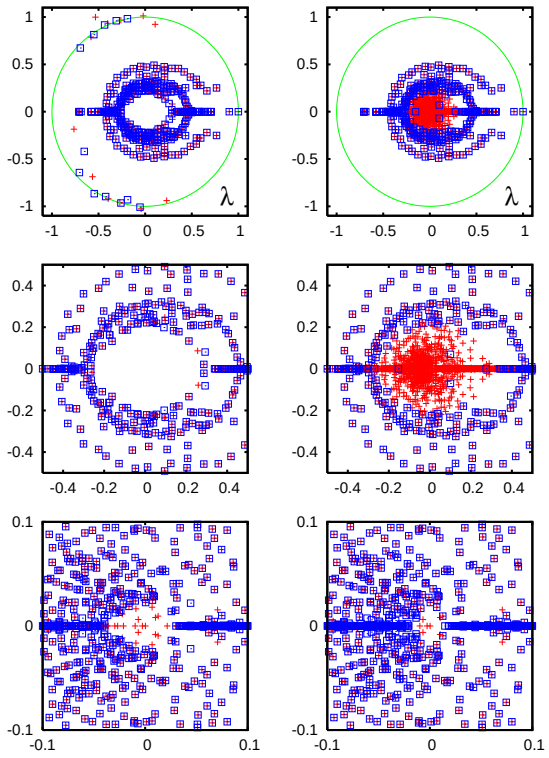


FIG. 9: (Color online) *Top Panels: Left:* Comparison of $n_R = 340$ core space eigenvalues of S for CNPR obtained by two variants of the rational interpolation method (see text) with the numerical precision of $p = 1024$ binary digits, 681 support points (first variant, red crosses) or 682 support points (second variant, blue squares). *Right:* Comparison of the core space eigenvalues of CNPR obtained by the high precision Arnoldi method with $n_A = 2000$ and $p = 768$ binary digits (red crosses, same data as blue squares in Fig. 7) with the eigenvalues obtained by (both variants of) the rational interpolation method with the numerical precision of $p = 1024$ binary digits and $n_R = 300$ eigenvalues (blue squares). Here both variants with 601 or 602 support points provide identical spectra (differences below 10^{-14}). *Middle panels:* Same as top panels with a zoomed range: $-0.5 \le \mathrm{Re}(\lambda), \mathrm{Im}(\lambda) \le 0.5$. *Bottom Panels: Left:* Comparison of the core space spectra obtained by the high precision Arnoldi method (red crosses, $n_A = 2000$ and $p = 768$) and by the rational interpolation method with $p = 12288$, $n_R = 2000$ eigenvalues (blue squares). *Right:* Same as left panel with $p = 16384$, $n_R = 2500$ for the rational interpolation method. Both panels are shown in a zoomed range: $-0.1 \le \mathrm{Re}(\lambda), \mathrm{Im}(\lambda) \le 0.1$. Eigenvalues outside the shown range coincide up to graphical precision and both variants of the rational interpolation method provide numerically identical spectra.

This method, despite the necessity of high precision calculations, is not very expensive, especially for the memory usage, compared, for example, with the high precision Arnoldi method. Furthermore, its efficiency for the computation time can be improved by the trick of summing up the largest terms in the series (17) as a geometrical series which allows to reduce the cutoff value of l by a good factor 3, i.e. replacing $\rho_1 \approx 0.902$ by $\rho_2 = 1/\sqrt{2} \approx 0.707$ in the estimate (18) of l which gives $l \approx 2\,p + \mathrm{const}$. We have increased the number of binary digits up to $p = 16384$ and we find that for $p = 1024, 2048, 4096, 6144, 8192, 12288, 16384$ we may use $n_R = 300, 500, 900, 1200, 1500, 2000, 2500$ and still avoid the appearance of artificial zeros. In Fig. 9 we also compare the result of the highest precisions $p = 12288$ (and $p = 16384$) using $n_R = 2000$ ($n_R = 2500$) with the high precision Arnoldi method with $n_A = 2000$ and $p = 768$ and these spectra coincide well apart from a minor number of smallest eigenvalues. In general, the complex isolated eigenvalues converge very well (with increasing values of p and n_R) while the strongly clustered eigenvalues on the real axis have more difficulties to converge. Comparing the results between $n_R = 2000$ and $n_R = 2500$ we see that the complex eigenvalues coincide on graphical precision for $|\lambda| \ge 0.04$ and the real eigenvalues for $|\lambda| \ge 0.1$. The Arnoldi method has even more difficulties on the real axis (convergence roughly for $|\lambda| \ge 0.15$) since it has implicitly to take care of the highly degenerate eigenvalues of the first group and for which it has difficulties to correctly find the degeneracies (see also Fig. 8).

Fig. 10 shows as summary the highest precision spectra of S with core space eigenvalues obtained by the Arnoldi method or the rational interpolation method (both at best parameter choices) and also taking into account the direct subspace eigenvalues of S and the above determined eigenvalues of the first group (degenerate subspace eigenvalues of S_0).

We remind that the rational interpolation method allows only to determine the eigenvalues of S of the second group, i.e. the eigenvalues which are not eigenvalues of S_0 and whose eigenvectors obey $d^T\psi \neq 0$. The eigenvalues of the first group (with $d^T\psi = 0$) have to be determined separately by the above described scheme of degenerate subspace eigenvalues of S_0. In particular the eigenvalues given in Table I and belonging to the first group are not zeros of the rational function $\mathcal{R}(z)$ (they are actually poles of this function) but it turns out that there are some zeros of $\mathcal{R}(z)$ which are very close but not identical to some of the values in Table I. For example the rational interpolation method provides the following zeros: $1/2 + 3.13401098 \times 10^{-5}$, $1/2 + 1.3279300 \times 10^{-7}$, $1/\sqrt{2} - 1.1597 \times 10^{-10}$ or $1/\sqrt{2} - 6.419004 \times 10^{-8}$ which are indeed accurate in the given precision since they are stable for all values of $p \ge 1024$ and the corresponding maximal value of n_R and we have stopped the Newton iteration when the error of a zero was clearly below 10^{-18}. These zeros are also found with the same precision in the data of the high precision Arnoldi method for the three different values of 256, 512 or 768 binary digits. However, based only on results of the Arnoldi method it is not really clear if the small corrections to $1/2$ or $1/\sqrt{2}$ are real and exact or numerically artificial since the Arnoldi method has indeed problems with degenerate and clustered eigenvalues [17]. Therefore the rational interpolation method provides an independent and strong confir-

mation of the accuracy of these type of eigenvalues. We attribute their existence to a quasi-subspace structure, similarly as discussed in [10], with a matrix subblock as in (14) but which is still very weakly coupled (by many indirect network links) to the core space.

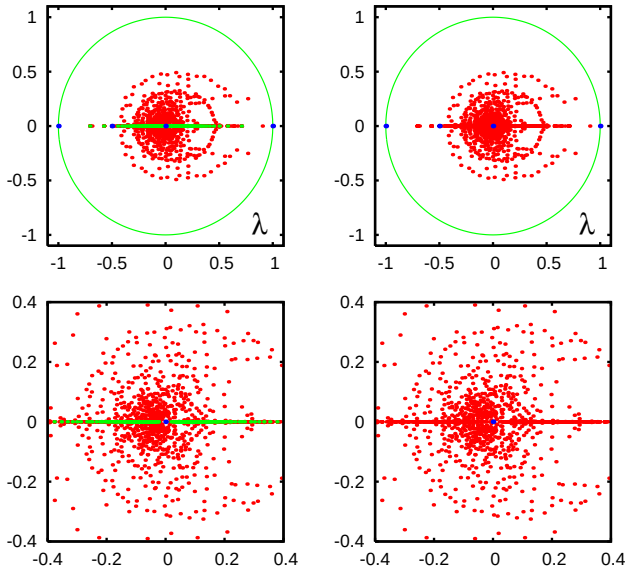


FIG. 10: (Color online) The most accurate spectrum of eigenvalues of S for CNPR. *Top panels: Left:* red dots represent the core space eigenvalues obtained by the rational interpolation method with the numerical precision of $p = 16384$ binary digits, $n_R = 2500$ eigenvalues. Green dots show the degenerate subspace eigenvalues of the matrix S_0 which are also eigenvalues of S with a degeneracy reduced by one (eigenvalues of the first group, see text). Blue dots show the direct subspace eigenvalues of S (same as blue dots in left upper panel in Fig. 3). *Right:* red dots represent the core space eigenvalues obtained by the high precision Arnoldi method with $n_A = 2000$ and the numerical precision of $p = 768$ binary digits and blue dots show the direct subspace eigenvalues of S. Note that the Arnoldi method determines implicitly also the degenerate subspace eigenvalues of S_0 which are therefore not shown in another color. *Bottom panels*: Same as top panels with a zoomed range: $-0.4 \leq \mathrm{Re}(\lambda), \mathrm{Im}(\lambda) \leq 0.4$.

III. FRACTAL WEYL LAW FOR CNPR

The concept of the fractal Weyl law [32, 33],[34] states that the number of states N_λ in a ring of complex eigenvalues with $\lambda_c \leq |\lambda| \leq 1$ scales in a polynomial way with the growth of matrix size:

$$N_\lambda = a N^b \ . \tag{19}$$

where the exponent b is related to the fractal dimension of underlying invariant set $d_f = 2b$. The fractal Weyl law was first discussed for the problems of quantum chaotic scattering in the semiclassical limit [32, 33],[34]. Later it was shown that this law also works for the Ulam matrix approximant of the Perron-Frobenius operators of dissipative chaotic systems with strange attractors [6, 7]. In [11] it was established that the time growing Linux Kernel network is also characterized by the fractal Weyl law with the fractal dimension $d_f \approx 1.3$.

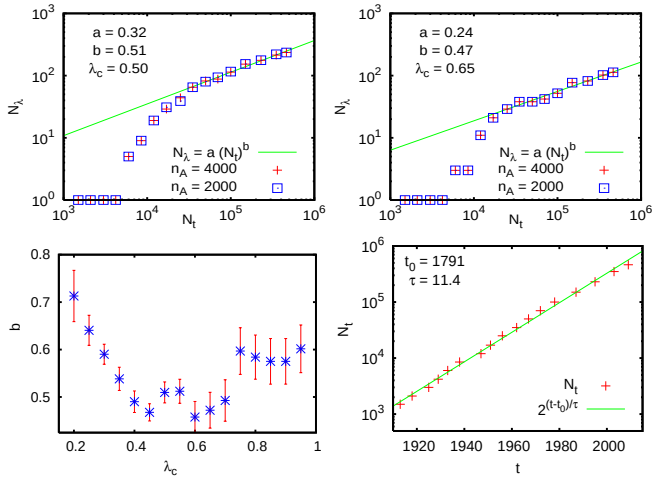


FIG. 11: (Color online) Data for the whole CNPR at different moments of time. *Top panels:* the left (right) panel shows the number N_λ of eigenvalues with $\lambda_c \leq \lambda \leq 1$ for $\lambda_c = 0.50$ ($\lambda_c = 0.65$) versus the effective network size N_t where the nodes with publication times after a cut time t are removed from the network. The green line shows the Weyl law $N_\lambda = a\,(N_t)^b$ with parameters $a = 0.32 \pm 0.08$ ($a = 0.24 \pm 0.11$) and $b = 0.51 \pm 0.02$ ($b = 0.47 \pm 0.04$) obtained from a fit in the range $3 \times 10^4 \leq N_t < 5 \times 10^5$. The number N_λ includes both exactly determined invariant subspace eigenvalues and core space eigenvalues obtained from the Arnoldi method with double-precision (52 binary digits) for $n_A = 4000$ (red crosses) and $n_A = 2000$ (blue squares). *Bottom panels: Left:* exponent b with error bars obtained from the fit $N_\lambda = a\,(N_t)^b$ in the range $3 \times 10^4 \leq N_t < 5 \times 10^5$ versus cut value λ_c. *Right:* effective network size N_t versus cut time t (in years). The green line shows the exponential fit $2^{(t-t_0)/\tau}$ with $t_0 = 1791 \pm 3$ and $\tau = 11.4 \pm 0.2$ representing the number of years after which the size of the network (number of papers published in all Physical Review journals) is effectively doubled.

The fact that $b < 1$ implies that the majority of eigenvalues drop to zero. We see that this property also appears for the CNPR if we test here the validity of the fractal Weyl law by considering a time reduced CNPR of size N_t including the N_t papers published until the time t (measured in years) for different times t in order to obtain a scaling behavior of N_λ as a function of N_t. The data presented in Fig. 11 shows that the network size grows approximately exponentially as $N_t = 2^{(t-t_0)/\tau}$ with the fit parameters $t_0 = 1791$, $\tau = 11.4$. The time interval considered in Fig. 11 is $1913 \leq t \leq 2009$ since the first data point corresponds to $t = 1913$ with $N_t = 1500$ papers published between 1893 and 1913. The results for N_λ show that its growth is well described by the relation $N_\lambda = a\,(N_t)^b$ for the range when the number of articles

becomes sufficiently large $3 \times 10^4 \leq N_t < 5 \times 10^5$. This range is not very large and probably due to that there is a certain dependence of the exponent b on the range parameter λ_c. However, we have $0.47 < b < 0.6$ for all $\lambda_c \geq 0.4$ that is definitely smaller than unity and thus the fractal Weyl law is well applicable to the CNPR. The value of b increases up to 0.7 for the data points with $\lambda_c < 0.4$ but this is due to the fact here N_λ also includes some numerically incorrect eigenvalues related to the numerical instability of the Arnoldi method at standard double-precision (52 binary digits) as discussed in the beginning of the previous section.

We think that the most appropriate choice for the description of the data is obtained at $\lambda_c = 0.4$ which from one side excludes small, partly numerically incorrect, values of λ and on the other side gives sufficiently large values of N_λ. Here we have $b = 0.49 \pm 02$ corresponding to the fractal dimension $d = 0.98 \pm 0.04$. Furthermore, for $0.4 \leq \lambda_c \leq 0.7$ we have a rather constant value $b \approx 0.5$ with $d_f \approx 1.0$. Of course, it would be interesting to extend this analysis to a larger size N of CNPR but for that we still should wait about 10 years until the network size will be doubled comparing to the size studied here.

IV. PROPERTIES OF EIGENVECTORS

The results for the eigenvalue spectra of CNPR presented in the previous sections show that most of the visible eigenvalues on the real axis (except for the largest one) in Figs. 9 and 10 are due to the effect of future citations. They appear either directly due to 2×2 subblocks of the type (14) with a cycle where two papers mutually cite each other giving the degenerate eigenvalues of the first group, or indirectly by eigenvalues of the second group which are also numerous on the real axis. On the other hand, as can be seen in Fig. 6, for the triangular CNPR, where all future citations are removed, there is only the leading eigenvalue $\lambda = 1$ and a small number of negative eigenvalues with $-0.27 < \lambda < 0$ on the real axis. All other eigenvalues are complex and a considerable number of the largest ones are relatively close to corresponding complex eigenvalues for the whole CNPR with future citations.

The appearance of future citations is quite specific and is not a typical situation for citation networks. Therefore we consider the eigenvectors of complex eigenvalues for the triangular CNPR which indeed represent the typical physical situation without future citations. There is no problem to evaluate these eigenvectors by the Arnoldi method, either with double-precision, provided the eigenvalue of the eigenvector is situated in the region of numerically accurate eigenvalues, or with the high precision variant of the Arnoldi method. However, for the triangular CNPR we have, according to the semi-analytical theory presented above, the explicit formula:

$$\psi \propto (\lambda \mathbb{1} - S_0)^{-1}\, e/N = \sum_{j=0}^{l-1} \lambda^{-(1+j)}\, S_0^j e/N \qquad (20)$$

where the normalization is given by $\sum_i |\psi(i)| = 1$. This expression is quite convenient and we verified that it provides the same eigenvectors (up to numerical errors) as the Arnoldi method.

In Fig. 12 we show two eigenvectors of S: one ψ_0 for the leading eigenvalue $\lambda_0 = 1$ and another ψ_{39} for a complex eigenvalue at $|\lambda_{39}| < 1$. The eigenvector of λ_0 gives the PageRank probability for the triangular CNPR (at $\alpha = 1$). We also consider the eigenvector for the complex eigenvalue $\lambda_{39} = -0.3738799 + i\,0.2623941$ (eigenvalues are ordered by their absolute values starting from $\lambda_0 = 1$). In this figure the modulus of $|\psi_j(N_t)|$ is shown versus the time index N_t as introduced in Fig. 11. We also indicate the positions of five famous papers: BCS 1957 [35] at $K = 6$, Anderson 1958 [36] $K = 63$, Benettin et al. 1976 [37] $K = 441$, Thouless 1977 [38] $K = 256$ and Abrahams et al. 1979 [39] $K = 74$. In the first eigenvector for $\lambda_0 = 1$ all of these papers have quite dominating positions, especially BCS 1957 and Abrahams et al. 1979 which are the most important ones if compared to papers of comparable publication date. Only considerably older papers have higher positions in this vector.

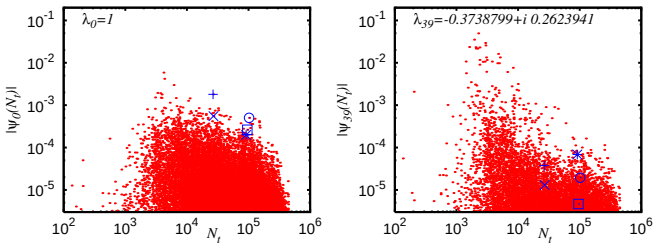


FIG. 12: (Color online) Two eigenvectors of the matrix S for the triangular CNPR. Both panels show the modulus of the eigenvector components $|\psi_j(N_t)|$ versus the time index N_t (as used in Fig. 11) with nodes/articles ordered by the publication time (small red dots). The blue points represent five particular articles: BCS 1957 (+), Anderson 1958 (×), Benettin et al. 1976 (∗), Thouless 1977 (⊡) and Abrahams et al. 1979 (⊙). The left (right) panel corresponds to the real (complex) eigenvalue $\lambda_0 = 1$ ($\lambda_{39} = -0.3738799 + i\,0.2623941$).

For the second eigenvector with complex eigenvalue the older papers (with $10^3 < N_t < 10^4$ corresponding to publications times between 1910 and 1940) are strongly enhanced in its importance while the above five famous papers lose their importance. The top 3 positions of largest amplitude $|\psi_{39}(i)|$ correspond to DOI 10.1103/PhysRev.14.409 (1919), 10.1103/PhysRev.8.561 (1916), 10.1103/PhysRev.24.97 (1917). These old articles study the radiating potentials of nitrogen, ionization impact in gases and the abnormal low voltage arc. It is clear that this eigenvector selects a certain community of

old articles related to a certain ancient field of interest. This fact is in agreement with the studies of eigenvectors of Wikipedia network [13] showing that the eigenvectors with $0 < |\lambda| < 1$ select specific communities.

It is interesting to note that the top node of the vector ψ_0 appears in the position $K_{39} = 39$ in local rank index of the vector ψ_{39} (ranking in decreasing order by modulus of $|\psi(i)|$). On the other side the top node of ψ_{39} appears at position $K_0 = 30$ of vector ψ_0. This illustrates how different nodes contribute to different eigenvectors of S.

It is useful to characterize the eigenvectors by their Inverse Participation Ratio (IPR) $\xi_i = (\sum_j |\psi_i(j)|^2)^2 / \sum_j |\psi_i(j)|^4$ which gives an effective number of nodes populated by an eigenvector ψ_i (see e.g. [8, 13]). For the above two vectors we find $\xi_0 = 20.67$ and $\xi_{39} = 10.76$. This means that ξ_{39} is mainly located on approximately 11 nodes. For ξ_0 this number is twice larger in agreement with data of Fig. 12 which show a clearly broader distribution comparing to ξ_{39}.

We also considered a few tens of eigenstates of S of the whole CNPR. They are mainly located on the complex plane around the largest oval curve well visible in the spectrum (see Fig. 10 top right panel). The IPR value of these eigenstates with $|\lambda| \sim 0.4$ varies in the range $4 < \xi < 13$ showing that they are located on some effective quasi-isolated communities of articles. About 10 of them are related to the top article of ψ_{39} shown in Fig. 12 meaning that these ten vectors represent various linear combinations of vectors on practically the same community. In global, we can say that the eigenstates of G are well localized since $\xi \ll N$. A similar situation was seen for the Wikipedia network [13].

Of course, in addition to ξ it is also useful to consider the whole distribution of ψ amplitudes over the nodes. Such a consideration has been done for the Wikipedia network in [13]. For the CNPR we leave such detailed studies for further investigations.

V. CHEIRANK VERSUS PAGERANK FOR CNPR

The dependence of PageRank probability $P(K)$ on PageRank index K is shown in Fig. 13. The results are similar to those of [22]. We note that the PageRank of the triangular CNPR has the same top 9 articles as for the whole CNPR (both at $\alpha = 0.85$ and with a slight interchanged order of positions 7, 8, 9). This confirms that the future citations produce only a small effect on the global ranking.

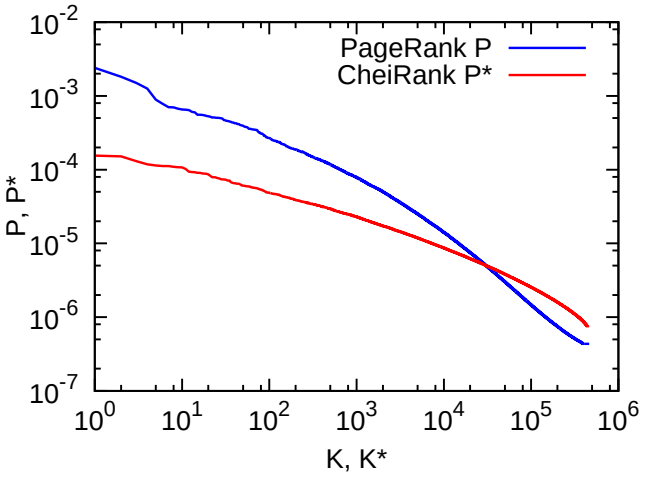


FIG. 13: (Color online) Dependence of probability of PageRank P (CheiRank P^*) on corresponding index K (K^*) for the CNPR at $\alpha = 0.85$.

Following previous studies [24],[25, 26], in addition to the Google matrix G we also construct the matrix G^* following the same definition (1) but for the network with inverted direction of links. The PageRank vector of this matrix G^* is called the CheiRank vector with probability $P^*(K_i^*)$ and CheiRank index K^*. The dependence of $P^*(K_i^*)$ is shown in Fig. 13. We find that the IPR values of P and P^* are $\xi = 59.54$ and 1466.7 respectively. Thus P^* is extended over significantly larger number of nodes comparing to P. A power law fit of the decay $P \propto 1/K^\beta$, $P^* \propto 1/K^{*\beta}$, done for a range $K, K^* \leq 2 \times 10^5$ gives $\beta \approx 0.57$ for P and $\beta \approx 0.4$ for P^*. However, this is only an approximate description since there is a visible curvature (in a double logarithmic representation) in these distributions. The corresponding frequency distributions of ingoing links have exponents $\mu = 2.87$ while the distribution of outgoing links has $\mu \approx 3.7$ for outdegree $k \geq 20$, even if the whole frequency dependence in this case is rather curved and a power law fit is rather approximate in this case. Thus the usual relation $\beta = 1/(\mu - 1)$ [4, 8, 25] approximately works.

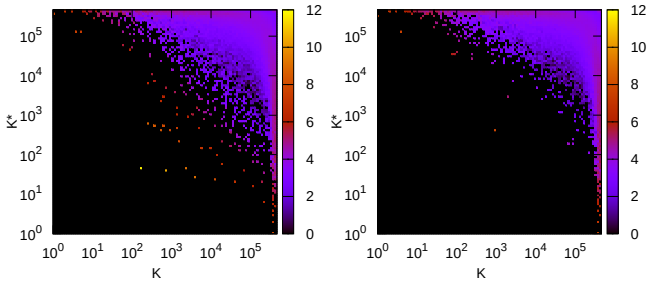


FIG. 14: (Color online) Density distribution $W(K,K^*) = dN_i/dKdK^*$ of Physical Review articles in the PageRank-CheiRank plane (K,K^*). Color bars show the natural logarithm of density, changing from minimal nonzero density (dark) to maximal one (white), zero density is shown by black. Left panel: all articles of CNPR; right panel: CNPR without Rev. Mod. Phys.

The correlation between PageRank and CheiRank vectors can be characterized by the correlator $\kappa = N\sum_{i=1}^{N} P(i)P^*(i) - 1$ [24, 26]. Here we find $\kappa = -0.2789$ for all CNPR, and $\kappa = -0.3187$ for CNPR without Rev. Mod. Phys. This is the most strong negative value of κ among all directed networks studied previously [26]. In a certain sense the situation is somewhat similar to the Linux Kernel network where $\kappa \approx 0$ or slightly negative ($\kappa > -0.1$ [24]). For CNPR, we can say that due to a almost triangular structure of G and G^* there is a very little overlap of top ranking in K and K^* that leads to a negative correlator value, since the components $P(i)P^*(i)$ of the sum for κ are small.

Each article i has two indexes K_i, K_i^* so that it is convenient to see their distribution on 2D PageRank-CheiRank plane. The density distribution $W(K,K^*) = dN_i/dKdK^*$ is shown in Fig. 14. It is obtained from 100×100 cells equidistant in log-scale (see details in [25, 26]). For the CNPR the density is homogeneous along lines $K = -K^* + const$ that corresponds to the absence of correlations between P and P^* [25, 26]. For the CNPR without Rev. Mod. Phys. we have an additional suppression of density at low K^* values. Indeed, Rev. Mod. Phys. contains mainly review articles with a large number of citations that place them on top of CheiRank. At the top 3 positions of K^* of CNPR we have DOI 10.1103/PhysRevA.79.062512, 10.1103/PhysRevA.79.062511, 10.1103/RevModPhys.81.1551 of 2009. These are articles with long citation lists on K shell diagram 4d transition elements; hypersatellites of 3d transition metals; superconducting phases of f electron compounds. For CNPR without Rev. Mod. Phys. the first two articles are the same and the third one has DOI 10.1103/PhysRevB.80.224501 being about model for the coexistence of d wave superconducting and charge density wave order in in high temperature cuprate superconductors. We see that the most recent articles with long citation lists are dominating.

The top PageRank articles are analyzed in detail in [22] and we do not discuss them here.

It is also useful to consider two-dimensional rank 2DRank K_2 defined by counting nodes in order of their appearance on ribs of squares in (K,K^*) plane with the square size growing from $K = 1$ to $K = N$ [25]. It selects highly cited articles with a relatively long citation list. For CNPR, we have top 3 such articles with DOI 10.1103/RevModPhys.54.437 (1982), 10.1103/RevModPhys.65.851 (1993), 10.1103/RevModPhys.58.801 (1986). Their topics are electronic properties of two dimensional systems, pattern formation outside of equilibrium, spin glasses facts and concepts. The 1st one located at $K = 183$, $K^* = 49$ is well visible in the left panel of Fig. 14. For CNPR without Rev. Mod. Phys. we find at $K_2 = 1$ the article with DOI 10.1103/PhysRevD.54.1 (1996) entitled *Review of Particle Physics* with a lot of information on physical constants.

For the ranking of articles about persons in Wikipedia networks [14, 25],[40], PageRank, 2DRank, CheiRank highlights in a different manner various sides of human activity. For the CNPR, these 3 ranks also select different types of articles, however, due a triangular structure of G, G^* and absence of correlations between PageRank and CheiRank vectors the useful side of 2DRank and CheiRank remains less evident.

VI. IMPACTRANK FOR INFLUENCE PROPAGATION

It is interesting to quantify how an influence of a given article propagates through the whole CNPR. To analyze this property we consider the following propagator acting on an initial vector v_0 located on a given article:

$$v_f = \frac{1-\gamma}{1-\gamma G}\, v_0 \quad , \quad v_f^* = \frac{1-\gamma}{1-\gamma G^*}\, v_0 \; . \qquad (21)$$

Here G, G^* are the Google matrices defined above, γ is a new impact damping factor being in a range $\gamma \sim 0.5-0.9$, v_f in the final vector generated by the propagator (21). This vector is normalized to unity $\sum_i v_f(i) = 1$ and one can easily show that it is equal to the PageRank vector of a modified Google matrix given by

$$\tilde{G} = \gamma\, G + (1-\gamma)\, v_0\, e^T \qquad (22)$$

where e is the vector with unit elements. This modified Google matrix corresponds to a stochastic process where at a certain time a given probability distribution is propagated with probability γ using the initial Google matrix G and with probability $(1-\gamma)$ the probability distribution is reinitialized with the vector v_0. Then v_f is the stationary vector from this stochastic process. Since the initial Google matrix G has a similar form, $G = \alpha S + (1-\alpha) e\, e^T/N$ with the damping factor α, the modified Google matrix can also be written as:

$$\tilde{G} = \tilde{\alpha}\, S + (1-\tilde{\alpha})\, v_p\, e^T \quad , \quad \tilde{\alpha} = \gamma\alpha \quad , \qquad (23)$$

with the personalization vector [4]

$$v_p = \frac{\gamma(1-\alpha)e/N + (1-\gamma)v_0}{1-\gamma\alpha} \qquad (24)$$

which is also sum normalized: $\sum_i v_p(i) = 1$. Obviously similar relations hold for G^* and v_f^*.

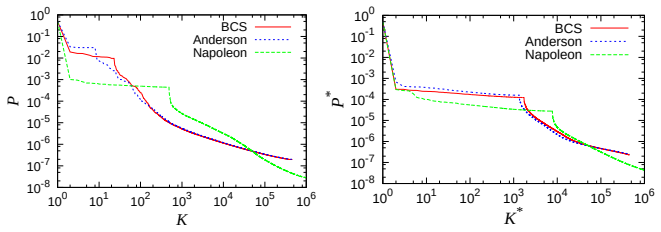


FIG. 15: (Color online) Dependence of impact vector v_f probability P and P^* (left and right panels) on the corresponding ImpactRank index K and K^* for an initial article v_0 as BCS [35] and Anderson [36] in CNPR, and Napoleon in English Wikipedia network from [40]. Here the impact damping factor is $\gamma = 0.5$.

The relation (21) can be viewed as a Green function with damping γ. Since $\gamma < 1$ the expansion in a geometric series is convergent and v_f can be obtained from about 200 terms of the expansion for $\gamma \sim 0.5$. The stability of v_f is verified by changing the number of terms. The obtained vectors v_f, v_f^* can be considered as effective PageRank, CheiRank probabilities P, P^* and all nodes can be ordered in the corresponding rank index K, K^*, which we will call ImpactRank.

The results for 2 initial vectors located on BCS [35] and Anderson [36] articles are shown in Fig. 15. In addition we show the same probability for the Wikipedia article *Napoleon* for the English Wikipedia network analyzed in [40]. The direct analysis of the distributions shows that the original article is located at the top position, the next step like structure corresponds to the articles reached by first outgoing (ingoing) links from v_0 for G (G^*). The next visible step correspond to a second link step.

Top ten articles for these 3 vectors are shown in Tables II, III, IV, V, VI. The analysis of these top articles confirms that they are closely linked with the initial article and thus the ImpactRank gives relatively good ranking results. At the same time, some questions for such ImpactRanking still remain to be clarified. For example, in Table V we find at the third position the well known Rev. Mod. Phys. on Anderson transitions but the paper of Abrahams *et al.* [39] appears only on far positions $K^* \approx 300$. The situation is changed if we consider all CNPR links as bi-directional obtaining a non-directional network. Then the paper [39] appears on the second position directly after initial article [36]. We think that such a problem appears due to triangular structure of CNPR where there is no intersection of forward and backward flows. Indeed, for the case of Napoleon we do not see such difficulties. Thus we hope that such an approach can be applied to other directed networks.

VII. MODELS OF RANDOM PERRON-FROBENIUS MATRICES

In this section we discuss the spectral properties of several random matrix models of Perron-Frobenius operators characterized by non-negative matrix elements and column sums normalized to unity. We call these models Random Perron-Frobenius Matrices (RPFM). To construct these models for a given matrix G of dimension N we draw N^2 independent matrix elements $G_{ij} \geq 0$ from a given distribution $p(G)$ (with $p(G) = 0$ for $G < 0$) with average $\langle G \rangle = 1/N$ and finite variance $\sigma^2 = \langle G^2 \rangle - \langle G \rangle^2$. A matrix obtained in this way obeys the column sum normalization only in average but not exactly for an arbitrary realization. Therefore we renormalize all columns to unity after having drawn the matrix elements. This renormalization provides some (hopefully small) correlations between the different matrix elements.

Neglecting these correlations for sufficiently large N the statistical average of the RPFM is simply given by $\langle G_{ij} \rangle = 1/N$ which is a projector matrix with the eigenvalue $\lambda = 1$ of multiplicity 1 and the corresponding eigenvector being the uniform vector e (with $e_i = 1$ for all i). The other eigenvalue $\lambda = 0$ is highly degenerate of multiplicity $N-1$ and its eigenspace contains all vectors orthogonal to the uniform vector e. Writing the matrix elements of a RPFM as $G_{ij} = \langle G_{ij} \rangle + \delta G_{ij}$ we may consider the fluctuating part δG_{ij} as a perturbation which only weakly modifies the unperturbed eigenvector e for $\lambda = 1$ but for the eigenvalue $\lambda = 0$ we have to apply degenerate perturbation theory which requires the diagonalization of δG_{ij}. According to the theory of non-symmetric real random Gaussian matrices [5, 41, 42] it is well established that the complex eigenvalue density of such a matrix is uniform on a circle of radius $R = \sqrt{N}\sigma$ with σ^2 being the variance of the matrix elements. One can also expect that this holds for more general, non-Gaussian, distributions with finite variance provided that we exclude extreme long tail distribution where the typical values are much smaller than σ. Therefore we expect that the eigenvalue density of a RPFM is determined by a single parameter being the variance σ^2 of the matrix elements resulting in a uniform density on a circle of radius $R = \sqrt{N}\sigma$ around $\lambda = 0$, in addition to the unit eigenvalue $\lambda = 1$ which is always an exact eigenvalue due to sum normalization of columns.

We now consider different variants of RPFM. The first variant is a full matrix with each element uniformly distributed in the interval $[0, 2/N[$ which gives the variance $\sigma^2 = 1/(3N^2)$ and the spectral radius $R = 1/\sqrt{3N}$. The second variant is a sparse RPFM matrix with Q nonvanishing elements per column and which are uniformly distributed in the interval $[0, 2/Q[$. Then the probability distribution is given by $p(G) = (1 - Q/N)\delta(G) +$

TABLE II: Spreading of impact on "Theory of superconductivity" paper by "J. Bardeen, L. N. Cooper and J. R. Schrieffer (doi:10.1103/PhysRev.108.1175) by Google matrix G with $\alpha = 0.85$ and $\gamma = 0.5$

ImpactRank	DOI	Title of paper
1	10.1103/PhysRev.108.1175	Theory of superconductivity
2	10.1103/PhysRev.78.477	Isotope effect in the superconductivity of mercury
3	10.1103/PhysRev.100.1215	Superconductivity at millimeter wave frequencies
4	10.1103/PhysRev.78.487	Superconductivity of isotopes of mercury
5	10.1103/PhysRev.79.845	Theory of the superconducting state. i. the ground ...
6	10.1103/PhysRev.80.567	Wave functions for superconducting electrons
7	10.1103/PhysRev.79.167	The hyperfine structure of ni^{61}
8	10.1103/PhysRev.97.1724	Theory of the Meissner effect in superconductors
9	10.1103/PhysRev.81.829	Relation between lattice vibration and London ...
10	10.1103/PhysRev.104.844	Transmission of superconducting films ...

TABLE III: Spreading of impact on "Absence of diffusion in certain random lattices" paper by P. W. Anderson (doi:10.1103/PhysRev.109.1492) by Google matrix G. with $\alpha = 0.85$ and $\gamma = 0.5$

ImpactRank	DOI	Title of paper
1	10.1103/PhysRev.109.1492	Absence of diffusion in certain random lattices
2	10.1103/PhysRev.91.1071	Electronic structure of f centers: saturation of ...
3	10.1103/RevModPhys.15.1	Stochastic problems in physics and astronomy
4	10.1103/PhysRev.108.590	Quantum theory of electrical transport phenomena
5	10.1103/PhysRev.48.755	Theory of pressure effects of foreign gases on spectral lines
6	10.1103/PhysRev.105.1388	Multiple scattering by quantum-mechanical systems
7	10.1103/PhysRev.104.584	Spectral diffusion in magnetic resonance
8	10.1103/PhysRev.74.206	A note on perturbation theory
9	10.1103/PhysRev.70.460	Nuclear induction
10	10.1103/PhysRev.90.238	Dipolar broadening of magnetic resonance lines ...

TABLE IV: Spreading of impact on "Theory of superconductivity" paper by "J. Bardeen, L. N. Cooper and J. R. Schrieffer (doi:10.1103/PhysRev.108.1175) by Google matrix G^* with $\alpha = 0.85$ and $\gamma = 0.5$

ImpactRank	DOI	Title of paper
1	10.1103/PhysRev.108.1175	Theory of superconductivity
2	10.1103/PhysRevB.77.104510	Temperature-dependent gap edge in strong-coupling ...
3	10.1103/PhysRevC.79.054328	Exact and approximate ensemble treatments of thermal ...
4	10.1103/PhysRevB.8.4175	Ultrasonic attenuation in superconducting molybdenum
5	10.1103/RevModPhys.62.1027	Properties of boson-exchange superconductors
6	10.1103/PhysRev.188.737	Transmission of far-infrared radiation through thin films ...
7	10.1103/PhysRev.167.361	Superconducting thin film in a magnetic field - theory of ...
8	10.1103/PhysRevB.77.064503	Exact mesoscopic correlation functions of the Richardson ...
9	10.1103/PhysRevB.10.1916	Magnetic field attenuation by thin superconducting lead films
10	10.1103/PhysRevB.79.180501	Exactly solvable pairing model for superconductors with ...

$(Q/N)\,\chi_{[0,2/Q[}(G)$ where $\chi_{[0,2/Q[}(G)$ is the characteristic function on the interval $[0, 2/Q[$ (with values being 1 for G in this interval and 0 for G outside this interval). The average is indeed $\langle G \rangle = 1/N$ and the vari-

TABLE V: Spreading of impact on "Absence of diffusion in certain random lattices" paper by P. W. Anderson (doi:10.1103/PhysRev.109.1492) by Google matrix G^*. with $\alpha = 0.85$ and $\gamma = 0.5$

ImpactRank	DOI	Title of paper
1	10.1103/PhysRev.109.1492	Absence of diffusion in certain random lattices
2	10.1103/PhysRevA.80.053606	Effects of interaction on the diffusion of atomic ...
3	10.1103/RevModPhys.80.1355	Anderson transitions
4	10.1103/PhysRevE.79.041105	Localization-delocalization transition in hessian ...
5	10.1103/PhysRevB.79.205120	Statistics of the two-point transmission at ...
6	10.1103/PhysRevB.80.174205	Localization-delocalization transitions ...
7	10.1103/PhysRevB.80.024203	Statistics of renormalized on-site energies and ...
8	10.1103/PhysRevB.79.153104	Flat-band localization in the Anderson-Falicov-Kimball model
9	10.1103/PhysRevB.74.104201	One-dimensional disordered wires with Poschl-Teller potentials
10	10.1103/PhysRevB.71.235112	Critical wave-packet dynamics in the power-law bond ...

TABLE VI: Spreading of impact on the article of "Napoleon" in English Wikipedia by Google matrix G and G^*. with $\alpha = 0.85$ and $\gamma = 0.5$

ImpactRank	Articles (G case)	Articles (G^* case)
1	Napoleon	Napoleon
2	French Revolution	List of orders of battle
3	France	Lists of state leaders by year
4	First French Empire	Names inscribed under the Arc de Triomphe
5	Napoleonic Wars	List of battles involving France
6	French First Republic	Order of battle of the Waterloo Campaign
7	Saint Helena	Napoleonic Wars
8	French Consulate	Wagram order of battle
9	French Directory	Departments of France
10	National Convention	Jena-Auerstedt Campaign Order of Battle

ance is $\sigma^2 = 4/(3NQ)$ (for $N \gg Q$) providing the spectral radius $R = 2/\sqrt{3Q}$. We may also consider a sparse RPFM where we have exactly Q non-vanishing constant elements of value $1/Q$ in each column with random positions resulting in a variance $\sigma^2 = 1/(NQ)$ and $R = 1/\sqrt{Q}$. The theoretical predictions for these three variants of RPFM coincide very well with numerical simulations. In Fig. 16 the complex eigenvalue spectrum for one realization of each of the three cases is shown for $N = 400$ and $Q = 20$ clearly confirming the circular uniform eigenvalue density with the theoretical values of R. We also confirm numerically the scaling behavior of R as a function of N or Q.

Motivated by the Google matrices of DNA sequences [43], where the matrix elements are distributed with a power law, we also considered a power law variant of RPFM with $p(G) = D(1 + aG)^{-b}$ for $0 \leq G \leq 1$ and with an exponent $2 < b < 3$. The condition $G \leq 1$ is required because of the column sum normalization. The parameters D and a are determined by normalization and the average $\langle G \rangle = 1/N$. In the limit $N^{b-2} \gg 1$ we find $a \approx N/(b-2)$ and $D \approx N(b-1)/(b-2)$. For $b > 3$ the variance would scale with $\sim N^{-2}$ resulting in $R \sim 1/\sqrt{N}$ as in the first variant with uniformly distributed matrix elements. However, for $b < 3$ this scaling is different and we find (for $N^{b-2} \gg 1$) :

$$R = C(b)\, N^{1-b/2} \quad , \quad C(b) = (b-2)^{(b-1)/2} \sqrt{\frac{b-1}{3-b}} \quad . \tag{25}$$

Fig. 17 shows the results of numerical diagonalization for one realization with $N = 400$ and $b = 2.5$ such that we expect $R \sim N^{-0.25}$. It turns out that the circular eigenvalue density is rather well confirmed and the "theoretical radius" is indeed given by $R = \sqrt{N}\sigma$ if the variance σ^2 of matrix elements is determined by an av-

erage over the N^2 matrix elements of the given matrix. A study for different values of N with $50 \leq N \leq 2000$ also confirms the dependence $R = C\, N^{-\eta}$ with fit values $C = 0.67 \pm 0.03$ and $\eta = 0.22 \pm 0.01$. The value of $\eta = 0.22$ is close to the theoretical value $1 - b/2 = 0.25$ but the prefactor $C = 0.67$ is smaller than its theoretical value $C(2.5) \approx 1.030$. This is due to the correlations introduced by the additional column sum normalization after drawing the random matrix elements. Furthermore for the power law model with $b < 3$ we should not expect a precise confirmation of the uniform circular density obtained for Gaussian distribution matrix elements. Actually, a more detailed numerical analysis of the density shows that the density for the power law model is not exactly uniform, in particular for values of b close to 2.

The important observation is that a generic RPFM (full, sparse or with power law distributed matrix elements) has a complex eigenvalue density rather close to a uniform circle of a quite small radius (depending on the parameters N, Q or b). The fact, that the realistic networks (e.g. certain university WWW-networks) have Google matrix spectra very different from this [10], shows that in these networks there is indeed a subtle network structure and that already slight random perturbations or variations immediately result in uniform circular eigenvalue spectra. This was already observed in [8, 9], where it was shown that certain modest random changes in the network links already provide such circular eigenvalue spectra.

We also determine the PageRank for the different variants of the RPFM, i.e. the eigenvector for the eigenvalue $\lambda = 1$. It turns out that it is rather uniform that is rather natural since this eigenvector should be close to the uniform vector e which is the "PageRank" for the average matrix $\langle G_{ij} \rangle = 1/N$. This also holds when we use a damping factor $\alpha = 0.85$ for the RPFM.

Following the above discussion about triangular networks (with $G_{ij} = 0$ for $i \geq j$) we also study numerically a triangular RPFM where for $j \geq 2$ and $i < j$ the matrix elements G_{ij} are uniformly distributed in the interval $[0, 2/(j-1)[$ and for $i \geq j$ we have $G_{ij} = 0$. Then the first column is empty, that means it corresponds to a dangling node and it needs to be replaced by $1/N$ entries. For the triangular RPFM the situation changes completely since here the average matrix $\langle G_{ij} \rangle = 1/(j-1)$ (for $i < j$ and $j \geq 2$) has already a non-trivial structure and eigenvalue spectrum. Therefore the argument of degenerate perturbation theory which allowed to apply the results of standard full non-symmetric random matrices does not apply here. In Fig. 16 one clearly sees that for $N = 400$ the spectra for one realization of a triangular RPFM and its average are very similar for the eigenvalues with large modulus but both do not have at all a uniform circular density in contrast to the RPRM models without the triangular constraint discussed above. For the triangular RPFM the PageRank behaves as $P(K) \sim 1/K$ with the ranking index K being close to the natural order of nodes $\{1, 2, 3, \ldots\}$ that reflects the fact that the node 1 has the maximum of $N - 1$ incoming links etc.

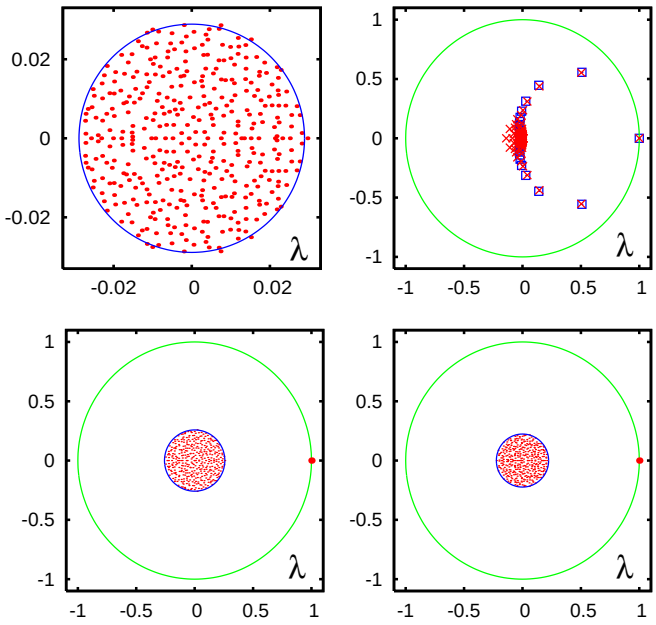


FIG. 16: (Color online) Top left panel shows the spectrum (red dots) of one realization of a full uniform RPFM with dimension $N = 400$ and matrix elements uniformly distributed in the interval $[0, 2/N[$. The blue circle represents the theoretical spectral border with radius $R = 1/\sqrt{3N} \approx 0.02887$. The unit eigenvalue $\lambda = 1$ is not shown due to the zoomed presentation range. Top right panel shows the spectrum of one realization of triangular RPFM (red crosses) with non-vanishing matrix elements uniformly distributed in the interval $[0, 2/(j-1)[$ and a triangular matrix with non-vanishing elements $1/(j-1)$ (blue squares). Here $j = 2, 3, \ldots, N$ is the index-number of non-empty columns and the first column with $j = 1$ corresponds to a dangling node with elements $1/N$ for both triangular cases. Bottom panels show the complex eigenvalue spectrum (red dots) of a sparse RPFM with dimension $N = 400$ and $Q = 20$ non-vanishing elements per column at random positions. The left (right) panel corresponds to the case of uniformly distributed non-vanishing elements in the interval $[0, 2/Q[$ (constant non-vanishing elements being $1/Q$). The blue circle represents the theoretical spectral border with radius $R = 2/\sqrt{3Q} \approx 0.2582$ ($R = 1/\sqrt{Q} \approx 0.2236$). In both bottom panels $\lambda = 1$ is shown by a larger red dot for better visibility. The unit circle is shown by green line (top right and bottom panels).

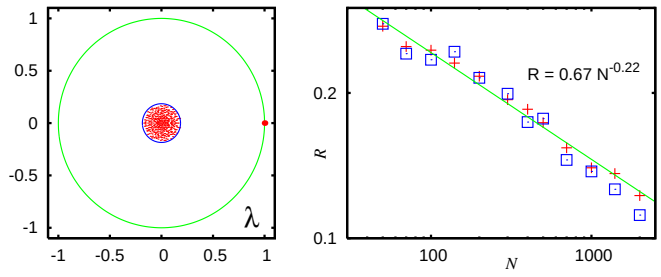


FIG. 17: (Color online) Left panel shows the spectrum (red dots) of one realization of the power law RPFM with dimension $N = 400$ and decay exponent $b = 2.5$ (see text). The unit eigenvalue $\lambda = 1$ is shown by a large red dot, the unit circle is shown by green curve. The blue circle represents the spectral border with theoretical radius $R = \approx 0.1850$ (see text). Right panel shows the dependence of the spectrum border radius on matrix size N for $50 \leq N \leq 2000$. Red crosses represent the radius obtained from theory (see text). Blue squares correspond to the spectrum border radius obtained numerically from a small number of eigenvalues with maximal modulus. The green line shows the fit $R = C\, N^{-\eta}$ of red crosses with $C = 0.67 \pm 0.03$ and $\eta = 0.22 \pm 0.01$.

The study of above models shows that it is not so simple to find a good RPFM model which reproduces a typical spectral structure of real directed networks.

VIII. DISCUSSION

In this study we presented a detailed analysis of the spectrum of the CNPR for the period 1893 – 2009. It happens that the numerical simulations should be done with a high accuracy (up to $p = 16384$ binary digits for the rational interpolation method or $p = 768$ binary digits for the high precision Arnoldi method) to determine correctly the eigenvalues of the Google matrix of CNPR at small eigenvalues λ. Due to the time ordering of citations, the CNPR G matrix is close to the triangular form with a nearly nilpotent matrix structure. We show that special semi-analytical methods allow to determine efficiently the spectrum of such matrices. The eigenstates with large modulus of λ are shown to select specific communities of articles in certain research fields but there is no clear way on how to identify a community one is interested in. The obtained results show that the spectrum of CNPR is characterized by the fractal Weyl law with the fractal dimension $d_f \approx 1$.

The ranking of articles is analyzed with the help of PageRank and CheiRank vectors corresponding to forward and backward citation flows in time. It is shown that the correlations between these two vectors are small and even negative that is similar to the case of Linux Kernel networks [26] and significantly different from networks of universities and Wikipedia. The 2DRanking on the PagRank-CheiRank plane allows to select articles which efficiently redistribute information flow on the CNPR.

To characterize the local impact propagation for a given article we introduce the concept of ImpactRank which efficiently determines its domain of influence.

Finally we perform the analysis of several models of RPFM showing that such full random matrices are very far from the realistic cases of directed networks. Random sparse matrices with a limited number Q of links per nodes seem to be closer to typical Google matrices concerning the matrix structure. However, still such random models give a rather uniform eigenvalue density with a spectral radius $\sim 1/\sqrt{Q}$ and also a flat PageRank distribution. Furthermore they do not capture the existence of quasi-isolated communities which generates quasi-degenerate spectrum at $\lambda = 1$. Further development of RPFM models is required to reproduce the spectral properties of real modern directed networks.

IX. ACKNOWLEDGMENTS

We thank the American Physical Society for letting us use their citation database for Physical Review [15]. This research is supported in part by the EC FET Open project "New tools and algorithms for directed network analysis" (NADINE *No* 288956). This work was granted access to the HPC resources of CALMIP (Toulouse) under the allocation 2012-P0110.

[1] S. Brin and L. Page, Computer Networks and ISDN Systems **30**, 107 (1998).

[2] A.A. Markov, *Rasprostranenie zakona bol'shih chisel na velichiny, zavisyaschie drug ot druga*, Izvestiya Fiziko-matematicheskogo obschestva pri Kazanskom universitete, 2-ya seriya, **15**, 135 (1906) (in Russian) [English trans.: *Extension of the limit theorems of probability theory to a sum of variables connected in a chain* reprinted in Appendix B of Howard RA *Dynamic Probabilistic Systems*, volume 1: *Markov models*, Dover Publ. (2007)].

[3] M. Brin and G. Stuck, *Introduction to Dynamical Systems*, Cambridge University Press, Cambridge, England, 2002.

[4] A. M. Langville and C. D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press (Princeton, 2006).

[5] M.L.Mehta, *Random matrices*, Elsevier-Academic Press, Amsterdam (2004).

[6] D.L.Shepelyansky and O.V.Zhirov, Phys. Rev. E **81**, 036213 (2010).

[7] L.Ermann and D.L.Shepelyansky, Eur. Phys. J. B **75**, 299 (2010).

[8] O.Giraud, B.Georgeot and D.L.Shepelyansky, Phys. Rev. E **80**, 026107 (2009).

[9] B.Georgeot, O.Giraud and D.L.Shepelyansky, Phys. Rev. E **81**, 056109 (2010).

[10] K.M.Frahm, B.Georgeot and D.L.Shepelyansky, J. Phys, A: Math. Theor. **44**, 465101 (2011)

[11] L.Ermann, A.D.Chepelianskii and D.L.Shepelyansky, Eur. Phys. J. B **79**, 115 (2011).

[12] K.M.Frahm and D.L.Shepelyansky, Eur. Phys. J. B **85**, 355 (2012).

[13] L.Ermann, K.M.Frahm and D.L.Shepelyansky, Eur. Phys. J. B **86**, 193 (2013).

[14] Y.-H.Eom, K.M.Frahm, A.Benczur and D.L.Shepelyansky, preprint arXiv:1304.6601 [physics.soc-ph] (2013).

[15] Web page of Physical Review http://publish.aps.org/

[16] R. Albert and A.-L. Barabási, Phys. Rev. Lett. **85**, 5234 (2000).

[17] G. W. Stewart, *Matrix Algorithms Volume II: Eigensystems*, SIAM, 2001.

[18] K.M. Frahm and D.L. Shepelyansky, Eur. Phys. J. B **76**, 57 (2010).

[19] K.M.Frahm, A.D.Chepelianskii and D.L.Shepelyansky, J. Phys. A: Math. Theor. **45**, 405101 (2012).

[20] S. Redner, Phys. Today **58(6)**, 49 (2005)

[21] P. Chen, H. Xie, S. Maslov and S.Redner, J. Infometrics **1**, 8 (2007)

[22] F. Radicchi, S. Fortunato, B. Markines and A. Vespignani, Phys. Rev. E **80**, 056103 (2009).

[23] J.D. West, T.C. Bergstrom and C.T. Bergstrom, Coll. Res. Libr. **71**, 236 (2010); http://www.eigenfactor.org/

[24] A.D. Chepelianskii, *Towards physical laws for software architecture*, preprint arXiv:1003.5455[cs.Se] (2010)

[25] A.O.Zhirov, O.V.Zhirov and D.L.Shepelyansky, Eur. Phys. J. B **77**, 523 (2010)

[26] L.Ermann, A.D.Chepelianskii and D.L.Shepelyansky, J. Phys. A: Math. Theor. **45**, 275101 (2012)

[27] This number depends on the exact time ordering which is used and which is not unique because many papers are published at the same time and the order between them is not specified. We have chosen a time ordering where between these papers, degenerate in publication time, the initial node order of the raw data is kept.

[28] Note that some of the non-vanishing components of the iteration vector $S_0^i e$ may become very small, e.g. $\sim 10^{-100}$. In this context we count such components still as occupied despite their small size and N_i is the number of nodes which can be reached from some arbitrary other node after i iterations with the matrix S_0.

[29] In [19] a set of vectors without this prefactor was used but this provided a representation matrix which is numerically unstable for a direct diagonalization. The prefactor c_{j-1}^{-1} ensures that the representation matrix is numerically (rather) stable and of course both matrices are mathematically related by a similarity transformation and have identical eigenvalues.

[30] T. Granlund and the GMP development team, *GNU MP: The GNU Multiple Precision Arithmetic Library*, Version 5.0.5 (2012), http://gmplib.org/.

[31] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Springer (2002).

[32] J. Sjöstrand, Duke Math. J. **60**, 1 (1990).

[33] J. Sjöstrand and M. Zworski, Duke Math. J. **137**, 381 (2007).

[34] S. Nonnenmacher and M. Zworski, Commun. Math. Phys. **269**, 311 (2007).

[35] J. Bardeen, L. N. Cooper, and J. R. Schrieffer, Phys. Rev. **108**, 1175 (1957).

[36] P.W. Anderson, Phys. Rev. **109**, 1492 (1958)

[37] G. Benettin, L. Galgani, and J.-M. Strelcyn, Phys. Rev. A **14**, 2338 (1976).

[38] D.J. Thouless, Phys. Rev. Lett. **39**, 1167 (1977).

[39] E. Abrahams, P.W. Anderson, D.C. Licciardello, and T.V. Ramakrishnan, Phys. Rev. Lett. **42**, 673 (1979).

[40] Y.-H. Eom and D.L. Sepelyansky, PLoS ONE **8(10)**, e74554 (2013).

[41] J. Ginibre, J. Math. Phys. Sci. **6**, 440 (1965).

[42] H.-J. Sommers, A. Crisanti, H. Sompolinsky and Y. Stein, Phys. Rev. Lett. **60**, 1895 (1988); N. Lehmann and H.-J. Sommers, Phys. Rev. Lett. **67**, 941 (1991).

[43] V. Kandiah and D. L. Shepelyansky, PLoS One **8(5)**, e61519 (2013).

Temporal influence over the Last.fm social network

Róbert Pálovics[1,2] András A. Benczúr[1,3]
[1]Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI)
[2]Technical University Budapest
[3]Eötvös University Budapest
{rpalovics,benczur}@ilab.sztaki.hu

***Abstract*—Several recent results show the influence of social contacts to spread certain properties over the network, but others question the methodology of these experiments by proposing that the measured effects may be due to homophily or a shared environment. In this paper we justify the existence of the social influence by considering the temporal behavior of Last.fm users. In order to clearly distinguish between friends sharing the same interest, especially since Last.fm recommends friends based on similarity of taste, we separated the timeless effect of similar taste from the temporal impulses of immediately listening to the same artist after a friend. We measured strong increase of listening to a completely new artist in a few hours period after a friend compared to non-friends representing a simple trend or external influence. In our experiment to eliminate network independent elements of taste, we improved collaborative filtering and trend based methods by blending with simple time aware recommendations based on the influence of friends. Our experiments are carried over the two-year "scrobble" history of 70,000 Last.fm users.**

I. Introduction

Several results show the influence of friends and contacts to spread obesity [1], loneliness [2], alcohol consumption [3], religious belief [4] and many similar properties in social networks. Others question the methodology of these experiments [5] by proposing that the measured effects may be due to homophily, the fact that people tend to associate with others like themselves, and a shared environment also called confounding or contextual influence.

Part of the appeal of Web 2.0 is to find other people who share similar interests. Last.fm organizes its social network around music recommendation: users may automatically share their listening habits and at the same time grow their friendship. Based on the profiles shared, users may see what artists friends really listen to the most. Companies such as Last.fm use this data to organize and recommend music to people.

In this paper we exploit the timely information gathered by the Last.fm service on users with public profile to investigate how members of the social network may influence their friends' taste. Last.fm's service is unique in that we may obtain a detailed timeline and catch immediate effects by comparing the history of friends in time and comparing to pairs of random users instead of friends.

Our contribution to the dispute on whether social contacts influence one another or whether the observed similarity in taste and behavior is only due to homophily, we show a carefully designed experiment to subtract external effects that may result in friends listening to similar music. Homophily is handled by collaborative filtering, a method that is capable of learning patterns of similarity in taste without using friendship information. Another possible source for users listening to the same music may come from traditional media: news, album releases, concerts and ads. While the sources are hard to identify, common in them is that they cause temporal increase in popularity for the targeted artist. These effects are filtered by another method that measures popularity at the given time and recommends based on the momentary popularity.

We blend collaborative filtering and temporal popularity recommenders with a method for influence prediction that we describe in this paper. We consider events where a user listens to an artist for the first time closely after a friend listened to the same artist. We obtain a 4% of increase in recommendation quality, a strong result in view of the three-year Netflix Prize competition [6] to improve recommender quality by 10%. Note that we only give a single method that results in a stable strong improvement over the baselines.

Our new method is a lightweight recommender based on friends' past items that can be very efficiently computed even in real time. Part of the efficiency comes from the fact that potential items from influencing friends are relative rare. For this reason, the method in itself performs worse than the baselines, however it combines very well with them. Indeed, influence based predictions improve the accuracy of a traditional factor model recommender by nearly as much as measuring popularity at the given time, a prediction that is strong in itself. The fact that influences bend well prove that close events in the network bring in new information that can be exploited in a recommender system and also prove the existence of influence from friends beyond homophily.

Research supported in part by the EC FET Open project "New tools and algorithms for directed network analysis" (NADINE No 288956) and by the grant OTKA NK 105645. The work of Robert Palovics reported in this paper has been developed in the framework of the project "Talent care and cultivation in the scientific workshops of BME" project. This project is supported by the grant TAMOP - 4.2.2.B-10/1–2010-0009. Work conducted at the Eötvös University, Budapest was partially supported by the European Union and the European Social Fund through project FuturICT.hu (grant no.: TAMOP-4.2.2.C-11/1/KONV-2012-0013). The research was carried out as part of the EITKIC_12-1-2012-0001 project, which is supported by the Hungarian Government, managed by the National Development Agency, financed by the Research and Technology Innovation Fund and was performed in cooperation with the EIT ICT Labs Budapest Associate Partner Group. (www.ictlabs.elte.hu)

A. Related results

The Netflix Prize competition [6] has recently generated increased interest in recommender algorithms in the research community and put recommender algorithms under a systematic thorough evaluation on standard data [7]. The final best results blended a very large number of methods whose reproduction is out of the scope of this paper. As one of our baselines we selected a successful matrix factorization recommender described by Simon Funk in [8] that is based on an approach reminiscent of gradient boosting [9].

Closest to our results are the applications of network influence in collaborative filtering [10]. However in their data only ratings and no social contacts are given. In another result [11] over Flickr, both friendship and view information was present, but the main goal was to measure the strength of the influence and no measurements were designed to separate influence from other effects.

Bonchi [12] summarizes the data mining aspects of research on social influence. He concludes that "another extremely important factor is the temporal dimension: nevertheless the role of time in viral marketing is still largely (and surprisingly) unexplored", an aspect that is key in our result.

Since our goal is to recommend different artists at different times, our evaluation must be based on the quality of the top list produced by the recommender. This so-called top-k recommender task is known to be hard [13]. For a recent result on evaluating top-k recommenders is found in [14].

Music recommendation is considered in several results orthogonal to our methods that will likely combine well. Mood data set is created in [15]. Similarity search based on audio is given in [16]. Tag based music recommenders [17], [18, and many more], a few of them based on Last.fm tags, use annotation and fall into the class of content based methods as opposed to collaborative filtering considered in our paper. Best starting point for tag recommendation in general are the papers [19], [20], [21]. Note that the Netflix Prize competition put a strong vote towards the second class of methods [22].

As a social media service, Twitter is widely investigated for influence and spread of information. Twitter influence as followers has properties very different from usual social networks [23]. Deep analysis of influence in terms of retweets and mentions is given in [24]. Notion of influence similar to ours is derived in [25], [26] for Fickr and Twitter cascades, respectively. Note that by our measurement the Last.fm data contains only a negligible amount of cascades as opposed to Twitter or Flickr.

II. The Last.fm data set

Last.fm became a relevant online service in music based social networking. The idea of Last.fm is to create a recommendation system based on plugins nearly for all kind of music listening platforms. For registered users it collects, "scrobbles"[1] what they have listened. Each user has its own statistics on listened music that is shown in her profile.

[1]The name "scrobbling" is a word by Last.fm, meaning the collection of information about user listening.

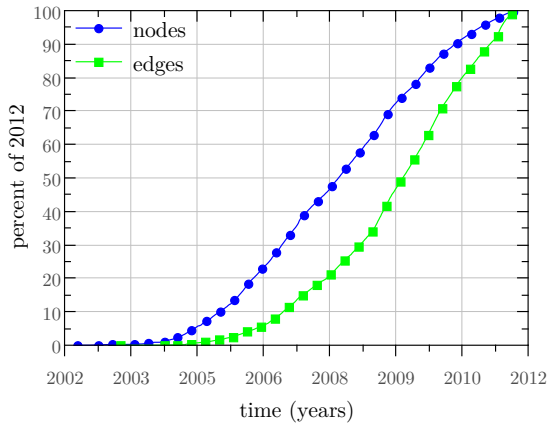


Fig. 1. The number of the users and friendship edges in time as the fraction of the values at the time of the data set creation (2012).

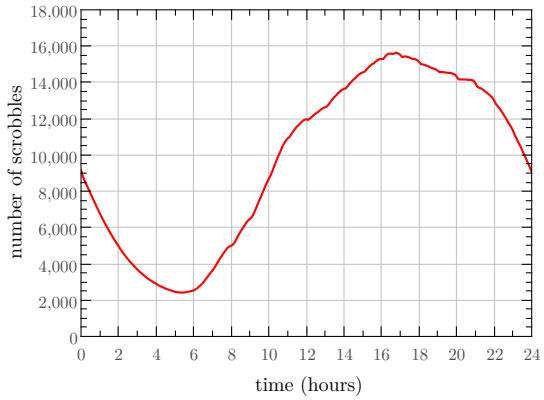


Fig. 2. Daily periodicity of scrobble count.

Most user profiles are public, and each user of Last.fm may have friends inside the Last.fm social network. Therefore one relevant information for the users is that they see their own and their friends' listening statistics. We focus on two types of user information,

- the timeline information of users: user u "scrobbled" artist a at time t (u, a, t),
- and the social network of users.

Our data set hence consists of the contacts and the musical taste of the users. Our goal is to justify the existence of the influence of social contacts, i.e. certain correlation the taste of friends in the user network. For privacy considerations, throughout our research, we selected an anonymous sample of users. Anonymity is provided by selecting random users while maintaining a connected friendship network. We set the following constraints for random selection:

- User location is stated in UK;
- Age between 14 and 50, inclusive;
- Profile displays scrobbles publicly (privacy constraint);
- Daily average activity between 5 and 500.
- At least 10 friends that meet the first four conditions.

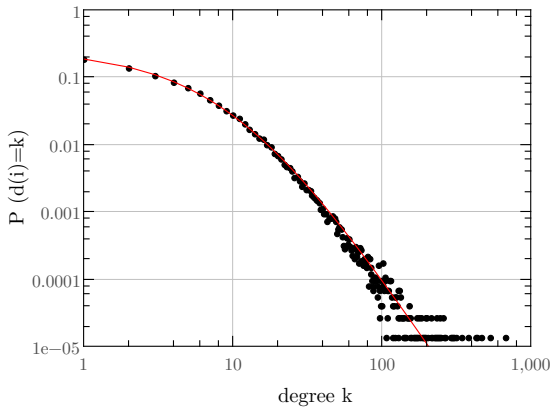


Fig. 3. Degree distribution in the friendship network.

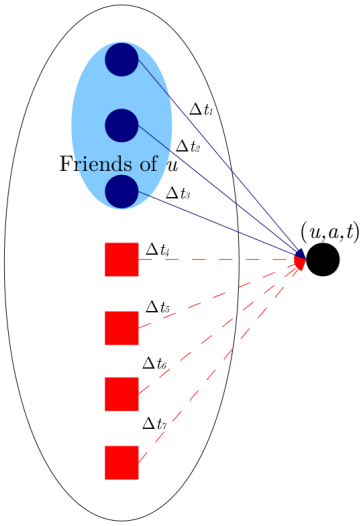


Fig. 4. Potential influence on u by some other user to scrobble (u, a, t).

The above selection criteria were set to select a representative part of Last.fm users and as much as possible avoid users who artificially generate inflated scrobble figures. In this anonymized data set of two years of artist scrobble timelime, edges of the social network are undirected and timestamped by creation date (Fig. 1). Note that no edges are ever deleted from the network.

The number of users both in the time series and in the network is 71,000 with 285,241 edges. The average degree is therefore 8, while the degree distribution follows shifted power-law as seen in Fig. 3

$$P(d(i) = k) \sim (x + s)^{-\alpha}$$

with exponent 3.8.

The time series contain 979,391,001 scrobbles from 2,073,395 artists and were collected between 01 January 2010 and 31 December 2011. Note that one user can scrobble an artist at different times. The number of unique user-artist scrobbles is 57,274,158. Fig. 2 shows the daily fluctuations in the users scrobbling activity.

III. Notion of network influence

The key concept in this paper is a user v *influencing* another u to scrobble a. This happens if u scrobbles artist a the *first time* at time t, after v *last scrobbling* the same artist at some time $t' < t$ before. The time difference $\Delta t = t - t'$ is the *delay* of the influence, as seen in Fig. 4. Our key assumption is that, in the above definition, we observe influences between non-friends only by coincidence while some of the observed influence between friends is the result of certain interaction between them. Our goal is to prove that friends indeed influence each other and this effect can be exploited for recommendations.

Similar influence definitions are given in [11], [25], [26]. As detailed in [26], one main difference between these definitions is that in some papers t' is defined as the first and not the last time when user v scrobbles a.

For smaller influence delay Δt, we are more certain that u is affected by the previous scrobble of v. The distribution of delay with respect to friends and non-friends will help us in determining the frequency and strength of influence over the Last.fm social network. Each time user u first scrobbles a, we compute the delay Δt for all users v who scrobbled a before u, if such users exist (see Fig. 4).

Out of the 57,274,158 first-time scrobbles of certain artist a by some user, we find a friend who scrobbled a before 10,993,042 times (19%). Note that one user can be influenced by more friends therefore the total number of influences is 24,204,977. There is no influencing user for the very first scrobbler of a in the data set. For other scrobbles there is always an earlier scrobble by some other user, however that user may not be a friend of u.

Some of the observed influences may result by pure coincidence, especially when a new album is released or the popularity of the artist increases for some other reason. In order to identify real influence, we compare the frequency of influence from friends and from non-friends along delay Δt as parameter. We compute the cumulative distribution function of all influences as a function of the delay,

$$\mathrm{CDF}_A(t) = \text{fraction of influences with delay } \Delta t \le t \text{ among all influences.} \tag{1}$$

Similarly, $\mathrm{CDF}_F(t)$ stands for the same function among influences between friends only. Fig. 5 shows the functions for all users and friends. The function of friends is above that of all users, i.e. we observe shorter delay more frequently among friends.

Next we quantify the importance of friendship in influencing others as the *effectivity* function. The effectivity at Δt is defined as the increase of influenced scrobbles among friends relative to all users that happen with delay at most t:

$$\mathrm{Eff}(t) = \frac{\mathrm{CDF}_F(t) - \mathrm{CDF}_A(t)}{\mathrm{CDF}_F(t)}. \tag{2}$$

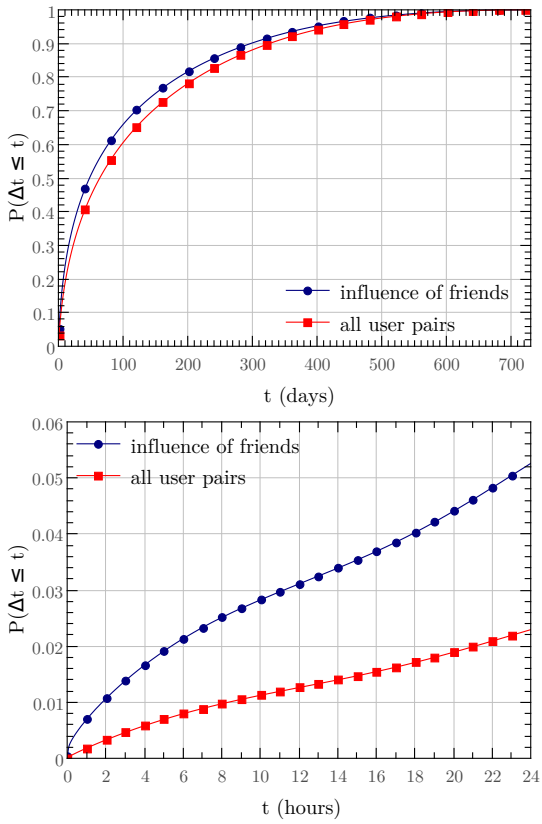


Fig. 5. Fraction of influences with delay $\Delta t \leq t$ as the function of t as in (1), in case of friends (CDF_F) and non-friends (CDF_A) over the entire timeline (**top**) and the first 24 hours (**bottom**).

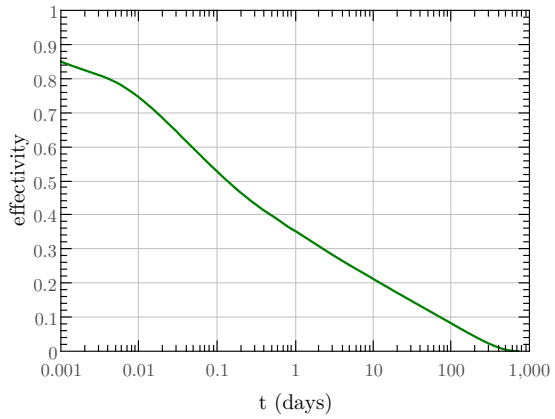


Fig. 6. The measured effectivity of the influence (ratio of increase among friends compared to all users) as in (2) very closely follows a logarithmic function of delay Δt.

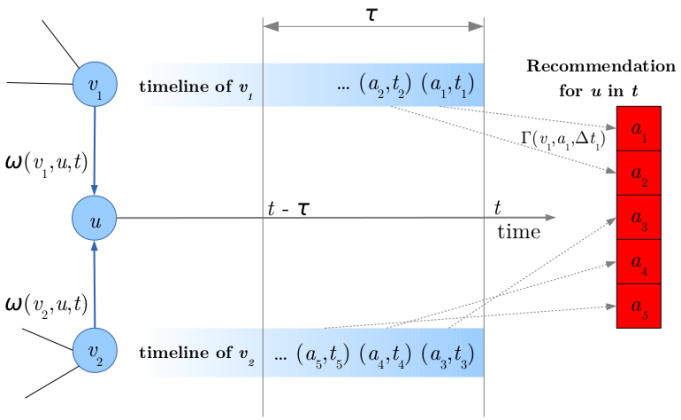


Fig. 7. Scheme of the influence based recommender algorithm.

Fig. 6 shows the measured effectivity curve in the community. As expected, $\text{Eff}(t)$ is a monotonically decreasing function of t. However, the decrease is slow unlike in some recent influence models that propose exponential decay in time [11]. Therefore, we approximate $\text{Eff}(t)$ with a slowly decreasing logarithmic function instead of an exponential decay.

IV. Influence Based Recommendation

Next we use our notion of influence in the task of artist recommendation. Influence depends on time and no matter how relative slow but the effectivity of a friend scrobbling an artist decays. For this reason the influence based recommendation must be updated more frequently than traditional collaborative filtering methods. Also note that for a given user, our recommendation can be computed very efficiently by a pass over the recent history of friends.

Based on the measurements in the previous Section, we give a temporal network influence based recommender algorithm. For a user u at time t, we recommend based on friends' scrobbles before t. The predicted score $\hat{r}(u, a, t)$ of an artist a is based on a function Γ of the time elapsed since the friend v scrobbling a (the delay Δt) and a function ω of the observed frequency of v influencing u in the past, as summarized in Fig. 7. Formally the predicted rating becomes

$$\hat{r}(u, a, t) = \sum_{v \in n(u)} \Gamma(v, a, \Delta t)\omega(v, u, t), \qquad (3)$$

where $n(u)$ denotes the friends of u, $\omega(v, u, t)$ is the strength of the influence between users u and v, and $\Gamma(v, a, \Delta t)$ is the weight between user v and artist a based on the delay.

Our implementation depends on the two functions ω and Γ defined in the next two subsections. In an efficient algorithm, the value of ω can be stored in memory for all pairs of friends. Alternately, ω can only be batch updated as the strength between two users are less time sensitive. The values of Γ, however, depend on the actual time when the recommendation is requested. As Γ quickly decays with Δt, we only need to retrieve the past srobbles of all v, the friends of u. This step can be efficiently implemented unless u has too many friends. In this latter case we could select only a few influencing friends based on the values of ω, otherwise the recommendation is noisy anyway. Our algorithm can hence be implemented even in real time.

A. Influence as function of delay

The potential of influence decays as time elapses since the influencer v scrobbled the given artist a. Based on the effectivity curve (see Fig. 6) we approximate the strength of the influence with a monotonically decreasing logarithmic function

$$\Gamma(v, a, \Delta t) = 1 - C \cdot \log(\Delta t), \tag{4}$$

where C is a global constant.

B. Strength of influence between user pairs

We recommend a recent scrobble by a friend by taking both the recency of the scrobble and the observed relation between the two users. For each pair of users u, the influenced and v, the influencer, we define the strength $\omega(v, u, t)$ as a step function in time as follows:

- We initialize $\omega(v, u, 0) = 0$ for all pairs.
- Assume that u and v become friends at time t_0. We take a step and set $\omega(u, v, t_0) = \omega(v, u, t_0) = 1$.
- If we observe an influence from v to u at time $t > t_0$ with time difference Δt, we take another step and increase $\omega(v, u, t)$ by

$$\omega(v, u, t) \leftarrow \omega(v, u, t) + (1 - C \cdot \log(\Delta t)), \tag{5}$$

where C is a global constant. For simplicity we use the same logarithmic function of the delay as in (4).

To speed up computations, we only consider influence with delay not more than a predefined time frame τ. We apply τ for defining both ω in (5) and Γ in (4) and hence in both cases we set

$$C = 1/\log(\tau). \tag{6}$$

V. Real time recommendation evaluation

Recommender systems in practice need to rank the best k items for the user in real time. In the so-called top-k recommendation task [13], [14], potentially we have to compute a new top list for every single scrobble in the test period. The top-k task is different from the standard recommender evaluation settings and needs carefully selected metrics that we describe next.

Out of the two year scrobbling data, we use the full first year as training period. The second year becomes the testing period where we consider scrobbles one by one. We allow a recommender algorithm to use part or full of the data before the scrobble in question for training and require a ranked top list of artists as output. We evaluate the given single actual scrobble a in question against the recommended top list by computing the discounted cumulative gain with treshold K

$$\text{DCG@K}(a) = \begin{cases} 0 & \text{if rank } (a) > K; \\ \dfrac{1}{\log_2(\text{rank}(a) + 1)} & \text{otherwise.} \end{cases} \tag{7}$$

Note that in this unusual setting there is a single relevant item and hence for example no normalization is needed as in case of the NDCG measure. Also note that the DCG values will be small since the NDCG of a relative short sequence of actual scrobbles will roughly be equal to the sum of the individual DCG values. The DCG measured over 100 subsequent scrobbles of different artists cannot be more than the ideal DCG, which is $\sum_{i=1}^{100} 1/\log_2(i+1) = 20.64$ in this case (the ideal value is 6.58 for $K = 20$). Hence the DCG of an individual scrobble will on average be less than 0.21 for $K = 100$ and 0.33 for $K = 20$.

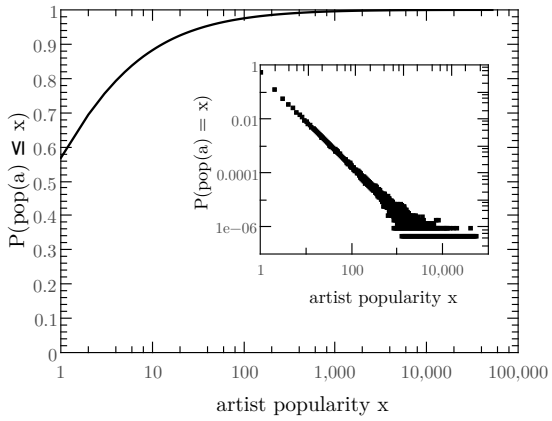


Fig. 8. Distribution of scrobble count to a given artist and the cumulative distribution.

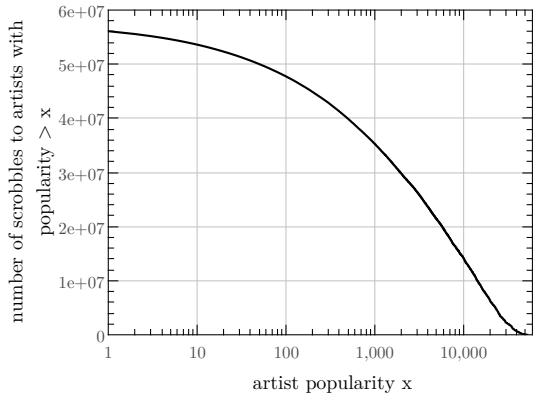


Fig. 9. Fraction of scrobbles for artists with popularity at least a given value x, as the function of x.

In our evaluation we discard infrequent artists from the data set both for efficiency considerations and due to the fact that our item based recommenders will have too little information on them. As seen in Fig. 8, the number of artists with a given scrobble count follow a power law distribution with near 60% of the artists appearing only once. While 90% of the artists gathered less than 20 scrobbles in two years, as seen in Fig. 9, they attribute to only less than 10% of the data set. In other words by discarding a large number of artists, we only loose a small fraction of the scrobbles. For efficiency we only consider artists of frequency more than 14.

As time elapses, we observe near linear increase in the number of artists that appear in the data set in Fig. 10. This

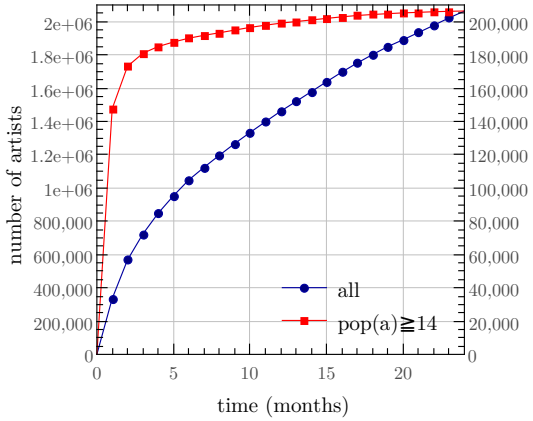


Fig. 10. The number of different artists scrobbled before a given time in the two year period of the data set.

figure shows artists with at least 14 scrobbles separately. Their count grows slower but still we observe a large number of new artist that appear in time and exceed the minimum count of 14. Very fast growth for infrequent artists may be a result of noise and unidentified artists from e.g. YouTube videos and similar Web sources.

VI. Music Recommendation Baseline Methods

We describe one baseline method based on dynamic popularity in Section VI-A and one based on factorization in Section VI-B.

A. Dynamic popularity based recommendation

Given a predefined time frame τ as in Section IV, at time t we recommend an artist based on the popularity in time not earlier than $t - \tau$ but before t. In our algorithm we update the counts and store artists sorted by the current popularity. In one time step we may either add a new scrobble event or remove the earliest one, corresponding to a count increment or decrement. For globally popular items the sorted order can be maintained by a few changes in the order only. To speed up the procedure, we may completely ignore part of the long tail and for others update the position only after a sufficiently large change in count. As future work we could also consider bursts and predict the popularity increase or decrease.

B. Factor model based recommendation

For our factor model based recommender we selected the implementation of Funk [8]. In the testing period we trained weekly models based on all data before the given week. For each user, we constructed three times as many negative training instances as positive by selecting random artists with probability proportional to their popularity in the training period. Each testing period lasted one week. For each user, we compute a top list of predictions once for the entire week and evaluate against the sequence of scrobbles in that week.

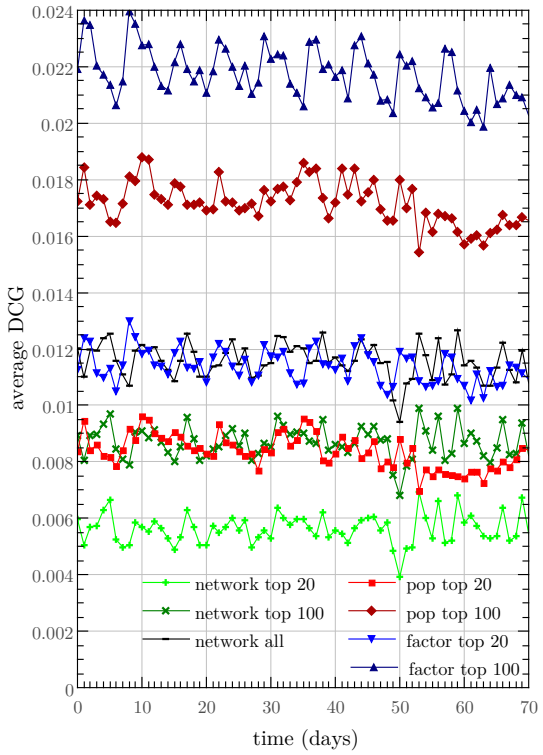


Fig. 11. Daily average DCG@K as in (7) in a 70-day sample of the test period. We show the three basic methods, from strongest to weakest, the factorization, temporal popularity, and the network influence recommenders. For K we measure two values, 20 and 100, except for network influence where we also show $K = \infty$ as the entire ranked list can be efficiently computed in this case.

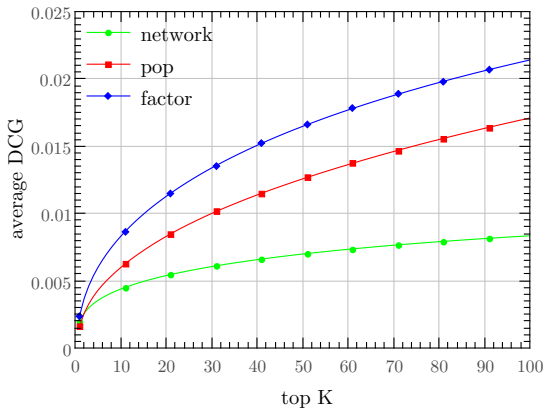


Fig. 12. DCG@K as the function of K for the three basic algorithms, for a time window τ equal to one week.

VII. Experiments

First we give the daily average DCG@K defined by equation (7) in the second year testing period for the influence based and the two baseline recommenders. Parameter K in equation (7) controls the length of the top list considered for evaluation.

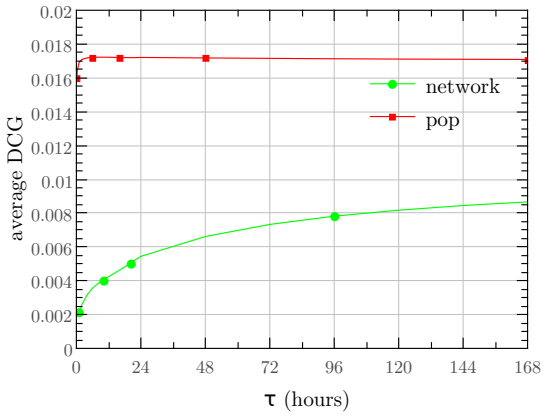


Fig. 13. DCG@100 defined by equation (7) as the function of the time window threshold τ as in Section IV-A.

In other words, K can be interpreted as the size of the list presented to the user. Practically K must be small in order not to flood the user with information. The performance of the three basic methods is shown in Fig. 11 for $K = 20$ and 100 and a time window τ in Section IV-A equal to one week.

The dependence on the top list size K is measured in Fig. 12 for $K \leq 100$. We observe that our influence based method saturates the fastest. This is due to the fact that the number of items recommended to a given user is usually small unless the user has a large number of very active friends. For this reason we give blending results not just for the value $K = 20$ that we consider practically feasible but also for 100 for comparison.

Next we investigate the parameters of the individual algorithms. For a matrix factorization based method we use Funk's algorithm [8] with the following parameters that turned out to perform best in our experiments: learning rate $= 0.001$, feature number $= 20$, and initial feature value $= 0.1$. We re-train the algorithm each week based on all past data. For this reason we see weekly periodicity in the 10-week timeline of Fig. 11: the factor model performs best immediately after the training period and slowly degrades in the testing period.

The popularity and influence based methods depend on the time frame: the longer we look back in time, the more artists we can recommend. If we carefully set the rank as a function of time, wider time frames are advantageous for quality but put extra computational load. For the influence recommender τ is the maximum delay Δt that we consider as influence while for the popularity one τ is the time interval that we use for frequency computation. We ran measurements in the second year test period with different time frames τ and computed the average DCG performance of the recommender systems. Figure 13 shows the average DCG scores with different time frames. The performance only slowly increases for time frames longer than a day. In what follows we set τ to be one week.

The final conclusion of the experiments is drawn by blending the three recommenders as shown in Figs. 15–14. In our experiments we obtained the best results by linearly combining 1/rank instead of the predicted score. As an advantage of 1/rank, we need no score normalization.

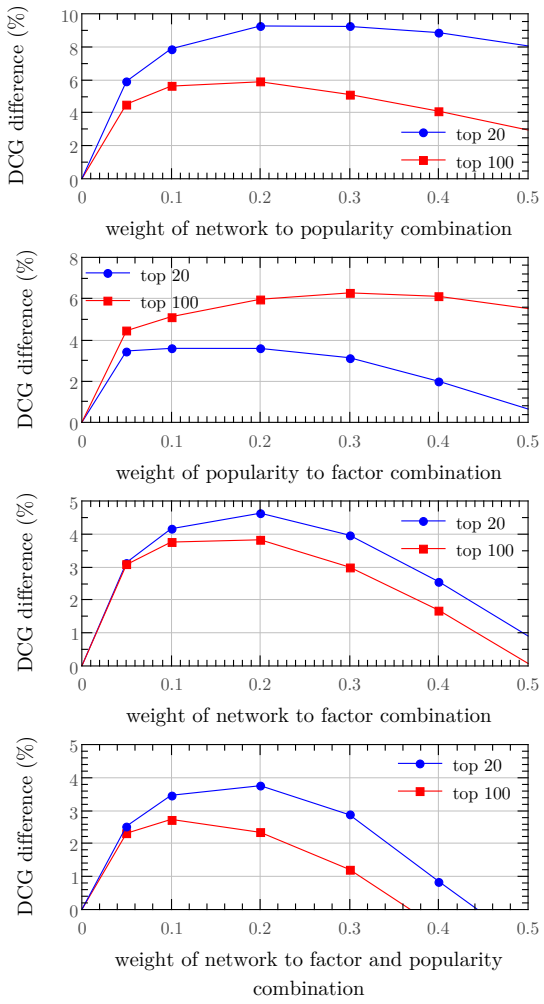


Fig. 14. Blending DCG@K defined by (7) as the function of the linear combination weight. From top to bottom: network influence and factor model; temporal popularity and factor model; network influence and factor model; finally network influence and the strogest combination of factor with popularity.

Figure 14 shows the relative improvement of the recommenders as the function of the blending weights. After blending the recommenders pairwise, we selected the strongest popularity-factor combinations (3:7 and 2:8) and blended it with the network recommender. One can see that the influence recommender not only improves the results of the factor and popularity recommenders, but combines well with their best blended result: the combination of the three methods outperforms the best blend of the factor and popularity models both for DCG@20 and DCG@100. The improvement is roughly 4%. Figure 15 shows the monthly average DCG@20 and DCG@100 curves in the testing period in case of the different blended recommenders. Each curve shows the result of the best combination of the corresponnding recommenders. In each case we observe stable improvement over the entire testing period.

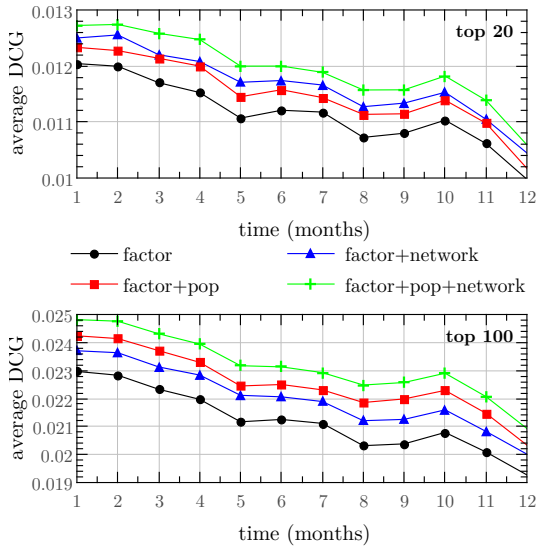


Fig. 15. Monthly average DCG@20 (**top**) and DCG@100 (**bottom**) as defined by (7) in the test period for the factor model and its combinations.

CONCLUSIONS

Based on a 70,000 sample of Last.fm users, we were able to measure the effect of certain user recommending an artist to her friends. Our results confirm the existence of influence through the social network as opposed to the pure similarity of taste between friends. We disproved the opinion that homophily could be the reason for friends listening to the same music or behave similarly by constructing a baseline that takes homophily and temporal effects into account. Over the baseline recommender, we achieved a 4% improvement in recommendation accuracy when presenting artists from friends' past scrobbles that the given user had never seen before. Our system has very strong time awareness: when we recommend, we look back in the near past and combine friends' scrobbles with the baseline methods. The influence from a friend at a given time is certain function of the observed influence in the past and the time elapsed since the friend scrobbled the given artist. In addition, our method can efficiently be computed even in real time.

For future work we plan to investigate whether the temporal social influence is specific to Last.fm dataset or can match to other kind of social network, e.g. Twitter. We also plan to break down the analysis of influence spread by type of music, by age range, or by artist.

ACKNOWLEDGEMENTS

To the Last.fm team for preparing us this volume of the anonymized data set that cannot be efficiently fetched through the public Last.fm API.

REFERENCES

[1] N. Christakis and J. Fowler, "The spread of obesity in a large social network over 32 years," *New England Journal of Medicine*, 357(4):370–379, 2007.

[2] J. Cacioppo, J. Fowler, and N. Christakis, "Alone in the crowd: The structure and spread of loneliness in a large social network.," *Journal of Personality and Social Psychology*, vol. 97, no. 6, p. 977, 2009.

[3] J. Rosenquist, J. Murabito, J. Fowler, and N. Christakis, "The spread of alcohol consumption behavior in a large social network," *Annals of Internal Medicine*, vol. 152, no. 7, p. 426, 2010.

[4] S. Stroope, "Social networks and religion: The role of congregational social embeddedness in religious belief and practice," *Sociology of Religion*, 2011.

[5] R. Lyons, "The spread of evidence-poor medicine via flawed social-network analysis," *Statistics, Politics, and Policy*, 2(1), p. 2, 2011.

[6] J. Bennett and S. Lanning, "The netflix prize," in *KDD Cup and Workshop in conjunction with KDD 2007*, 2007.

[7] R. Bell and Y. Koren, "Lessons from the Netflix prize challenge," 2007.

[8] S. Funk, "Netflix update: Try this at home. http://sifter.org/˜ simon/journal/20061211.html," 2006.

[9] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, 29(5):1189–1232, 2001.

[10] P. Domingos and M. Richardson, "Mining the network value of customers," in *SIGKDD*, pp. 57–66, ACM, 2001.

[11] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in *WSDM*, pp. 241–250, ACM, 2010.

[12] F. Bonchi, "Influence propagation in social networks: A data mining perspective," *IEEE Intelligent Informatics Bulletin*, 12(1):8–16, 2011.

[13] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM TOIS*, 22(1):143–177, 2004.

[14] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-n recommendation tasks," in *RecSys*, pp. 39–46, ACM, 2010.

[15] X. Hu, M. Bay, and J. Downie, "Creating a simplified music mood classification ground-truth set," in *ISMIR*, 2007.

[16] P. Knees, T. Pohle, M. Schedl, and G. Widmer, "A music search engine built upon audio-based and web-based similarity measures," in *Proc SIGIR*, pp. 447–454, ACM, 2007.

[17] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green, "Automatic generation of social tags for music recommendation," *Advances in neural information processing systems*, 20:385–392, 2007.

[18] K. Tso-Sutter, L. Marinho, and L. Schmidt-Thieme, "Tag-aware recommender systems by fusion of collaborative filtering algorithms," in *ACM symposium on Applied Computing*, pp. 1995–1999, ACM, 2008.

[19] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme, "Tag recommendations in folksonomies," *PKDD*, pp. 506–514, 2007.

[20] C. Marlow, M. Naaman, D. Boyd, and M. Davis, "Ht06, tagging paper, taxonomy, flickr, academic article, to read," in *Conf. on Hypertext and Hypermedia*, pp. 31–40, ACM, 2006.

[21] B. Markines, C. Cattuto, F. Menczer, D. Benz, A. Hotho, and G. Stumme, "Evaluating similarity measures for emergent semantics of social tagging," in *WWW*, pp. 641–641, 2009.

[22] I. Pilászy and D. Tikk, "Recommending new movies: even a few ratings are more valuable than metadata," in *RecSys*, pp. 93–100, ACM, 2009.

[23] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?," in *WWW*, pp. 591–600, ACM, 2010.

[24] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi, "Measuring user influence in Twitter: The million follower fallacy," in *ICWSM*, 2010.

[25] M. Cha, A. Mislove, B. Adams, and K. P. Gummadi, "Characterizing social cascades in flickr," in *Proc workshop on Online social networks*, pp. 13–18, ACM, 2008.

[26] E. Bakshy, J. M. H., W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on twitter," in *WSDM*, pp. 65–74, ACM, 2011.

Four Degrees of Separation

Lars Backstrom* Paolo Boldi† Marco Rosa† Johan Ugander* Sebastiano Vigna†

January 5, 2012

Abstract

Frigyes Karinthy, in his 1929 short story "Láncszemek" ("Chains") suggested that any two persons are distanced by at most six friendship links.[1] Stanley Milgram in his famous experiment [20, 23] challenged people to route postcards to a fixed recipient by passing them only through direct acquaintances. The average number of intermediaries on the path of the postcards lay between 4.4 and 5.7, depending on the sample of people chosen.

We report the results of the first world-scale social-network graph-distance computations, using the entire Facebook network of active users (≈ 721 million users, ≈ 69 billion friendship links). The average distance we observe is 4.74, corresponding to 3.74 intermediaries or "degrees of separation", showing that the world is even smaller than we expected, and prompting the title of this paper. More generally, we study the distance distribution of Facebook and of some interesting geographic subgraphs, looking also at their evolution over time.

The networks we are able to explore are almost two orders of magnitude larger than those analysed in the previous literature. We report detailed statistical metadata showing that our measurements (which rely on probabilistic algorithms) are very accurate.

1 Introduction

At the 20th World–Wide Web Conference, in Hyderabad, India, one of the authors (Sebastiano) presented a new tool for studying the distance distribution of very large graphs: HyperANF [3]. Building on previous graph compression [4] work and on the idea of diffusive computation pioneered in [21], the new tool made it possible to accurately study the distance distribution of graphs orders of magnitude larger than it was previously possible.

One of the goals in studying the distance distribution is the identification of interesting statistical parameters that can be used to tell proper social networks from other complex networks, such as web graphs. More generally, the distance distribution is one interesting *global* feature that makes it possible to reject probabilistic models even when they match local features such as the in-degree distribution.

In particular, earlier work had shown that the *spid*[2], which measures the *dispersion* of the distance distribution, appeared to be smaller than 1 (underdispersion) for social networks, but larger than one (overdispersion) for web graphs [3]. Hence, during the talk, one of the main open questions was "What is the spid of Facebook?".

Lars Backstrom happened to listen to the talk, and suggested a collaboration studying the Facebook graph. This was of course an extremely intriguing possibility: beside testing the "spid hypothesis", computing the distance distribution of the Facebook graph would have been the largest Milgram-like [20] experiment ever performed, orders of magnitudes larger than previous attempts (during our experiments Facebook has ≈ 721 million active users and ≈ 69 billion friendship links).

This paper reports our findings in studying the distance distribution of the largest electronic social network ever created. That world is smaller than we thought: the average distance of the current Facebook graph is 4.74. Moreover, the spid of the graph is just 0.09, corroborating the conjecture [3] that proper social networks have a spid well below one. We also observe, contrary to previous literature analysing graphs orders of magnitude smaller, both a stabilisation of the average distance over time, and that the density of the Facebook graph over time does not neatly fit previous models.

Towards a deeper understanding of the structure of the Facebook graph, we also apply recent compression techniques

*Facebook.

†DSI, Università degli Studi di Milano, Italy. Paolo Boldi, Marco Rosa and Sebastiano Vigna have been partially supported by a Yahoo! faculty grant and by MIUR PRIN "Query log e web crawling".

[1] The exact wording of the story is slightly ambiguous: "He bet us that, using no more than five individuals, one of whom is a personal acquaintance, he could contact the selected individual [...]". It is not completely clear whether the selected individual is part of the five, so this could actually allude to distance five or six in the language of graph theory, but the "six degrees of separation" phrase stuck after John Guare's 1990 eponymous play. Following Milgram's definition and Guare's interpretation (see further on), we will assume that "degrees of separation" is the same as "distance minus one", where "distance" is the usual path length (the number of arcs in the path).

[2] The spid (shortest-paths index of dispersion) is the variance-to-mean ratio of the distance distribution.

that exploit the underlying cluster structure of the graph to increase *locality*. The results obtained suggests the existence of overlapping clusters similar to those observed in other social networks.

Replicability of scientific results is important. While for obvious nondisclosure reasons we cannot release to the public the actual 30 graphs that have been studied in this paper, we distribute freely the derived data upon which the tables and figures of this papers have been built, that is, the WebGraph *properties*, which contain structural information about the graphs, and the probabilistic estimations of their neighbourhood functions (see below) that have been used to study their distance distributions. The software used in this paper is distributed under the (L)GPL General Public License.[3]

2 Related work

The most obvious precursor of our work is Milgram's celebrated "small world" experiment, described first in [20] and later with more details in [23]: Milgram's works were actually following a stream of research started in sociology and psychology in the late 50s [12]. In his experiment, Milgram aimed at answering the following question (in his words): "given two individuals selected randomly from the population, what is the probability that the minimum number of intermediaries required to link them is 0, 1, 2, ..., k?".

The technique Milgram used (inspired by [22]) was the following: he selected 296 volunteers (the *starting population*) and asked them to dispatch a message to a specific individual (the *target person*), a stockholder living in Sharon, MA, a suburb of Boston, and working in Boston. The message could not be sent directly to the target person (unless the sender knew him personally), but could only be mailed to a personal acquaintance who is more likely than the sender to know the target person. The starting population was selected as follows: 100 of them were people living in Boston, 100 were Nebraska stockholders (i.e., people living far from the target but sharing with him their profession) and 96 were Nebraska inhabitants chosen at random.

In a nutshell, the results obtained from Milgram's experiments were the following: only 64 chains (22%) were completed (i.e., they reached the target); the average number of intermediaries in these chains was 5.2, with a marked difference between the Boston group (4.4) and the rest of the starting population, whereas the difference between the two other subpopulations was not statistically significant; at the other end of the spectrum, the random (and essentially clueless) group from Nebraska needed 5.7 intermediaries on average (i.e., rounding up, "six degrees of separation"). The main conclusions outlined in Milgram's paper were that the average path length is small, much smaller than expected, and that geographic location seems to have an impact on the average length whereas other information (e.g., profession) does not.

There is of course a fundamental difference between our experiment and what Milgram did: Milgram was measuring the average length of a *routing path* on a social network, which is of course an upper bound on the average distance (as the people involved in the experiment were not necessarily sending the postcard to an acquaintance on a shortest path to the destination).[4] In a sense, the results he obtained are even more striking, because not only do they prove that the world is small, but that the actors living in the small world are able to exploit its smallness. It should be remarked, however, that in [20, 23] the purpose of the authors is to estimate the number of intermediaries: the postcards are just a tool, and the details of the paths they follow are studied only as an artifact of the measurement process. The interest in efficient routing lies more in the eye of the beholder (e.g., the computer scientist) than in Milgram's: with at his disposal an actual large database of friendship links and algorithms like the ones we use, he would have dispensed with the postcards altogether.

Incidentally, there have been some attempts to reproduce Milgram-like routing experiments on various large networks [18, 14, 11], but the results in this direction are still very preliminary because notions such as identity, knowledge or routing are still poorly understood in social networks.

We limited ourselves to the part of Milgram's experiment that is more clearly defined, that is, the measurement of shortest paths. The largest experiment similar to the ones presented here that we are aware of is [15], where the authors considered a *communication graph* with 180 million nodes and 1.3 billion edges extracted from a snapshot of the Microsoft Messenger network; they find an average distance of 6.6 (i.e., 5.6 intermediaries; again, rounding up, six degrees of separation). Note, however, that the communication graph in [15] has an edge between two persons only if they communicated during a specific one-month observation period, and thus does not take into account friendship links through which no communication was detected.

The authors of [24], instead, study the distance distribution of some small-sized social networks. In both cases the networks were undirected and small enough (by at least two orders of magnitude) to be accessed efficiently in a random fashion, so the authors used *sampling* techniques. We remark, however, that sampling is not easily applicable to di-

[3] See `http://{webgraph,law}.dsi.unimi.it/`.

[4] Incidentally, this observation is at the basis of one of the most intense monologues in Guare's play: Ouisa, unable to locate Paul, the con man who convinced them he is the son of Sidney Poitier, says "I read somewhere that everybody on this planet is separated by only six other people. Six degrees of separation. Between us and everybody else on this planet. [...] But to find the right six people." Note that this fragment of the monologue clearly shows that Guare's interpretation of the "six degree of separation" idea is equivalent to distance *seven* in the graph-theoretical sense.

rected networks (such as Twitter) that are not strongly connected, whereas our techniques would still work (for some details about the applicability of sampling, see [8]).

Analysing the evolution of social networks in time is also a lively trend of research. Leskovec, Kleinberg and Faloutsos observe in [16] that the average degree of complex networks increase over time while the *effective diameter* shrinks. Their experiments are conducted on a much smaller scale (their largest graph has 4 millions of nodes and 16 millions of arcs), but it is interesting that the phenomena observed seems quite consistent. Probably the most controversial point is the hypothesis that the number of edges $m(t)$ at time t is related to the number of nodes $n(t)$ by the following relation:

$$m(t) \propto n(t)^a,$$

where a is a fixed exponent usually lying in the interval $(1\,..\,2)$. We will discuss this hypothesis in light of our findings.

3 Definitions and Tools

The *neighbourhood function* $N_G(t)$ of a graph G returns for each $t \in \mathbf{N}$ the number of pairs of nodes $\langle x, y\rangle$ such that y is reachable from x in at most t steps. It provides data about how fast the "average ball" around each node expands. From the neighbourhood function it is possible to derive the distance distribution (between reachable pairs), which gives for each t the fraction of reachable pairs at distance exactly t.

In this paper we use HyperANF, a diffusion-based algorithm (building on ANF [21]) that is able to approximate quickly the neighbourhood function of very large graphs; our implementation uses, in turn, WebGraph [4] to represent in a compressed but quickly accessible form the graphs to be analysed.

HyperANF is based on the observation (made in [21]) that $B(x, r)$, the ball of radius r around node x, satisfies

$$B(x, r) = \bigcup_{x \to y} B(y, r-1) \cup \{\, x \,\}.$$

Since $B(x, 0) = \{\, x \,\}$, we can compute each $B(x, r)$ incrementally using sequential scans of the graph (i.e., scans in which we go in turn through the successor list of each node). The obvious problem is that during the scan we need to access randomly the sets $B(x, r-1)$ (the sets $B(x, r)$ can be just saved on disk on a *update file* and reloaded later).

The space needed for such sets would be too large to be kept in main memory. However, HyperANF represents these sets in an *approximate* way, using *HyperLogLog counters* [10], which should be thought as dictionaries that can answer reliably just questions about size. Each such counter is made of a number of small (in our case, 5-bit) *registers*. In a nutshell, a register keeps track of the maximum number M of trailing zeroes of the values of a good hash function applied to the elements of a sequence of nodes: the number of distinct elements in the sequence is then proportional to 2^M. A technique called *stochastic averaging* is used to divide the stream into a number of substreams, each analysed by a different register. The result is then computed by aggregating suitably the estimation from each register (see [10] for details).

The main performance challenge to solve is how to quickly compute the HyperLogLog counter associated to a union of balls, each represented, in turn, by a HyperLogLog counter: HyperANF uses an algorithm based on word-level parallelism that makes the computation very fast, and a carefully engineered implementation exploits multicore architectures with a linear speedup in the number of cores.

Another important feature of HyperANF is that it uses a *systolic* approach to avoid recomputing balls that do not change during an iteration. This approach is fundamental to be able to compute the entire distance distribution, avoiding the arbitrary termination conditions used by previous approaches, which have no provable accuracy (see [3] for an example).

3.1 Theoretical error bounds

The result of a run of HyperANF at the t-th iteration is an estimation of the neighbourhood function in t. We can see it as a random variable

$$\hat{N}_G(t) = \sum_{0 \le i < n} X_{i,t}$$

where each $X_{i,t}$ is the HyperLogLog counter that counts nodes reached by node i in t steps (n is the number of nodes of the graph). When m registers per counter are used, each $X_{i,t}$ has a guaranteed relative standard deviation $\eta_m \le 1.06/\sqrt{m}$.

It is shown in [3] that the output $\hat{N}_G(t)$ of HyperANF at the t-th iteration is an asymptotically almost unbiased estimator of $N_G(t)$, that is

$$\frac{E[\hat{N}_G(t)]}{N_G(t)} = 1 + \delta_1(n) + o(1) \text{ for } n \to \infty,$$

where δ_1 is the same as in [10][Theorem 1] (and $|\delta_1(x)| < 5 \cdot 10^{-5}$ as soon as $m \ge 16$). Moreover, $\hat{N}_G(t)$ has a relative standard deviation not greater than that of the X_i's, that is

$$\frac{\sqrt{\mathrm{Var}[\hat{N}_G(t)]}}{N_G(t)} \le \eta_m.$$

In particular, our runs used $m = 64$ ($\eta_m = 0.1325$) for all graphs except for the two largest Facebook graphs, where we

used $m = 32$ ($\eta_m = 0.187$). Runs were repeated so to obtain a uniform relative standard deviation for all graphs.

Unfortunately, the relative error for the neighbourhood function becomes an *absolute* error for the distance distribution. Thus, the theoretical bounds one obtains for the moments of the distance distribution are quite ugly. Actually, the simple act of dividing the neighbourhood function values by the last value to obtain the cumulative distribution function is nonlinear, and introduces bias in the estimation.

To reduce bias and provide estimates of the standard error of our measurements, we use the *jackknife* [9], a classical nonparametric method for evaluating arbitrary statistics on a data sample, which turns out to be very effective in practice [3].

4 Experiments

The graphs analysed in this paper are graphs of Facebook users who were active in May of 2011; an active user is one who has logged in within the last 28 days. The decision to restrict our study to active users allows us to eliminate accounts that have been abandoned in early stages of creation, and focus on accounts that plausibly represent actual individuals. In accordance with Facebook's data retention policies, historical user activity records are not retained, and historical graphs for each year were constructed by considering currently active users that were registered on January 1st of that year, along with those friendship edges that were formed prior that that date. The "current" graph is simply the graph of active users at the time when the experiments were performed (May 2011). The graph predates the existence of Facebook "subscriptions", a directed relationship feature introduced in August 2011, and also does not include "pages" (such as celebrities) that people may "like". For standard user accounts on Facebook there is a limit of 5 000 possible friends.

We decided to extend our experiments in two directions: regional and temporal. We thus analyse the entire Facebook graph (`fb`), the USA subgraph (`us`), the Italian subgraph (`it`) and the Swedish (`se`) subgraph. We also analysed a combination of the Italian and Swedish graph (`itse`) to check whether combining two regional but distant networks could significantly change the average distance, in the same spirit as in the original Milgram's experiment.[5] For each graph we compute the distance distribution from 2007 up to today by performing several HyperANF runs, obtaining an estimate of values of neighbourhood function with relative standard deviation at most 5.8%: in several cases, however, we performed more runs, obtaining a higher precision. We report the jackknife [9] estimate of derived values (such as average distances) and the associated estimation of the standard error.

4.1 Setup

The computations were performed on a 24-core machine with 72 GiB of memory and 1 TiB of disk space.[6] The first task was to import the Facebook graph(s) into a compressed form for WebGraph [4], so that the multiple scans required by HyperANF's diffusive process could be carried out relatively quickly. This part required some massaging of Facebook's internal IDs into a contiguous numbering: the resulting current `fb` graph (the largest we analysed) was compressed to 345 GB at 20 bits per arc, which is 86% of the information-theoretical lower bound ($\log \binom{n^2}{m}$ bits, there n is the number of nodes and m the number of arcs).[7] Whichever coding we choose, for half of the possible graphs with n nodes and m arcs we need at least $\lfloor \log \binom{n^2}{m} \rfloor$ bits per graph: the purpose of compression is precisely to choose the coding so to represent interesting graphs in a smaller space than that required by the bound.

To understand what is happening, we recall that WebGraph uses the BV compression scheme [4], which applies three intertwined techniques to the successor list of a node:

- successors are (partially) *copied* from previous nodes within a small window, if successors lists are similar enough;
- successors are *intervalised*, that is, represented by a left extreme and a length, if significant contiguous successor sequences appear;
- successors are *gap-compressed* if they pass the previous phases: instead of storing the actual successor list, we store the differences of consecutive successors (in increasing order) using instantaneous codes.

Thus, a graph compresses well when it exhibits *similarity* (nodes with near indices have similar successor lists) and *locality* (successor lists have small gaps).

The better-than-random result above (usually, randomly permuted graphs compressed with WebGraph occupy 10 – 20% more space than the lower bound) has most likely been induced by the renumbering process, as in the original stream of arcs all arcs going out from a node appeared consecutively;

[5] To establish geographic location, we use the users' *current* geo-IP location; this means, for example, that the users in the it-2007 graph are users who are today in Italy and were on Facebook on January 1, 2007 (most probably, American college students then living in Italy).

[6] We remark that the commercial value of such hardware is of the order of a few thousand dollars.

[7] Note that we measure compression with respect to the lower bound on *arcs*, as WebGraph stores *directed* graphs; however, with the additional knowledge that the graph is undirected, the lower bound should be applied to *edges*, thus doubling, in practice, the number of bits used.

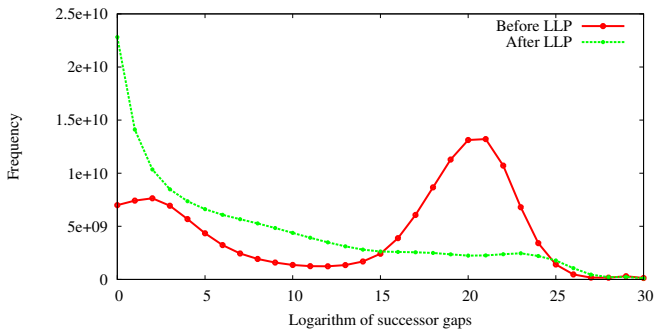


Figure 1: The change in distribution of the logarithm of the gaps between successors when the current `fb` graph is permuted by layered label propagation. See also Table 1.

as a consequence, the renumbering process assigned consecutive labels to all yet-unseen successors (e.g., in the initial stages successors were labelled contiguously), inducing some locality.

It is also possible that the "natural" order for Facebook (essentially, join order) gives rise to some improvement over the information-theoretical lower bound because users often join the network at around the same time as several of their friends, which causes a certain amount of locality and similarity, as circle of friends have several friends in common.

We were interested in the first place to establish whether more locality could be induced by suitably permuting the graph using *layered labelled propagation* [2] (LLP). This approach (which computes several clusterings with different levels of granularity and combines them to sort the nodes of a graph so to increase its locality and similarity) has recently led to the best compression ratios for social networks when combined with the BV compression scheme. An increase in compression means that we were able to partly understand the cluster structure of the graph.

We remark that each of the clusterings required by LLP is in itself a *tour de force*, as the graphs we analyse are almost two orders of magnitude larger than any network used for experiments in the literature on graph clustering. Indeed, applying LLP to the current Facebook graph required ten days of computation on our hardware.

We applied layered labelled propagation and re-compressed our graphs (the current version), obtaining a significant improvement. In Table 1 we show the results: we were able to reduce the graph size by 30%, which suggests that LLP has been able to discover several significant clusters.

The change in structure can be easily seen from Figure 1, where we show the distribution of the binary logarithm of gaps between successors for the current `fb` graph. The smaller the gaps, the higher the locality. In the graph with renumbered Facebook IDs, the distribution is bimodal: there is a local maximum at two, showing that there is some locality, but the bulk of the probability mass is around 20–21, which is slightly less than the information-theoretical lower bound (≈ 23).

In the graph permuted with LLP, however, the distribution radically changes: it is now (mostly) beautifully monotonically decreasing, with a very small bump at 23, which testifies the existence of a small core of "randomness" in the graph that LLP was not able to tame.

Regarding similarity, we see an analogous phenomenon: the number of successors represented by copy has doubled, going from 9% to 18%. The last datum is in line with other social networks (web graphs, on the contrary, are extremely redundant and more than 80% of the successors are usually copied). Moreover, disabling copying altogether results in modest increase in size ($\approx 5\%$), again in line with other social networks, which suggests that for most applications it is better to disable copying at all to obtain faster random access.

The compression ratio is around 53%, which is similar to other similar social networks, such as LiveJournal (55%) or DBLP (40%) [2][8]. For other graphs (see Table 1), however, it is slightly worse. This might be due to several phenomena: First, our LLP runs were executed with only half the number or clusters, and for each cluster we restricted the number of iterations to just four, to make the whole execution of LLP feasible. Thus, our runs are capable of finding considerably less structure than the runs we had previously performed for other networks. Second, the number of nodes is much larger: there is some cost in writing down gaps (e.g., using γ, δ or ζ codes) that is dependent on their absolute magnitude, and the lower bound does not take into account that cost.

4.2 Running

Since most of the graphs, because of their size, had to be accessed by memory mapping, we decided to store all counters (both those for $B(x, r-1)$ and those for $B(x, r)$) in main memory, to avoid eccessive I/O. The runs of HyperANF on the current whole Facebook graph used 32 registers, so the space for counters was about 27 GiB (e.g., we could have analysed a graph with four times the number of nodes on the same hardware). As a rough measure of speed, a run on the LLP-compressed current whole Facebook graph requires about 13.5 hours. Note that this timings would scale linearly with an increase in the number of cores.

4.3 General comments

In September 2006, Facebook was opened to non-college students: there was an instant surge in subscriptions, as our

[8]The interested reader will find similar data for several type of networks at the LAW web site (`http://law.dsi.unimi.it/`).

	it	se	itse	us	fb
Original	14.8 (83%)	14.0 (86%)	15.0 (82%)	17.2 (82%)	20.1 (86%)
LLP	10.3 (58%)	10.2 (63%)	10.3 (56%)	11.6 (56%)	12.3 (53%)

Table 1: The number of bits per link and the compression ratio (with respect to the information-theoretical lower bound) for the current graphs in the original order and for the same graphs permuted by layered label propagation [2].

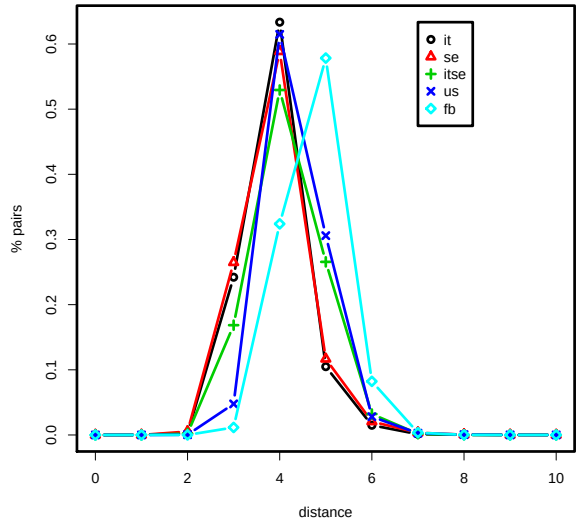


Figure 2: The probability mass functions of the distance distributions of the current graphs (truncated at distance 10).

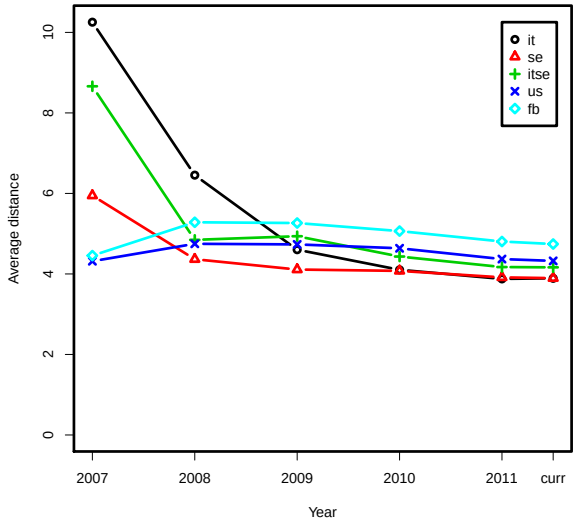


Figure 3: The average distance graph. See also Table 6.

data shows. In particular, the `it` and `se` subgraphs from January 1, 2007 were highly disconnected, as shown by the incredibly low percentage of reachable pairs we estimate in Table 9. Even Facebook itself was rather disconnected, but all the data we compute stabilizes (with small oscillations) after 2009, with essentially all pairs reachable. Thus, we consider the data for 2007 and 2008 useful to observe the evolution of Facebook, but we do not consider them representative of the underlying human social-link structure.

	it	se	itse	us	fb
2007	1.31	3.90	1.50	119.61	99.50
2008	5.88	46.09	36.00	106.05	76.15
2009	50.82	69.60	55.91	111.78	88.68
2010	122.92	100.85	118.54	128.95	113.00
2011	198.20	140.55	187.48	188.30	169.03
current	226.03	154.54	213.30	213.76	190.44

Table 4: Average degree of the datasets.

	it	se	itse	us	fb
2007	0.04	10.23	0.19	100.00	68.02
2008	25.54	93.90	80.21	99.26	89.04

Table 9: Percentage of reachable pairs 2007–2008.

4.4 The distribution

Figure 2 displays the probability mass functions of the current graphs. We will discuss later the variation of the average distance and spid, but qualitatively we can immediately distinguish the *regional* graphs, concentrated around distance four, and the *whole* Facebook graph, concentrated around distance five. The distributions of `it` and `se`, moreover, have significantly less probability mass concentrated on distance five than `itse` and `us`. The variance data (Table 7 and Figure 4) show that the distribution became quickly extremely concentrated.

	it	se	itse	us	fb
2007	159.8 K (105.0 K)	11.2 K (21.8 K)	172.1 K (128.8 K)	8.8 M (529.3 M)	13.0 M (644.6 M)
2008	335.8 K (987.9 K)	1.0 M (23.2 M)	1.4 M (24.3 M)	20.1 M (1.1 G)	56.0 M (2.1 G)
2009	4.6 M (116.0 M)	1.6 M (55.5 M)	6.2 M (172.1 M)	41.5 M (2.3 G)	139.1 M (6.2 G)
2010	11.8 M (726.9 M)	3.0 M (149.9 M)	14.8 M (878.4 M)	92.4 M (6.0 G)	332.3 M (18.8 G)
2011	17.1 M (1.7 G)	4.0 M (278.2 M)	21.1 M (2.0 G)	131.4 M (12.4 G)	562.4 M (47.5 G)
current	19.8 M (2.2 G)	4.3 M (335.7 M)	24.1 M (2.6 G)	149.1 M (15.9 G)	721.1 M (68.7 G)

Table 2: Number of nodes and friendship links of the datasets. Note that each friendship link, being undirected, is represented by a pair of symmetric arcs.

	it	se	itse	us	fb
2007	387.0 K	51.0 K	461.9 K	1.8 G	2.3 G
2008	3.9 M	96.7 M	107.8 M	4.0 G	9.2 G
2009	477.9 M	227.5 M	840.3 M	9.1 G	28.7 G
2010	3.6 G	623.0 M	4.5 G	26.0 G	93.3 G
2011	8.0 G	1.1 G	9.6 G	53.6 G	238.1 G
current	8.3 G	1.2 G	9.7 G	68.5 G	344.9 G

Table 3: Size in bytes of the datasets.

Lower bounds from HyperANF runs

	it	se	itse	us	fb
2007	41	17	41	13	14
2008	28	17	24	17	16
2009	21	16	17	16	15
2010	18	19	19	19	15
2011	17	20	17	18	35
current	19	19	19	20	58
Exact diameter of the giant component					
current	25	23	27	30	41

Table 10: Lower bounds for the diameter of all graphs, and exact values for the giant component ($> 99.7\%$) of current graphs computed using the iFUB algorithm.

4.5 Average degree and density

Table 4 shows the relatively quick growth in time of the average degree of all graphs we consider. The more users join the network, the more existing friendship links are uncovered. In Figure 6 we show a loglog-scaled plot of the same data: with the small set of points at our disposal, it is difficult to draw reliable conclusions, but we are not always observing the power-law behaviour suggested in [16]: see, for instance, the change of the slope for the `us` graph.[9]

The *density* of the network, on the contrary, decreases.[10] In Figure 5 we plot the density (number of edges divided by number of nodes) of the graphs against the number of nodes (see also Table 5). There is some initial alternating behaviour, but on the more complete networks (`fb` and `us`) the trend in sparsification is very evident.

Geographical concentration, however, increases density: in Figure 5 we can see the lines corresponding to our regional graphs clearly ordered by geographical concentration, with the `fb` graph in the lowest position.

4.6 Average distance

The results concerning average distance[11] are displayed in Figure 3 and Table 6. The average distance[12] on the Face-

[9] We remind the reader that on a log-log plot almost anything "looks like" a straight line. The quite illuminating examples shown in [17], in particular, show that goodness-of-fit tests are essential.

[10] We remark that the authors of [16] call *densification* the increase of the average degree, in contrast with established literature in graph theory, where *density* is the fraction of edges with respect to all possible edges (e.g., $2m/(n(n-1))$). We use "density", "densification" and "sparsification" in the standard sense.

[11] The data we report is about the average distance *between reachable pairs*, for which the name *average connected distance* has been proposed [5]. This is the same measure as that used by Travers and Milgram in [23]. We refrain from using the word "connected" as it somehow implies a bidirectional (or, if you prefer, undirected) connection. The notion of average distance between all pairs is useless in a graph in which not all pairs are reachable, as it is necessarily infinite, so no confusion can arise.

[12] In some previous literature (e.g., [16]), the 90% percentile (possibly with some interpolation) of the distance distribution, called *effective diameter*, has been used in place of the average distance. Having at our disposal tools that can compute easily the average distance, which is a parameterless, standard feature of the distance distribution that

	it	se	itse	us	fb
2007	8.224E-06	3.496E-04	8.692E-06	1.352E-05	7.679E-06
2008	1.752E-05	4.586E-05	2.666E-05	5.268E-06	1.359E-06
2009	1.113E-05	4.362E-05	9.079E-06	2.691E-06	6.377E-07
2010	1.039E-05	3.392E-05	7.998E-06	1.395E-06	3.400E-07
2011	1.157E-05	3.551E-05	8.882E-06	1.433E-06	3.006E-07
current	1.143E-05	3.557E-05	8.834E-06	1.434E-06	2.641E-07

Table 5: Density of the datasets.

	it	se	itse	us	fb
2007	10.25 (±0.17)	5.95 (±0.07)	8.66 (±0.14)	4.32 (±0.02)	4.46 (±0.04)
2008	6.45 (±0.03)	4.37 (±0.03)	4.85 (±0.05)	4.75 (±0.02)	5.28 (±0.03)
2009	4.60 (±0.02)	4.11 (±0.01)	4.94 (±0.02)	4.73 (±0.02)	5.26 (±0.03)
2010	4.10 (±0.02)	4.08 (±0.02)	4.43 (±0.03)	4.64 (±0.02)	5.06 (±0.01)
2011	3.88 (±0.01)	3.91 (±0.01)	4.17 (±0.02)	4.37 (±0.01)	4.81 (±0.04)
current	3.89 (±0.02)	3.90 (±0.04)	4.16 (±0.01)	4.32 (±0.01)	4.74 (±0.02)

Table 6: The average distance (± standard error). See also Figure 3 and 7.

book current graph is 4.74.[13] Moreover, a closer look at the distribution shows that 92% of the reachable pairs of individuals are at distance five or less.

We note that both on the `it` and `se` graphs we find a significantly lower, but similar value. We interpret this result as telling us that the average distance is actually dependent on the geographical closeness of users, more than on the actual size of the network. This is confirmed by the higher average distance of the `itse` graph.

During the fastest growing years of Facebook our graphs show a quick decrease in the average distance, which however appears now to be stabilizing. This is not surprising, as "shrinking diameter" phenomena are always observed when a large network is "uncovered", in the sense that we look at larger and larger induced subgraphs of the underlying global human network. At the same time, as we already remarked, density was going down steadily. We thus see the small-world phenomenon fully at work: a smaller fraction of arcs connecting the users, but nonetheless a lower average distance.

To make more concrete the "degree of separation" idea, in Table 11 we show the percentage of reachable pairs *within the ceiling of the average distance* (note, again, that it is the percentage relatively to the reachable pairs): for instance, in the current Facebook graph 92% of the pairs of reachable users are within distance five—four degrees of separation.

has been used in social sciences for decades, we prefer to stick to it. Experimentally, on web and social graphs the average distance is about two thirds of the effective diameter plus one [3].

[13]Note that both Karinthy and Guare had in mind the *maximum*, not the *average* number of degrees, so they were actually upper bounding the diameter.

4.7 Spid

The *spid* is the *index of dispersion* σ^2/μ (a.k.a. *variance-to-mean ratio*) of the distance distribution. Some of the authors proposed the spid [3] as a measure of the "webbiness" of a social network. In particular, networks with a spid larger than one should be considered "web-like", whereas networks with a spid smaller than one should be considered "properly social". We recall that a distribution is called under- or over-dispersed depending on whether its index of dispersion is smaller or larger than 1 (e.g., variance smaller or larger than the average distance), so a network is considered properly social or not depending on whether its distance distribution is under- or over-dispersed.

The intuition behind the spid is that "properly social" networks strongly favour short connections, whereas in the web long connection are not uncommon. As we recalled in the introduction, the starting point of the paper was the question "What is the spid of Facebook"? The answer, confirming the data we gathered on different social networks in [3], is shown in Table 8. With the exception of the highly disconnected regional networks in 2007–2008 (see Table 9), the spid is well below one.

Interestingly, across our collection of graphs we can confirm that there is in general little correlation between the average distance and the spid: Kendall's τ is -0.0105; graphical evidence of this fact can be seen in the scatter plot shown in Figure 7.

If we consider points associated with a single network, though, there appears to be some correlation between average distance and spid, in particular in the more connected

	it	se	itse	us	fb
2007	32.46 (±1.49)	3.90 (±0.12)	16.62 (±0.87)	0.52 (±0.01)	0.65 (±0.02)
2008	3.78 (±0.18)	0.69 (±0.04)	1.74 (±0.15)	0.82 (±0.02)	0.86 (±0.03)
2009	0.64 (±0.04)	0.56 (±0.02)	0.84 (±0.02)	0.62 (±0.02)	0.69 (±0.05)
2010	0.40 (±0.01)	0.50 (±0.02)	0.64 (±0.03)	0.53 (±0.02)	0.52 (±0.01)
2011	0.38 (±0.03)	0.50 (±0.02)	0.61 (±0.02)	0.39 (±0.01)	0.42 (±0.03)
current	0.42 (±0.03)	0.52 (±0.04)	0.57 (±0.01)	0.40 (±0.01)	0.41 (±0.01)

Table 7: The variance of the distance distribution (± standard error). See also Figure 4.

	it	se	itse	us	fb
2007	3.17 (±0.106)	0.66 (±0.016)	1.92 (±0.078)	0.12 (±0.003)	0.15 (±0.004)
2008	0.59 (±0.026)	0.16 (±0.008)	0.36 (±0.028)	0.17 (±0.003)	0.16 (±0.005)
2009	0.14 (±0.007)	0.14 (±0.004)	0.17 (±0.004)	0.13 (±0.003)	0.13 (±0.009)
2010	0.10 (±0.003)	0.12 (±0.005)	0.14 (±0.006)	0.11 (±0.004)	0.10 (±0.002)
2011	0.10 (±0.006)	0.13 (±0.006)	0.15 (±0.004)	0.09 (±0.003)	0.09 (±0.005)
current	0.11 (±0.007)	0.13 (±0.010)	0.14 (±0.003)	0.09 (±0.003)	0.09 (±0.003)

Table 8: The index of dispersion of distances, a.k.a. spid (± standard error). See also Figure 7.

networks (the values for Kendall's τ are all above 0.6, except for `se`). However, this is just an artifact, as the correlation between spid and average distance is *inverse* (larger average distance, smaller spid). What is happening is that in this case the variance (see Table 7) is changing in the same direction: smaller average distances (which would imply a larger spid) are associated with smaller variances. Figure 8 displays the mild correlation between average distance and variance in the graphs we analyse: as a network gets tighter, its distance distribution also gets more concentrated.

4.8 Diameter

HyperANF cannot provide exact results about the diameter: however, the number of steps of a run is necessarily a lower bound for the diameter of the graph (the set of registers can stabilize before a number of iterations equal to the diameter because of hash collisions, but never after). While there are no statistical guarantees on this datum, in Table 10 we report these maximal observations as lower bounds that differ significantly between regional graphs and the overall Facebook graph—there are people that are significantly more "far apart" in the world than in a single nation.[14]

To corroborate this information, we decided to also approach the problem of computing the exact diameter directly, although it is in general a daunting task: for very large graphs matrix-based algorithms are simply not feasible in space, and the basic algorithm running n breadth-first visits is not feasible in time. We thus implemented a highly parallel version of the iFUB (iterative Fringe Upper Bound) algorithm introduced in [6] (extending the ideas of [7, 19]) for undirected graphs.

The basic idea is as follows: consider some node x, and find (by a breadth-first visit) a node y farthest from x. Find now a node z farthest from y: $d(y, z)$ is a (usually very good) lower bound on the diameter, and actually it *is* the diameter if the graph is a tree (this is the "double sweep" algorithm).

We now consider a node c halfway between y and z: such a node is "in the middle of the graph" (actually, it would be a *center* if the graph was a tree), so if h is the eccentricy of c (the distance of the farthest node from c) we expect $2h$ to be a good upper bound for the diameter.

If our upper and lower bound match, we are finished. Otherwise, we consider the *fringe*: the nodes at distance exactly h from c. Clearly, if M is the maximum of the eccentricities of the nodes in the fringe, $\max\{\, 2(h-1), M \,\}$ is a new (and hopefully improved) upper bound, and M is a new (and hopefully improved) lower bound. We then iterate the process by examining fringes closer to the root until the bounds match.

Our implementation uses a multicore breadth-first visit: the queue of nodes at distance d is segmented into small blocks handled by each core. At the end of a round, we have computed the queue of nodes at distance $d + 1$. Our implementation was able to discover the diameter of the current `us` graph (which fits into main memory, thanks to LLP compression) in about twenty minutes. The diameter of Facebook required ten hours of computation of a machine with 1TiB of RAM (actually, 256GiB would have been sufficient, always because of LLP compression).

[14]Incidentally, as we already remarked, this is the measure that Karinthy and Guare actually had in mind.

	it	se	itse	us	fb
2007	65% (11)	64% (6)	67% (9)	95% (5)	91% (5)
2008	77% (7)	93% (5)	77% (5)	83% (5)	91% (6)
2009	90% (5)	96% (5)	75% (5)	86% (5)	94% (6)
2010	98% (5)	97% (5)	91% (5)	91% (5)	97% (6)
2011	90% (4)	86% (4)	95% (5)	97% (5)	89% (5)
current	88% (4)	86% (4)	97% (5)	97% (5)	91% (5)

Table 11: Percentage of reachable pairs within the ceiling of the average distance (shown between parentheses).

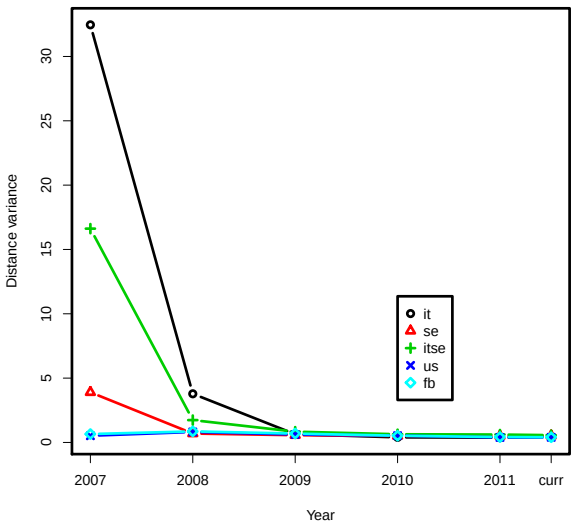


Figure 4: The graph of variances of the distance distributions. See also Table 7.

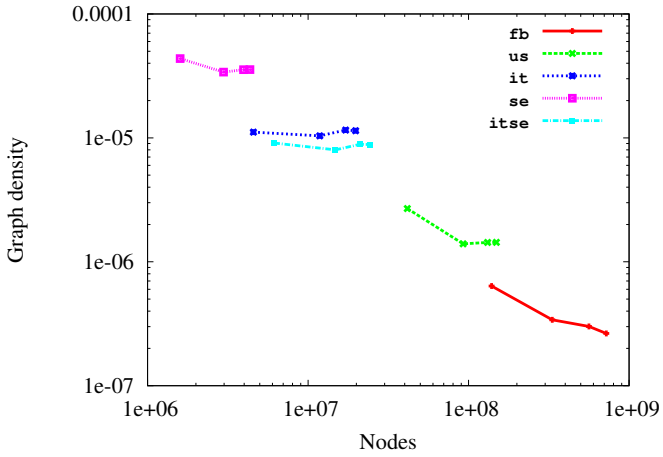


Figure 5: A plot correlating number of nodes to graph density (for the graph from 2009 on).

The values reported in Table 10 confirm what we discovered using the approximate data provided by the length of HyperANF runs, and suggest that while the distribution has a low average distance and it is quite concentrated, there are nonetheless (rare) pairs of nodes that are much farther apart. We remark that in the case of the current `fb` graph, the diameter of the giant component is actually *smaller* than the bound provided by the HyperANF runs, which means that long paths appear in small (and likely very irregular) components.

4.9 Precision

As already discussed in [3], it is very difficult to obtain strong theoretical bounds on data derived from the distance distribution. The problem is that when passing from the neighbourhood function to the distance distribution, the relative error bound becomes an *absolute* error bound: since the dis-

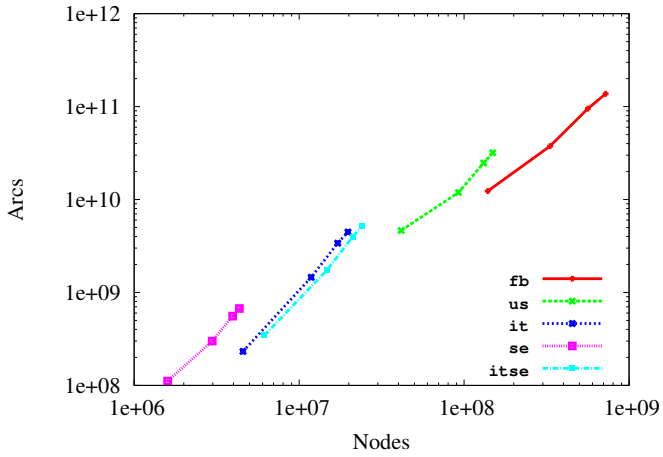


Figure 6: A plot correlating number of nodes to the average degree (for the graphs from 2009 on).

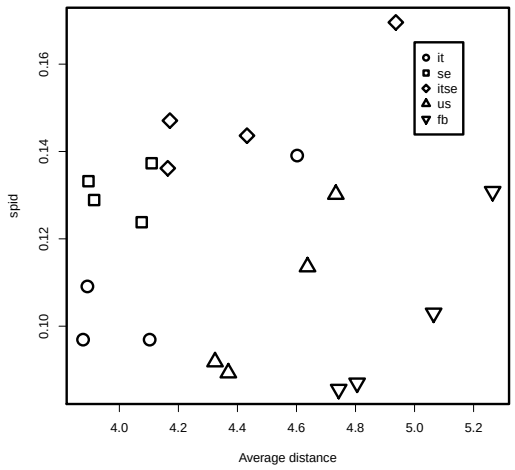


Figure 7: A scatter plot showing the (lack of) correlation between the average distance and the spid.

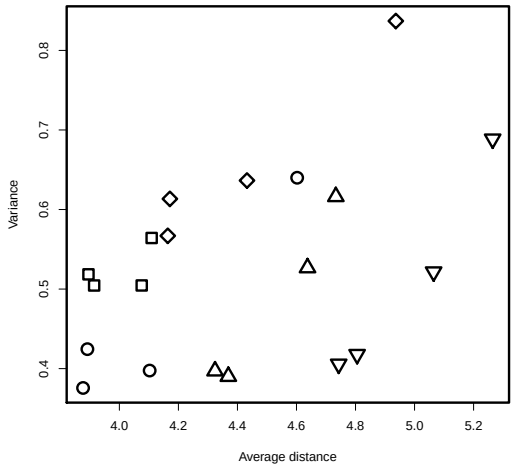


Figure 8: A scatter plot showing the mild correlation between the average distance and the variance.

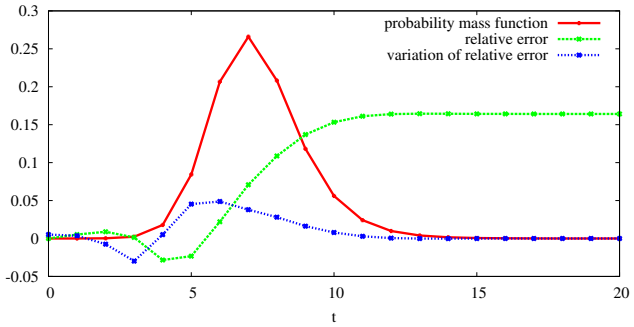


Figure 9: The evolution of the relative error in a HyperANF computation with relative standard deviation 9.25% on a small social network (`dblp-2010`).

tance distribution attains very small values (in particular in its tail), there is a concrete risk of incurring significant errors when computing the average distance or other statistics. On the other hand, the distribution of derived data is extremely concentrated [3].

There is, however, a clear empirical explanation of the unexpected accuracy of our results that is evident from an analysis of the evolution of the empirical relative error of a run on a social network. We show an example in Figure 9.

- In the very first steps, all counters contain essentially disjoint sets; thus, they behave as *independent random variables*, and under this assumption their relative error should be significantly smaller than expected: indeed, this is clearly visible from Figure 9.
- In the following few steps, the distribution reaches its highest value. The error oscillates, as counters are now significantly dependent from one another, but in this part the *actual value of the distribution is rather large*, so the absolute theoretical error turns out to be rather good.
- Finally, in the tail each counter contains a very large subset of the reachable nodes: as a result, all counters behave in a similar manner (as the hash collisions are essentially the same for every counter), and the relative error stabilises to an almost fixed value. Because of this stabilisation, *the relative error on the neighbourhood function transfers, in practice, to a relative error on the distance distribution.* To see why this happen, observe the behaviour of the *variation* of the relative error, which is quite erratic initially, but then converges quickly to zero. The variation is the only part of the relative error that becomes an absolute error when passing to the distance distribution, so the computation on the tail is much more accurate than what the theoretical bound would imply.

We remark that our considerations remain valid for any diffusion-based algorithm using approximate, statistically dependent counters (e.g., ANF [21]).

5 Conclusions

In this paper we have studied the largest electronic social network ever created (≈ 721 million active Facebook users and their ≈ 69 billion friendship links) from several viewpoints.

First of all, we have confirmed that layered labelled propagation [2] is a powerful paradigm for increasing locality of a social network by permuting its nodes. We have been able to compress the `us` graph at 11.6 bits per link—56% of the information-theoretical lower bound, similarly to other, much smaller social networks.

We then analysed using HyperANF the complete Facebook graph and 29 other graphs obtained by restricting geographically or temporally the links involved. We have in fact carried out the largest Milgram-like experiment ever performed. The average distance of Facebook is 4.74, that is, 3.74 "degrees of separation", prompting the title of this paper. The spid of Facebook is 0.09, well below one, as expected for a social network. Geographically restricted networks have a smaller average distance, as it happened in Milgram's original experiment. Overall, these results help paint the picture of what the Facebook social graph looks like. As expected, it is a small-world graph, with short paths between many pairs of nodes. However, the high degree of compressibility and the study of geographically limited subgraphs show that geography plays a huge role in forming the overall structure of network. Indeed, we see in this study, as well as other studies of Facebook [1] that, while the world is connected enough for short paths to exist between most nodes, there is a high degree of locality induced by various externalities, geography chief amongst them, all reminiscent of the model proposed in [13].

When Milgram first published his results, he in fact offered two opposing interpretations of what "six degrees of separation" actually meant. On the one hand, he observed that such a distance is considerably smaller than what one would naturally intuit. But at the same time, Milgram noted that this result could also be interpreted to mean that people are on average six "worlds apart": "When we speak of five[15] intermediaries, we are talking about an enormous psychological distance between the starting and target points, a distance which seems small only because we customarily regard 'five' as a small manageable quantity. We should think of the two points as being not five persons apart, but 'five circles of acquaintances' apart—five 'structures' apart." [20]. From this gloomier perspective, it is reassuring to see that our findings show that people are in fact only four world apart, and not six: when considering another person in the world, a friend of your friend knows a friend of their friend, on average.

[15] Five is the median of the number of intermediaries reported in the first paper by Milgram [20], from which our quotation is taken. More experiments were performed with Travers [23] with a slightly greater average, as reported in Section 2.

References

[1] Lars Backstrom, Eric Sun, and Cameron Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. In *Proceedings of the 19th international conference on World wide web*, pages 61–70. ACM, 2010.

[2] Paolo Boldi, Marco Rosa, Massimo Santini, and Sebastiano Vigna. Layered label propagation: A multiresolution coordinate-free ordering for compressing social networks. In Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar, M. P. Ravindra, Elisa Bertino, and Ravi Kumar, editors, *Proceedings of the 20th international conference on World Wide Web*, pages 587–596. ACM, 2011.

[3] Paolo Boldi, Marco Rosa, and Sebastiano Vigna. HyperANF: Approximating the neighbourhood function of very large graphs on a budget. In Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar, M. P. Ravindra, Elisa Bertino, and Ravi Kumar, editors, *Proceedings of the 20th international conference on World Wide Web*, pages 625–634. ACM, 2011.

[4] Paolo Boldi and Sebastiano Vigna. The WebGraph framework I: Compression techniques. In *Proc. of the Thirteenth International World Wide Web Conference (WWW 2004)*, pages 595–601, Manhattan, USA, 2004. ACM Press.

[5] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. Graph structure in the Web: experiments and models. *Computer Networks*, 33(1–6):309–320, 2000.

[6] P. Crescenzi, R. Grossi, M. Habib, L. Lanzi, and A. Marino. On Computing the Diameter of Real-World Undirected Graphs. Presented at Workshop on Graph Algorithms and Applications (Zurich–July 3, 2011) and selected for submission to the special issue of Theoretical Computer Science in honor of Giorgio Ausiello in the occasion of his 70th birthday, 2011.

[7] Pierluigi Crescenzi, Roberto Grossi, Claudio Imbrenda, Leonardo Lanzi, and Andrea Marino. Finding the diameter in real-world graphs: Experimentally turning a

lower bound into an upper bound. In Mark de Berg and Ulrich Meyer, editors, *Algorithms - ESA 2010, 18th Annual European Symposium, Liverpool, UK, September 6-8, 2010. Proceedings, Part I*, volume 6346 of *Lecture Notes in Computer Science*, pages 302–313. Springer, 2010.

[8] Pierluigi Crescenzi, Roberto Grossi, Leonardo Lanzi, and Andrea Marino. A comparison of three algorithms for approximating the distance distribution in real-world graphs. In Alberto Marchetti-Spaccamela and Michael Segal, editors, *Theory and Practice of Algorithms in (Computer) Systems*, volume 6595 of *Lecture Notes in Computer Science*, pages 92–103. Springer Berlin, 2011.

[9] Bradley Efron and Gail Gong. A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician*, 37(1):36–48, 1983.

[10] Philippe Flajolet, Éric Fusy, Olivier Gandouet, and Frédéric Meunier. HyperLogLog: the analysis of a near-optimal cardinality estimation algorithm. In *Proceedings of the 13th conference on analysis of algorithm (AofA 07)*, pages 127–146, 2007.

[11] Sharad Goel, Roby Muhamad, and Duncan Watts. Social search in "small-world" experiments. In *Proceedings of the 18th international conference on World wide web*, pages 701–710. ACM, 2009.

[12] Michael Gurevitch. *The social structure of acquaintanceship networks*. PhD thesis, Massachusetts Institute of Technology, Dept. of Economics, 1961.

[13] Jon M. Kleinberg. Navigation in a small world. *Nature*, 406(6798):845–845, 2000.

[14] Silvio Lattanzi, Alessandro Panconesi, and D. Sivakumar. Milgram-routing in social networks. In *Proceedings of the 20th international conference on World wide web*, pages 725–734. ACM, 2011.

[15] Jure Leskovec and Eric Horvitz. Planetary-scale views on a large instant-messaging network. In *Proceeding of the 17th international conference on World Wide Web*, pages 915–924. ACM, 2008.

[16] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2–es, 2007.

[17] Lun Li, David L. Alderson, John Doyle, and Walter Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Math.*, 2(4), 2005.

[18] David Liben-Nowell, Jasmine Novak, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Geographic routing in social networks. *Proceedings of the National Academy of Sciences of the United States of America*, 102(33):11623–11628, August 2005.

[19] Clémence Magnien, Matthieu Latapy, and Michel Habib. Fast computation of empirically tight bounds for the diameter of massive graphs. *J. Exp. Algorithmics*, 13:10:1.10–10:1.9, 2009.

[20] Stanley Milgram. The small world problem. *Psychology Today*, 2(1):60–67, 1967.

[21] Christopher R. Palmer, Phillip B. Gibbons, and Christos Faloutsos. Anf: a fast and scalable tool for data mining in massive graphs. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 81–90, New York, NY, USA, 2002. ACM.

[22] Anatol Rapoport and William J. Horvath. A study of a large sociogram. *Behavorial Science*, 6:279–291, October 1961.

[23] Jeffrey Travers and Stanley Milgram. An experimental study of the small world problem. *Sociometry*, 32(4):425–443, 1969.

[24] Qi Ye, Bin Wu, and Bai Wang. Distance distribution and average shortest path length estimation in real-world networks. In *Proceedings of the 6th international conference on Advanced data mining and applications: Part I*, volume 6440 of *Lecture Notes in Computer Science*, pages 322–333. Springer, 2010.

Four Degrees of Separation, Really

Paolo Boldi Sebastiano Vigna
Dipartimento di Informatica
Università degli Studi di Milano
Italy

***Abstract*—We recently measured the average distance of users in the Facebook graph, spurring comments in the scientific community as well as in the general press [1]. A number of interesting criticisms have been made about the meaningfulness, methods and consequences of the experiment we performed. In this paper we want to discuss some methodological aspects that we deem important to underline in the form of answers to the questions we have read in newspapers, magazines, blogs, or heard from colleagues. We indulge in some reflections on the actual meaning of "average distance" and make a number of side observations showing that, yes,** 3.74 **"degrees of separation" are really few.**

FOUR DEGREES OF SEPARATION

In 2011, together with Marco Rosa, we developed a new tool for studying the distance distribution of very large (unweighted) graphs, called HyperANF [2]: this algorithm built on powerful graph compression techniques [3] and on the idea of diffusive computation pioneered in [4]. The new tool made it possible to accurately study the distance distribution of graphs orders of magnitude larger than it was previously possible. The work on HyperANF was presented at the 20th World-Wide Web Conference, in Hyderabad (India), and Lars Backstrom happened to listen to the talk; he was intrigued by the possibility of experimenting our software on the Facebook graph and suggested a collaboration.

Experiments were performed in the summer of 2011, resulting in the first world-scale social-network graph-distance computations, using the entire Facebook network of active users (721 million users, 69 billion friendship links). The average distance (i.e., shortest-path length) observed was 4.74, corresponding to 3.74 intermediaries (or "degrees of separation", in Milgram's parlance). These and other findings were finally presented in [1] and made public by Facebook through its technical blog on November 19, 2011. Immediately after the announcement, the news appeared in the general press, starting from the New York Times [5][1] and soon spreading worldwide in newspapers, blogs and forums.

A number of interesting criticisms have been made about the meaningfulness, methods and consequences of the experiment we performed. In this paper we want to discuss some methodological aspects that we deem important. We shall consider such issues in an answer-to-question style, with the double aim of replying to doubts and attacks and of stimulating new discussions and further interest.

I. NOT ALL PAIRS ARE CONNECTED: HOW CAN THE AVERAGE DISTANCE BE EVEN FINITE?

If by "average distance" we mean "average of the distances between all pairs", of course Facebook has an infinite average distance, as we know that there is a very large connected component containing almost all (99.9%) nodes, but there are also some (few) unreachable pairs.

This is an interesting comment, as it shows an actual black hole in all the literature: people studying social problems (starting with the 50s, at least) had in mind very small groups, possibly groups that would fit one room (actually, in some cases, just sitting around a table). Or small communities. The very idea of "unreachable" was not part of the picture. In the famous paper by Travers and Milgram [6], the vast majority of postcards did not

Partially supported by a Yahoo! faculty grant and by by the EU-FET grant NADINE (GA 288956).

[1]Incidentally, with an off-by-one error, as 4.74 is the average *distance*, whereas the average number of degrees of separation is 3.74 (see [1]).

reach the target[2]. Nonetheless, the "six degrees of separation" idea came from the average distance (5.4 to 6.7, depending on the group) obtained in the experiment, computed *just on reachable pairs*.[3]

We discuss here in some detail two possible mathematical solutions to this problem—not only because they are interesting, but because we want to urge researchers to take the problem into consideration more seriously, and to remark to those objecting to the use of reachable pairs that old results would be really stated differently if unreachable pairs were correctly taken into account.

An obvious patch is to quote the average distance between reachable pairs, sided by the percentage of reachable pairs, which should be considered as a sort of *confidence* on the measure. If the percentace of reachable pairs is low, the average distance is telling us little. On a completely disconnected graph, the average distance is 0, but with "confidence" $1/n$. On a perfect match,[4] the average distance is $1/2$, but the "confidence" is $2/n$ (in both cases, almost zero for large graphs).

Seen in this perspective, Milgram's experiment proposes an average distance of 6.2 but provides an incredibly low level of confidence—just 22%,[5] whereas in our case we can claim confidence 99.9% for our value (4.74).

The problem is that we like to compare results, and comparing two pairs of numbers can be difficult, if not impossible (see, e.g., the plethora of methods used to combine somehow precision and recall in information retrieval).

A solution that does not show the latter drawback is to consider *harmonic means* when working with distances. We recall that the harmonic mean is the reciprocal of the mean of the reciprocals. It is always smaller than the arithmetic mean, as it tends to give less relevance to large outliers and more relevance to small values, and it is used in a number of contexts[6].

The important feature of the harmonic mean is that if we stipulate that $1/\infty = 0$, it can take in ∞ as a perfectly valid distance. Its effect is that of making the mean larger in a hyperbolic fashion. This is why Marchiori and Latora [9] proposed to consider the harmonic mean of *all* distances between distinct nodes[7], which we call *harmonic diameter* following Fogaras [10] (rather than "average distance *between reachable pairs*"), as a measure of tightness of a network. For instance, a disconnected graph has average distance zero, but infinite harmonic diameter; and a perfect match has average distance $1/2$, but harmonic diameter $n-1$.

What happens if we switch from the average distance to the harmonic diameter? On highly disconnected network, with many missing paths, we get a larger number. On the LAW web site[8] you can find the basic statistics of several web-graph snapshots, and the harmonic diameter is always significantly larger than the average distance between reachable pairs.

In the case of Facebook, the harmonic diameter is 4.59—even smaller than the average distance. The situation, however, is quite different if we make the same computation with Milgrams' experiment and assume that incomplete chains correspond to unreachable pairs: overall, the harmonic mean is 18.29, almost four times larger than the average distance. If we restrict to the Nebraska random group (i.e., we avoid geographical or cultural clues), the harmonic mean is more than five times larger. By this measure, the improvement described in [1] is even more impressive.

The problem with the harmonic diameter is that even if it is a clearly and sensibly defined mathematical feature, it deprives us from the "degree of separation" metaphore. The fact that in 2007 the harmonic diameter of `it` was more than 15 000 does not mean, of course, that you need to pass through

[2]It should be noted, as an aside, that in Milgram's experiment the interrupted chains do not actually imply unreachability, a point that will be better discussed later.

[3]Indeed, the authors of one of the first studies of the web as a whole [7] noted the same problem, and proposed the name *average connected distance*. We refrain, however, from using the word "connected" as it somehow implies a bidirectional (or, if you prefer, undirected) connection. The notion of average distance between all pairs is useless in a graph in which not all pairs are reachable, as it is necessarily infinite, so no confusion can arise.

[4]A *perfect match* is an undirected 1-regular graph, that is, a set of disconnected edges.

[5]Travers and Milgram's paper [6] reports 29%, as this is the percentage of chains that *started and completed* with respect to those that *started*. Some of the chains did not start at all, and we are considering them as incomplete, which explains the slightly slower value we are reporting.

[6]Incidentally, the HyperLogLog counters [8] used by HyperANF [2], the algorithm with which the average distance of Facebook was computed, use the harmonic mean to perform stochastic averaging.

[7]The fact that we do not consider the distances $d(x,x)$ is essential, as otherwise the harmonic mean becomes zero.

[8]`http://law.dsi.unimi.it/`

TABLE I
HARMONIC DIAMETER OF THE GRAPHS FROM [1].

	it	se	itse	us	fb
2007	15083.99 (±298.82)	51.07 (±1.50)	3760.77 (±161.28)	4.16 (±0.14)	6.33 (±0.26)
2008	23.66 (±0.75)	4.37 (±0.15)	6.44 (±0.21)	4.61 (±0.16)	5.74 (±0.24)
2009	4.74 (±0.11)	4.37 (±0.11)	4.71 (±0.11)	4.67 (±0.16)	5.07 (±0.21)
2010	3.92 (±0.13)	3.90 (±0.16)	4.24 (±0.18)	4.68 (±0.15)	5.03 (±0.21)
2011	3.76 (±0.11)	3.93 (±0.16)	4.29 (±0.18)	4.23 (±0.13)	4.70 (±0.30)
current	3.68 (±0.10)	3.69 (±0.20)	3.90 (±0.13)	4.45 (±0.11)	4.59 (±0.13)

TABLE II
THE HARMONIC MEAN AND THE MEAN OF ALL DISTANCES (INCLUDING ∞ FOR BROKEN CHAINS) FOR THE GROUPS DETAILED IN TRAVERS AND MILGRAM'S PAPER [6]. NOTE THE SIGNIFICANTLY LOWER VALUE OF THE HARMONIC MEAN FOR THE BOSTON GROUP.

Group	Harmonic mean	Median distance
Nebraska random	26.68	∞
Nebraska stockholders	19.37	∞
All Nebraska	22.40	∞
Boston random	12.63	∞
All	18.29	∞

15 000 friendship links!

Another possibility for taking into account infinite distances is to use the *median of all distances* as a measure of closeness. That is, we list in increasing order the n^2 values of $d(x, y)$, and we take that of index $\lfloor n^2/2 \rfloor$ (numbering from zero). This number is significantly larger than the average distance if several pairs are unreachable because the ∞ values at the end of the list "push" the median to the right. Again, on the LAW web site you can see that in several web graphs the median of all distances is significantly larger than the average distance, as it takes into account the existence of unreachable pairs. It is a good idea to complement the median with the fraction of pairs within its value: in any case, we know that at least 50% of the pairs (of *all* pairs, not just the reachable ones) are within its value, which gives us a concrete handle.

The median of all distances for Facebook is 5 (and 92% of all pairs is within this distance). So, again, "four degrees of separation". Obviously, for Milgram in all cases the median is ∞. So, using this measure we progressed really a lot.

With the collaboration of Jure Leskovec we were able to compute similar measures for Horvitz and Leskovec's Messenger experiment [11]: the average distance, 6.618, has confidence 71.3%; the harmonic diameter is 8.935, whereas the median distance is 7, covering 78.7% of all pairs.[9] Note that these figures are due to the presence of isolated nodes, that is, nodes that did not participate in any communication in the observed month: if the graph is reduced to non isolated nodes, essentially all values collapse.

II. THE SAMPLE IS BIASED, AND ANYWAY IT JUST REPRESENTS 10% OF HUMANITY!

As a first consideration, we invite the reader to observe that there is no such things as a "uniform" or "unbiased" sample of a graph. One can, of course, sample the *nodes* or the *arcs* of a graph, and consider the induced subgraph, but there is no guarantee that the induced subgraph preserves the properties of interest of the whole graph—much more sophisticated strategies are necessary, and in any case, it must be proved beforehand that the selected strategy creates an induced subgraph that is sufficiently similar to the whole graph (whatever notion of "similar" we want to take into account).

In any case, let us take a step back and look for a moment at the conditions of Milgram's experiment:

- *number of pairs examined*: 296;
- *sample of the population*: 100 United States citizens living in Boston, 96 random United States citizens living in Nebraska, 100 stockholders living in Nebraska;
- *completed chains*: $\approx$ 22%;
- *definition of link*: instructions to send the letter only to a "first-name acquaintance".

Our case:

- *number of pairs examined*: 250 millions of billions;
- *sample of the population*: 721 million people spread in several continents;
- *completed chains*: $\approx$ 99.8%;

[9] We cannot report statistical metadata such as the standard error, because we were provided with already-aggregated breadth-first samples only.

- *definition of link*: sharing a friendship link on Facebook.

We realize, obviously, that Facebook is not a random sample, and that being on Facebook implies already sharing a mindset, or certain areas of interest. We are also aware of the digital divide problem (that introduces a strong geopolitical and economical bias) and that there are links on Facebook between people that never met each other in person (e.g., gamers).

On the other hand, a random sample of 96 people from Nebraska is not a random sample of the world population, either. And, again, we will never know if some letters in the experiment actually passed through, say, two pen pals who never met in person. What a lot of people did not realize is that, essentially, the only thing we know about how people were involved in Milgram's experiment is that the sender judged that it had a "first-name acquaintance" with the receiver. The link between sender and receiver might have been in some cases even *weaker* than sharing a friendship link of Facebook.

There is, moreover, another important factor to take into account: since there will be many first-name acquaintances who are *not* on Facebook (and hence not Facebook friends) some short paths will be missing. These two phenomena will likely, at least in part, balance each other; so, although we do not have (and cannot obtain) a precise proof of this fact, we do not think we are losing or gaining much in considering the notion of Facebook friend as a surrogate of first-name friendship.

All in all, we see a definite progress in stating that the world is small. Thanks to Facebook, which is the largest ever-created database of human relationships, we have been able to make Milgram's experiment (or at least the part of it that has to do with measuring shortest paths) much more concrete and objectively measurable.

Nonetheless, let us take another step back and consider, for a moment, the genius of a man who approached a mind-boggling (even for us, now) problem on a worldwide scale armed with three hundred postcards and an incredibly clever experiment. Obtaining a result almost unbelievably close to what we obtained using a number of pairs that is *fifteen orders of magnitude larger*. One is tempted to draw a comparison with Galileo's celebrated mental experiment in the *Dialogo sopra i due massimi sistemi del mondo* [12]: you do not need an expensive lab to test the principle of relativity—you just need a ship, some butterflies and some fish. Of course, once you do it, an expensive lab to check it thoroughly is definitely not a bad idea.

III. You measured the average distance, but degrees of separation are algorithmic

Just after we disseminated our paper, we learned that an experiment was trying to settle the "degree of separation" problem, which was "still unresolved" using Facebook.[10] We were, of course, quite surprised. While we certainly did not "resolve" anything, it was difficult to imagine an experiment at present time with a larger sample or significantly more precise measurements.

The point is the distinction between "routing" and "distance". Milgram's postcard were routed locally (each sender did not know whether the recipient was the best choice to get to the destination, i.e., if it lay on a shortest path to the destination). Apparently, the question is still unresolved because by studying Facebook we have only computed the "topological", not the "algorithmic" degrees of separation.

We believe, however, that this is a red herring. Reading carefully Travers and Milgram's papers [13], [6], it is clear that the very purpose of the authors was to estimate the number of intermediaries: the postcards were just a tool, and the details of the paths they followed were studied only as an artifact of the measurement process. In the words of Milgram, the problem was defined by "given two individuals selected randomly from the population, what is the probability that the minimum number of intermediaries required to link them is 0, 1, 2, ..., k?". Said otherwise, Milgram was interested in estimating the *distance distribution* of the acquaintance graph.

The interest in efficient routing lies more in the eye of the beholder (e.g., the computer scientist) than in Milgram's: if he had at his disposal an actual large database of friendship links and algorithms like the ones we used, he would have dispensed with the postcards altogether. Thus, the fact that we measured *actual* shortest paths between individuals, instead of the paths of a greedy routing, is a definite progress. Routing is an interesting computer-science

[10] `http://smallworld.sandbox.yahoo.com/`.

(and sociological) problem, but it had little or no interest for Milgram—actually, the main interest in the routing process was understanding the convergence of paths. From the paper:

> The theoretical machinery needed to deal with social networks is still in its infancy. The empirical technique of this research has two major contribution to make to the development of that theory. First it sets an upper bound on the minimum number of intermediaries required to link widely separated Americans. Since subjects cannot always foresee the most efficient path to a targer, our trace procedure must inevitably produce chains longer than those generated by an accurate theoretical model which takes full account of all paths emanating from an individual.

That said, the results obtained in Milgram's experiment are even more stunning because the average routing distance they computed (with the provisos about uncompleted chains discussed above) is so close to the average shortest-path length. The latter observation seems to suggest that human beings are extremely good at routing, so good that they almost route messages along the shortest possible path. However, taking uncompleted paths into consideration gives a slightly different twist to this remark: it seems that when someone felt confident enough to continue the experiment, (s)he did so almost in the best possible way; but more often than not, the experiment was stopped probably because the message arrived at an individual that did not know how to route it further efficiently.

Apart for the attempts to measure the routing distance in real-world social graphs, there is an ever increasing focus on developing a theory of distributed efficient routing on small worlds, starting from Kleinberg's intriguing notion of navigability [14], [15]; this is however outside of the scope of our paper.

IV. Just add a few links here and there and we'll all be at one degree of separation

Another, closely related, question is: "We have seen that the degree of separation has constantly decreased since 2008, reaching its current value. What can we expect for the future?"

To answer the above comment/question, notice that the average distance is

$$\sum_{k>0} kP_k/r,$$

where P_k is the number of pairs at distance exactly k and r is the number of reachable pairs, which is n^2 if and only if the graph is strongly connected. Of course, if we have bounds $B_k \geq P_k$ for some $1 \leq k \leq \ell$, it is immediate to see that, if $\sum_{k=1}^{\ell-1} B_k \leq r$ then

$$\sum_{k>0} kP_k \geq \sum_{k=1}^{\ell-1} kB_k + \ell\Big(r - \sum_{k>0} B_k\Big). \qquad (1)$$

Now, depending on how much you want to consider a graph similar to the Facebook graph described in [1], there are many ways to generate some B_k's.

a) First bound (depending on n, m and D).: There are instrinsic bounds on the number of short paths you can generate when the number of neighbours of a node is limited. The simplest observation is that (letting D be the maximum degree and m be the number of arcs in the graph, i.e., twice the number of edges) you cannot have more than m pairs at distance one, mD pairs at distance 2, and so on; more precisely, we can set $B_k = mD^{k-1}$, getting (from (1)) the lower bound

$$\sum_{k>0} kP_k \geq m + 2mD + 3(r - m - mD)$$

provided that $m + mD \leq r$; in the case of Facebook ($D = 5000$, $n \approx 721 \times 10^6$, $r = 5 \times 10^{17}$, $m \approx 69 \times 10^9$) the inequality $m + mD \leq r$ is satisfied and the lower bound obtained is ≈ 2.999. In other words, no graphs with the same number of nodes, arcs and maximum outdegree of the graph we considered can have an average distance smaller than 2.999.

b) Second bound (depending on the degree sequence).: To improve over the previous trivial bound, we can use the actual degree distribution.[11] This is a bit like answering to the question: what if some omniscent being "rewired" Facebook in an optimised way to reduce the average distance as much as possible, but leaving each user with its current number of friends? Let us first notice that P_2 can be bounded by $\sum_x d(x)^2$, which, being the sum of entries of the square of the adjacency matrix, is

[11] The degree distribution is publicly available as part of the dataset associated with [1].

an upper bound for the number of pairs at distance 2. Providing a good bound for P_3 is slightly more difficult:

Theorem 1 *Let $d_0 \geq d_1 \geq \dots d_{n-1}$ be the degree sequence of the graph, $s = \sum_{i=0}^{n-1} d_i^2$ and define, for every t,*

$$\delta(t) = \sum_{i=0}^{d_t - 1} d_i.$$

Then P_3 (the number of pairs of nodes at distance exactly 3) can be bounded by

$$P_3 \leq \sum_{k=0}^{\ell} d_k \delta(k) + d_{\ell+1}\Big(s - \sum_{k=0}^{\ell} \delta(k)\Big)$$

where ℓ is the greatest integer such that $\sum_{k=0}^{\ell} \delta(k) < s$.

Proof: We can bound P_3 from above by counting the number p of tuples (u_i, v_i, w_i, z_i) corresponding to paths of length 3. Let $V = \{v_0, \dots, v_{k-1}\}$ be the set of nodes appearing as second component in at least one such tuple, sorted by non-increasing node degree; clearly $p \leq d(v_0)\pi(v_0) + \dots + d(v_{k-1})\pi(v_{k-1})$ where $d(x)$ is as usual the degree of x and $\pi(x)$ is the number of paths of length 2 starting from x: this is because every single path of length 3 of the form $(-, v_i, -, -)$ is obtained by choosing a neighbor of v_i and a path of length 2 leaving from v_i.

Observe that $\pi(v_0) + \dots + \pi(v_{k-1})$ cannot be larger than s (because the latter is an upper bound to the number of paths of length 2 in the graph). Now, of course, for every $t = 0, \dots, k-1$, $d(v_t) \leq d_t$, so $p \leq d_0\pi(v_0) + \dots + d_{k-1}\pi(v_{k-1})$; it is convenient to think of the latter as a summation of a list L of length $s \geq \pi(v_0) + \dots + \pi(v_{k-1})$, where d_0 occurs $\pi(v_0)$ times, d_1 occurs $\pi(v_1)$ times etc., and at the end of the list 0 occurs enough times to reach the desired length.

Now $\pi(v_t)$ can be bounded from above by the number of paths of length 2 leaving from a node of degree d_t. But the latter can be obtained by choosing at the first step the d_t nodes with largest degree, and summing up their degree; that is, $\pi(v_t) \leq \delta(t)$. So we can safely substitute the above list L with another list L' of the same length where d_0 is repeated $\delta(0) \geq \pi(v_0)$ times, d_1 is repeated $\delta(1) \geq \pi(v_1)$ times etc. The resulting list L' dominates L elementwise, hence the thesis. ■

Plugging $B_1 = m$, $B_2 = \sum_{i=0}^{n-1} d_i^2$ and B_3 as in Theorem 1, and using the actual degree sequence of Facebook, we obtain ≈ 3.6. Thus, Facebook is essentially just one step (distance or degree doesn't matter) away from the best possible, given that every individual keeps the current number of friends.

V. It's just because of the nodes with very high degree that we observe such a low value

Since the first studies on the structure of complex graphs [16], and in particular of social networks, the degree distributions have been a central topic on which many authors focused, concluding that both in- and out-degrees exhibit a heavy-tailed distribution: this fact implies that there are many nodes whose degree largely exceeds the average. It is a widely assumed tenet that those nodes, sometimes referred to as *hubs*, represent a sort of "social glue" that keeps the whole network structure together and that shortcut friendship paths. In the case of social networks, such as Twitter or Facebook, hubs are superstars like Lady Gaga or Barack Obama, whose account often do not even correspond to real persons.

But, is this the case? In our analysis of the Facebook graph we excluded *pages* (the accounts that people may "like"), and standard accounts have a hardwired limit of 5 000 friends. Nonetheless, we cannot rule out the possibility that there are some fake celebrity accounts remaining in the graph we studied.

The general question we are asking can be restated as follows: take a social network and start removing the nodes of largest degrees; how much does the distribution of distances change? in particular: how does the average distance change (presumably: increase)? We considered this question in a previous paper [17] (see also [18]), where we actually studied the more general problem of which removal strategies are more disruptive under the viewpoint of distance distributions.

We report an anticipation of a subset of the results of [18], as they suggest that high-degree node removal is not going to cause drastic changes in the structure of the network. We show results for a

TABLE III
CHANGE IN AVERAGE DISTANCE OF WEB AND SOCIAL GRAPHS AFTER REMOVING THE LARGEST (IN-)DEGREE NODES. THE REMOVAL PROCESS IS STOPPED WHEN THE NUMBER OF ARCS REMOVED REACHES THE 10% AND 30%.

Graph	original	10%	30%
.in	15.34	16.11 (+5.0%)	18.98 (+23.7%)
Hollywood	3.92	4.02 (+2.5%)	4.23 (+7.9%)
LiveJournal	5.99	6.15 (+2.7%)	6.55 (+9.3%)
Orkut	4.21	4.43 (+5.2%)	4.67 (+10.9%)

TABLE IV
CHANGE IN HARMONIC DIAMETER OF WEB AND SOCIAL GRAPHS AFTER REMOVING THE LARGEST (IN-)DEGREE NODES. THE REMOVAL PROCESS IS STOPPED WHEN THE NUMBER OF ARCS REMOVED REACHES THE 10% AND 30%.

Graph	original	10%	30%
.in	32.26	47.03 (+45.8%)	87.68 (+171.8%)
Hollywood	4.08	4.12 (+1.0%)	4.40 (+7.8%)
LiveJournal	7.36	7.74 (+5.2%)	8.67 (+17.8%)
Orkut	4.06	4.33 (+6.7%)	4.61 (+13.6%)

small[12] snapshot of the Indian web (.in), for the Hollywood co-starship graph, for a snapshot of the LiveJournal network kindly provided by the authors of [19], and a snapshot of the Orkut network kindly provided by the authors of [20].[13]

The results we obtained are the following. Removing largest-degree nodes does affect the average distance on web graphs: after the removal of 30% of the arcs[14] the average distance gets increased of about 24%. Nonetheless, the same removal strategy seems to have a weaker impact on genuine social networks: under the same condition, the increase in average distance ranges between 8% and 11% (see Table III).

Nonetheless, we are actually missing a very important point: in the social networks we studied, removing 30% of the arcs actually does not change the percentage of reachable pairs, whereas in web graphs the percentage (which is already lower) is reduced by a half. As we discussed in Section I, the average distance turns out again to be a very rough and unrealiable measure when the number of unreachable pairs is large.

Thus, in Table IV we show what happens to the harmonic diameter. The results show that the increase for social networks is very modest (less than 20% after the removal of as many as the 30% of the arcs), whereas for web graphs the harmonic diameter almost triplicates! This confirms again that the harmonic diameter is more reliable value to be associated to the "tightness" or "connectedness" of a network.

We remark that LiveJournal and Orkut are people-to-people friendship networks as Facebook (note, however, that LiveJournal is directed). We believe that the resistance to high-degree removal is actually a common phenomenon in such networks, which prompts us to conjecture that similar node-removal prodedures will not change Facebook average distance or harmonic diameter significantly, albeit we have no empirical data to support our hypothesis at this point.

Actually, a more general conclusion obtained in the cited paper [17] is that social networks seem very robust to node removal, and we could not find any node order that determined radical changes in the distance distribution. This observation leaves an intriguing question still open to debate: if hubs are not the inherent cause behind short distances, then what is the *real* reason of this phenomenon?

VI. ARE YOU SAYING THAT FACEBOOK REDUCED THE AVERAGE DISTANCE BETWEEN PEOPLE?

Some of the comments in the general press took the outcomes of our experiments as an evidence that online social networks (such as Facebook) reduced the average distance between people; of course, this was not the purpose (neither the content) of the experiment and in any case there is no direct way to know if this is true or not, because our measurements *are performed on Facebook*. We can see, however, that the distance between Facebook users constantly decreased over time: it used to be 5.28 in 2008, 5.06 in 2010 and 4.74 in our most recent dataset. Whether this decrease is *due* to Facebook, or whether it simply Facebook reflecting

[12]Similar results have been obtained with a lesser degree of precision on a snapshot of a 100 million pages in [17]; computations are underway to obtain high-precision data similar to what we report here about the smaller snapshot, and the results will be included in the final version of this paper.

[13]All these datasets are public and available at http://law.dsi.unimi.it/. The identifiers of the datasets are in-2004, hollywood-2011, ljournal-2008 and orkut-2007.

[14]We emphasize that we remove nodes (in decreasing order of their in-degree) and all incident edges, but count how many *arcs* are removed, because it is the number of deleted arcs that determines the expected loss in connectivity. We invite the reader to consult [17] for more details.

better and better the situation in the "real world" is hard to say. In the former case, as someone suggested, we would be observing a reduction in path lengths due probably to the presence of *weak ties* [21] that hardly correspond to a real friendship relation and would probably not even show up in a non–electronically-mediated environment.

Understanding how online social networks are changing our way of interacting, communicating and thinking is absolutely beyond the scope of our paper, whose aim was much humbler and certainly not as far-reaching. We believe, however, that giving a concrete and realistic explanation of what is going on requires a co-ordinated effort and calls for an interdisciplinary endeavor, putting together sociology, psychology, computer science and mathematics. This is, we think, one of the most important challenges for people working in these disciplines, with yet unknown consequences of philosophical, social and even economical value.

REFERENCES

[1] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, and S. Vigna, "Four degrees of separation," Arxiv preprint arXiv:1111.4570, 2012, accepted at ACM Web Science 2012.

[2] P. Boldi, M. Rosa, and S. Vigna, "HyperANF: Approximating the neighbourhood function of very large graphs on a budget," in *Proceedings of the 20th international conference on World Wide Web*, S. Srinivasan, K. Ramamritham, A. Kumar, M. P. Ravindra, E. Bertino, and R. Kumar, Eds. ACM, 2011, pp. 625–634.

[3] P. Boldi and S. Vigna, "The WebGraph framework I: Compression techniques," in *Proc. of the Thirteenth International World Wide Web Conference (WWW 2004)*. Manhattan, USA: ACM Press, 2004, pp. 595–601.

[4] C. R. Palmer, P. B. Gibbons, and C. Faloutsos, "Anf: a fast and scalable tool for data mining in massive graphs," in *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2002, pp. 81–90.

[5] J. Markoff and S. Sengupta, "Separating you and me? 4.74 degrees," *The New York Times*, no. 325, p. B1, 21 November 2011.

[6] J. Travers and S. Milgram, "An experimental study of the small world problem," *Sociometry*, vol. 32, no. 4, pp. 425–443, 1969.

[7] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener, "Graph structure in the Web: experiments and models," *Computer Networks*, vol. 33, no. 1–6, pp. 309–320, 2000.

[8] P. Flajolet, É. Fusy, O. Gandouet, and F. Meunier, "HyperLogLog: the analysis of a near-optimal cardinality estimation algorithm," in *Proceedings of the 13th conference on analysis of algorithm (AofA 07)*, 2007, pp. 127–146.

[9] M. Marchiori and V. Latora, "Harmony in the small-world," *Physica A: Statistical Mechanics and its Applications*, vol. 285, no. 3-4, pp. 539 – 546, 2000.

[10] D. Fogaras, "Where to start browsing the web?" in *Innovative Internet Community Systems, Third International Workshop, IICS 2003*, ser. Lecture Notes in Computer Science, vol. 2877. Springer, 2003, pp. 65–79.

[11] J. Leskovec and E. Horvitz, "Planetary-scale views on a large instant-messaging network," in *Proceeding of the 17th international conference on World Wide Web*. ACM, 2008, pp. 915–924.

[12] G. Galilei, *Dialogo sopra i due massimi sistemi del mondo*. Landini, 1632.

[13] S. Milgram, "The small world problem," *Psychology Today*, vol. 2, no. 1, pp. 60–67, 1967.

[14] J. M. Kleinberg, "Navigation in a small world," *Nature*, vol. 406, no. 6798, pp. 845–845, 2000.

[15] ——, "The small-world phenomenon: an algorithm perspective," in *Proceedings of the 32nd ACM symposium on theory of computing*. ACM, 2000, pp. 163–170.

[16] A.-L. Barabási, R. Albert, H. Jeong, and G. Bianconi, "Power-law distribution of the World Wide Web," *Science*, vol. 287, p. 2115a, 2000.

[17] P. Boldi, M. Rosa, and S. Vigna, "Robustness of social networks: Comparative results based on distance distributions," in *Social Informatics, Third International Conference, SocInfo 2011*, ser. Lecture Notes in Computer Science, vol. 6894. Springer, 2011, pp. 8–21.

[18] ——, "Removal strategies for web and social graphs," 2012, submitted for publication.

[19] F. Chierichetti, R. Kumar, S. Lattanzi, M. Mitzenmacher, A. Panconesi, and P. Raghavan, "On compressing social networks," in *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2009, pp. 219–228.

[20] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and Analysis of Online Social Networks," in *Proceedings of the 5th ACM/Usenix Internet Measurement Conference (IMC'07)*, San Diego, CA, October 2007.

[21] M. Granovetter, "The Strength of Weak Ties," *The American Journal of Sociology*, vol. 78, no. 6, pp. 1360–1380, 1973.

Arc-Community Detection via Triangular Random Walks

Paolo Boldi Marco Rosa
Dipartimento di Informatica, Università degli Studi di Milano, Italy

***Abstract*—Community detection in social networks is a topic of central importance in modern graph mining, and the existence of overlapping communities has recently given rise to new interest in arc clustering. In this paper, we propose the notion of triangular random walk as a way to unveil arc-community structure in social graphs: a triangular walk is a random process that insists differently on arcs that close a triangle. We prove that triangular walks can be used effectively, by translating them into a standard weighted random walk on the line graph; our experiments show that the weights so defined are in fact very helpful in determining the similarity between arcs and yield high-quality clustering. Even if our technique gives a weighting scheme on the line graph and can be combined with any node-clustering method in the final phase, to make our approach more scalable we also propose an algorithm (ALP) that produces the clustering directly without the need to build the weighted line graph explicitly. Our experiments show that ALP, besides providing the largest accuracy, it is also the fastest and most scalable among all arc-clustering algorithms we are aware of.**

I. Introduction

Complex networks and, especially, social networks often exhibit a finer internal structure where individuals interact in small subgroups (called communities or modules), based on the individuals' common interests, geographic location, political opinions etc. Understanding how such subgroups are structured and evolve in time is essential for applications like targeted advertising, viral marketing, friend suggestion etc. Social-network mining traditionally identifies a community as a densely connected set of nodes that is in turn only loosely attached to the rest of the network [9]; in this view, community detection translates into finding a partition of the nodes that optimizes some quality function. Most of the literature on this topic focused on the discussion of the mutual merits of various quality functions and on the comparison of algorithms that try to optimize (in an exact or approximate way) some of those functions. It is worth noticing that we are here thinking of the clustering problem in a situation where the only available information is the (directed or undirected) graph underlying the social network, possibly with some weights on its arcs denoting the strength of that bound[1].

The main limit of the approach discussed above is that rarely a node is part of a single community: more often than not, communities overlap giving rise to a complex intertwining that can hardly be reflected into a node partition. For this reason, recent research (see, for example, [2], [17]) has turned its attention to the problem of finding overlapping communities, where each node can be a member of more than one module.

This idea is well motivated and neat for those (frequent) situations in which membership to multiple communities is an exception more than a rule, and most nodes belong clearly to one single communities, with a number of borderline individuals for whom membership is less straightforward. In a large number of scenarioes, however, belonging to more groups is a rule more than an exception, and actually the notion of node community hardly makes sense: like a point in the Cartesian plane belongs to infinitely many lines, an individual in a social network plays potentially infinitely many roles. In those cases, it is often more sensible and interesting to individuate *communities of arcs* rather than *communities of nodes*: this shift of interest (witnessed in the most recent literature [27]) can be thought of as trying to find the reasons behind relations rather than trying to find the reason behind individuals. Or, going on with our metaphor, it is like determining the line to which *two* given points belong—a single point lies on infinitely many lines, but there is only a single line passing through two given points.

This idea is clear if one thinks of social networks such as Facebook: every Facebook user has probably many interests and belongs to a multiplicity of communities; however, every friendship is probably due to one main reason (working together, being relatives, having the same hobby etc.). This thought is so natural that Google+ has explicitly introduced the notion of "circle", later adopted also by Facebook.

In this work, we propose to continue along this line of research trying to exploit the following simple observation: if xy and yz are two relations that have the same motivation (e.g., working together), then probably xz will also be present: in other words, triangles tend to live inside communities. Based on this intuition, we propose the notion of triangular random walk, a stochastic process that treats differently triangular and non-triangular arcs; although this process is not memoryless, we can reduce it to a standard Markov chain on the line graph (using a tool similar to [8], but in a different way). With our approach, we obtain a weighted version of the line graph (a graph whose nodes correspond to the arcs of the original network). The weighted line graph can in turn be clustered using standard tools, hence employing state-of-the-art algorithms for the actual clustering phase: the main

Partially supported by a Yahoo! faculty grant and by by the EU-FET grant NADINE (GA 288956).

[1]Even if other information about vertices and edges may be available, it is usually computationally unfeasible to leverage it to detect communities.

978-0-7695-4839-5/12 $26.00 © 2012 IEEE
DOI 10.1109/LA-WEB.2012.19

limit of this approach is that the line graph is itself some orders of magnitudes larger than the original graph, so even its construction can become a computational burden (let alone the time and resources that the clustering algorithm will then require). For this reason, we develop an *ad hoc* version, called ALP, of a well-known clustering technique that carries out the clustering on the weighted line graph without having to compute it explicitly. Experiments on real-world networks of different sizes and types show that triangular walks can be extremely helpful in finding meaningful communities, outperforming significantly all other approaches; moreover, ALP turns out to be very efficient and can be used on large networks for which all other approaches would be prohibitive.

To summarize, the main contributions of this paper are: a) the definition of two weighting schemes (called w_T and v_T in this paper) for the arcs of the line graph that allow one to individuate arc-communities in the underlying graph; b) a clustering algorithm (ALP) that is able to use such schemes without the need to compute the line graph explicitly; c) a series of experiments proving that the weighting schemes proposed produce a significant improvement over all known techniques (in terms of quality, independently of the clustering algorithm adopted), and that ALP in itself can obtain the same results much more efficiently; in fact, it is the fastest and most scalable among all arc-clustering algorithms we are aware of.

II. Triangular random walks

Given a (directed) graph $G = (V_G, A_G)$ with no self-loops, we let $n_G = |V_G|$ and $m_G = |A_G|$ be the number of nodes and arcs of G, respectively; for every node x we let $N_G(x)$ be the set of *successors* of x and $d_G(x) = |N_G(x)|$ (the *(out)degree* of x). If G is symmetric (i.e., undirected), we use the term *edge* to refer to an unordered pair of nodes that are connected by an arc. We sometimes write xy to denote the arc (x, y) (or the edge $\{(x, y), (y, x)\}$, if the graph is undirected).

A *random walk* on a directed graph G is a stochastic process $X_0, X_1, \ldots$ where $X_0, \cdots \in V$, and for each $x, y \in V$, $P[X_0 = x] = 1/n$ and $P[X_{t+1} = y \mid X_t = x]$ is $1/d(x)$ if $y \in N(x)$, 0 otherwise[2]; this definition can be easily extended to positively weighted graphs (making $P[X_{t+1} = y \mid X_t = x]$ proportional to the weight of (x, y)). Intuitively, a random walk describes the behavior of a surfer wandering in the graph, who starts from a random node and at each step chooses uniformly at random (or proportionally to the weights) among the successors of the current node (jumping to a random node if the current one has no successors).

The random walk is a Markov chain and if G is undirected, connected and not bipartite, then the random walk has a unique stationary distribution $\mathbf{v}$ with $v_x = d(x)/2m$ [22]. For a general graph, however, the random walk is not ergodic, hence the stationary distribution may not be unique; to circumvent this problem, one can introduce [4], [13], [24] the notion of restart.

[2]For the sake of completeness, when $d(x) = 0$ we let $P[X_{t+1} = y \mid X_t = x] = 1/n$ for all y.

For a fixed $\alpha \in [0, 1]$, a *random walk with restart with damping factor* α on G is a stochastic process $X_0, X_1, \ldots$ as before, but where the surfer chooses the next node as follows: with probability α she picks a node uniformly at random among the successors of the current node; with probability $1 - \alpha$, instead, she jumps to a random node in the graph[3]. The latter event is called *teleportation* or "restart". It can be shown [4] that for all $\alpha < 1$ the random walk with restart has a unique stationary distribution (actually, the PageRank of G with damping factor α); when $\alpha = 1$ we get back to the standard random walks, instead.

One suggestive way to think of this random process is the following: a random surfer is trying to collect some knowledge and every node represents an expert that may provide some piece of information. After the surfer has finished visiting expert x she receives a list of other possible people that x trusts; the surfer may decide (with probability α) to accept x's suggestion and to visit one of them, or may rather decide to do it her way and to teleport to a random expert instead.

It is interesting to observe that one may also actually consider the stationary distribution *on the arcs of* G: the probability $P[X_t = x, X_{t+1} = y]$ that the random surfer goes along the arc (x, y) is $P[X_{t+1} = y \mid X_t = x]P[X_t = x] = v_x(\alpha w(x, y) + (1 - \alpha)/n)$, where $\mathbf{v}$ is the stationary distribution on the nodes and $w(x, y)$ is the weight on the arc (x, y) (that is, $1/d(x)$ in the unweighted case). We will refer to this distribution as the *arc-stationary distribution*.

The main idea of this paper is that we want to introduce a bias in the behavior of the random surfer, by allowing her some amount of short-term memory; in particular, the choice of the next node will not depend only on the current node but *also on the previous one*. The bias is finalized to privilege (or punish) triangles, i.e., suggestions of the current node that were also suggested by the previous node. Whether we decide to privilege triangles or to punish them depends on our interpretation of triangles: if we think that the double suggestion reinforces the idea that the suggested node is reliable, we will privilege triangles; if otherwise we suspect that the double suggestion is rather a form of lobbying, we will tend to avoid triangles.

Thus, we will define a triangular random walk $X_0, X_1, \ldots$ on an *unweighted*[4] graph using two parameters, $\alpha, \beta \in [0, 1]$: α is a damping factor and will have the same meaning as before (it is used to decide whether to follow a link or to teleport); β will instead be used to determine whether triangles or non-triangles should be privileged.

Two subtly different definitions of triangular random walks can be given, depending on the specific meaning of β: we will call them mass-triangular and ratio-triangular, respectively. In a triangular random walk with parameters α and β, the next node (x_{t+1}) is chosen depending on the current node x_t and

[3]As before, if the current node has no successors then the next node is chosen at random among all nodes in the graph.

[4]As before, extending this notion to weighted graphs is trivial, but for the sake of readability in this paper we prefer to limit ourselves to the unweighted case.

on the previous node x_{t-1}, as follows: (i) with probability $1-\alpha$, we teleport: x_{t+1} is a randomly chosen node; (ii) otherwise, we choose among the successors $N(x_t)$ of the current node, but treating differently the *triangular successors* (the set $N(x_t) \cap N(x_{t-1})$) and the *non-triangular successors* (the set $N(x_t) \setminus N(x_{t-1})$)[5]; here, the two definitions differ: in the *(mass-)triangular random walk*, we first decide whether we shall select a non-triangular successor (with probability β) or a triangular one (with probability $1-\beta$); then, the specific non-triangular or triangular successor is chosen uniformly at random; in the *ratio-triangular random walk*, all triangular successors are selected with the same probability, say p, and all non-triangular successors with probability βp (p should be chosen so that the sum of such probabilities is 1).

The names we adopted for the two kinds of random walks should be evocative of the meaning of β: in the mass-triangular random walk, β is the overall amount of probability of choosing a non-triangular successor; in the ratio-triangular random walk, it is the ratio between the probability of choosing a(ny) non-triangular successor over the probability of choosing a(ny) triangular one.

The two kinds of processes coincide when $\beta = 0$ (in that case, they both only choose triangular successors, except when teleporting). Moreover, ratio-triangular random walks reduce to standard random walks with restart when $\beta = 1$ (because, in that case, the probability of choosing triangles and non-triangles is the same), whereas there is no choice of β that makes a mass-triangular random walk the same as a standard random walk. The latter observation may suggest that ratio-triangular random walks should be preferred, but the mathematical treatment of mass-triangular walks is simpler, and for this reason we shall actually treat the latter as our "default" type of triangular walk (and omit "mass" in the following). Triangular walks can have a number of potential applications; for example, they may be used fruitfully in bibliometrics to moderate the problem of nepotistic citations in scientific works (in this case, triangles should be punished rather than promoted). In this paper, however, we wish to speculate on the possible usage of triangular walks to single out arc-communities in social networks, where triangles are used as a form of reinforcement.

To start playing with our idea, let us consider Zachary's famous karate club network [28]: this is an undirected graph whose nodes represent the members of a karate club and with an edge between two individuals if they happened to have seen each other outside of the club for some reason; the club ended up splitting in two (in our drawings, the nodes are depicted differently according to the group they will end up in), and one can hope to find information about how the members will decide to group based solely on their friendship relations. We first tried a standard random walk on this dataset to see how frequently each edge was run through in either direction (Figure 1): no pattern is evident. But if we do the same with a triangular walk some edges get more emphasis, witnessing that some bounds are stronger than others (Figure 2, with $\beta = 0.2$): those edges are usually between members that will end up in the same group (with an exception concerning node 9 that indeed seems to be more strictly bound to the group of circles than to the group of squares). If we decrease β to 0.01, some clans would become almost grotesquely evident.

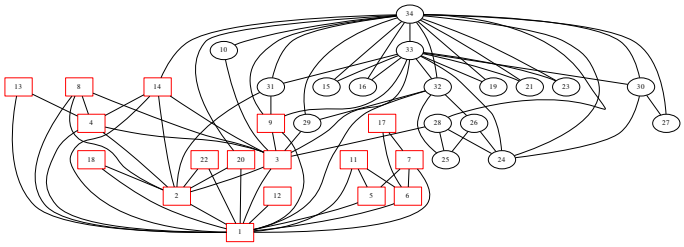


Fig. 1. Standard random walk on the karate club dataset; edge width is proportional to the frequency with which that edge was run through in either direction.

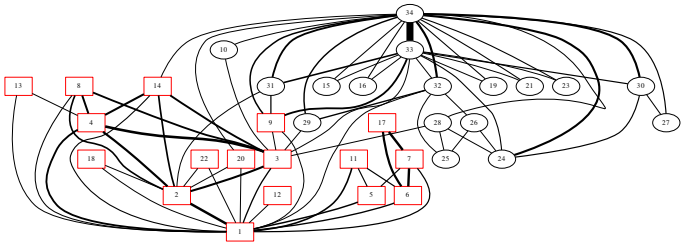


Fig. 2. Triangular random walk on the karate club dataset, with $\beta = 0.2$ (see also Figure 1).

A. Triangular walks and line graphs

A triangular random walk is a Markov chain of order 2 [22], because the next state depends on the current state *and* on the previous one. To study the long-term behavior of higher order chains, it is customary to change the state space and reduce the stochastic process to an equivalent one that is memoryless; this is easily solved by using the notion of *line graph*.

Given a graph G, its line graph $L = L(G)$ has the arcs of G as vertices (i.e., $V_L = A_G$), and arcs of the form (xy, yz) (where xy and yz are two arcs of G). Note that even when G is symmetric, $L(G)$ is not; for example, if G is the undirected graph in Figure 3, its corresponding line graph $L(G)$ is represented in Figure 3 (for the time being, ignore the colors on its arcs). The idea of using line graphs to study the behavior of an arc-aware random surfer was already proposed in [8], but they adopt a subtly different notion of line graph that is undirected; for our purposes, instead, the directed definition is much more well-suited (also because it adapts readily to the case when the original graph is itself directed).

[5]If either set is empty (or if $t = 1$) we choose uniformly in $N(x_t)$ (or in V, if the latter is empty), as in a standard random walk. The rationale behind this choice is that, based on the knowledge that we have (the current node and the previous one), all the outgoing arcs are equivalent.

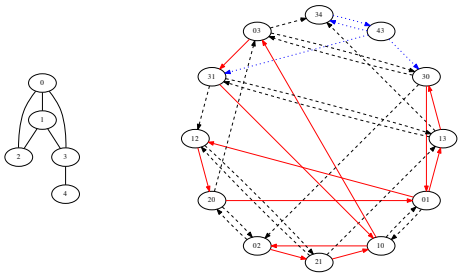


Fig. 3. A small undirected graph G (left) and the corresponding line graph $L(G)$. Continuous (red) arcs correspond to choosing triangular successors; dashed (black) arcs correspond to the choice of non-triangular successors; dotted (blue) arcs are used for the cases where either set is empty.

Now, it is easy to see that a triangular random walk with parameters α, β on the (unweighted) graph G is equivalent to a random walk with damping factor α on the weighted line graph $L(G)$, where

$$w_T(xy, yz) \triangleq \begin{cases} \frac{1-\beta}{|N(y)\cap N(x)|} & \text{if } z \in N(y) \cap N(x) \\ \frac{\beta}{|N(y)\setminus N(x)|} & \text{if } z \in N(y) \setminus N(x). \end{cases} \quad (1)$$

In other words, every arc in $L(G)$ (that is to say, every two-step walk $x \to y \to z$ in the original graph) has a different weight depending on whether it can be closed by a triangle (i.e., if $x \to z$ was also an arc of G) or not. If you look again at Figure 3, continuous (red) arcs correspond to the first case (e.g., $10 \to 03$ is one such arc, because 13 is also an arc of G), whereas dashed (black) arcs correspond to the second case (e.g., $31 \to 12$); note, in particular, that all arcs of the form $xy \to yx$ fall in the second class[6]. Some nodes of $L(G)$ (i.e., arcs of G) require some care, because their outgoing arcs are all non-triangular; those outgoing arcs are hence not weighted using the formula above (it would not make sense since one of the denominators is zero), but they have a constant weight instead (such arcs are drawn as dotted (blue) arrows in Figure 3).

For $\alpha < 1$ the random walk with restart on $L(G)$ weighted by w_T has a stationary distribution $\mathbf{v}_T$: note that, since the nodes of $L(G)$ are arcs of G, $\mathbf{v}_T$ assigns a probability $v_T(xy)$ with each arc xy of the original graph. Note also that, as explained in the previous section, the stationary distribution on the nodes of $L(G)$ induces a stationary distribution on its arcs:

$$v_T(xy, yz) \triangleq v_T(xy)(\alpha w_T(xy, yz) + (1-\alpha)/n_{L(G)}). \quad (2)$$

This is the fraction of time that the random surfer walking on $L(G)$ with weights w_T spends on the path $x \to y \to z$, and can be used as way of weighting the graph $L(G)$ alternative to (1).

Computing the stationary distribution $\mathbf{v}_T$ is a well-understood task (it amounts to a weighted version of PageRank) for which efficient and computationally sound algorithms exist [13], [26]; of course, $L(G)$ is larger than G (it has m_G nodes and $\sum_x d_G(x)^2$ arcs), but not much larger actually because of the sparsity of G and of the way its degrees are distributed. In particular, if G is undirected and has $\approx Ck^{-\alpha}$ nodes of degree k, then $L(G)$ will have $\approx C^2k^{-2\alpha}$ nodes of outdegree k.

[6]Differently from [8], we do not reserve stuttering walks (walks of the form $x \to y \to x$) a special treatment.

III. Arc-clustering via triangular random walks: a) using an off-the-shelf algorithm

As outlined in the previous sections, along the same line as [8], instead of clustering directly the arcs of G (as done, for example, by [12]), we turn to some suitably weighted version of the line graph $L(G)$, where we can make good use of all the paraphernalia for node-clustering of a directed graph. In other words, we can use an off-the-shelf node-clustering algorithm feeding it with the weighted (directed) graph $L(G)$. As weighting function (on the arcs of $L(G)$), we can use either of the weighting schemes defined in (1) and (2). For comparison, we may consider the weights of a standard random walk $w_S(xy, yz) = 1/d(y)$ or the corresponding arc stationary distribution $v_S(xy, yz)$ (as before, $v_S(xy)$ is the stationary distribution of the standard random surfer on the node xy); here, the subscript "S" stands for "standard". Another baseline is to feed the clustering algorithm with the unweighted graph $L(G)$ itself.

The main limit of the proposed method is that it cannot be directly applied to truly undirected graphs: since it is designed for directed graphs, reciprocal arcs (i.e., parallel arcs in opposite directions) may end up in two different communities. In cases when this fact can be a problem, one has to decide what to do about reciprocal arcs that happened to be clustered differently—one possible solution is to place the corresponding edge in either community, or to use a special community that corresponds to the given pair.

a) Computational issues: Computing the line graph $L(G)$ and its weights w_T is straightforward and can be performed in time $O(m_{L(G)})$ (i.e., linear in the output size), provided that one has direct access to G; moreover, although their size is obviously larger than the original graph (see Table I), line graphs turn out to be easily compressible (about 2 to 3 bits/link in their natural order, much less if suitably permuted [5]). After $L(G)$ has been produced, weighted PageRank can be computed very quickly (using for example the techniques of [7]), and in our experiments always resulted to converge in less than 20 iterations even for $\alpha = 1-10^{-2}$. The final node-clustering phase clearly depends on the algorithm used, but our method of choice [3] turns out to be reasonably fast — actually, the line graph construction is almost as expensive as the clustering itself. In fact, the explicit construction of the line graph is the main limit of this approach, especially for networks that are comparatively denser (such as Hollywood).

	n_G	$m_G = n_{L(G)}$	$m_{L(G)}$
free word assoc.	10 225	71 679	955 552
DBLP	986 324	6 707 236	211 808 396
Hollywood	2 180 759	228 985 632	242 026 293 162

TABLE I
SIZE OF LINE GRAPHS FOR SOME OF THE DATASETS WE SHALL USE IN SECTION VI; OBSERVE THAT HOLLYWOOD IS COMPARATIVELY DENSER THAN THE OTHER GRAPHS (WITH AN AVERAGE DEGREE OF ABOUT 105), WHICH IS WHY THE NUMBER OF ARCS IN $L(G)$ IS SO LARGE (THE AVERAGE DEGREE IS IN THIS CASE 1 057).

IV. ARC-CLUSTERING VIA TRIANGULAR RANDOM WALKS: B) USING ALP

When the graph is comparatively denser having to compute explicitly $L(G)$ can become a serious limitation; nonetheless, there is conceptually no need to do so—the graph $L(G)$ might be handled *implicitly*. If we want to approach the problem this way, however, we need to develop a specially tailored clustering algorithm that mimics what it would do on the (weighted version of) $L(G)$ without having it represented explicitly.

We tackled this idea by writing an implementation of the LP (Label Propagation) algorithm [18] that clusters the arcs of G based on an implicit representation of $L(G)$, weighted as in (1) or (2): we call this implementation ALP (for "Arc Label Propagation"); the reason behind the choice of LP with respect to other clustering algorithms is that it provides a good compromise between quality and speed. Moreover, due to its very diffusive nature, LP is best suited to translate into an algorithm that implicitly propagates information on the line graph. ALP takes G as input and works almost exactly as a standard LP [18] would do if run on $L(G)$, with the following adjustments: (a) LP is natively intended to be run on unweighted graphs, and it is based on a diffusive process where each node (arc, for ALP) decides whether to change its own label based on the majority of the labels in its neighboring nodes (arcs, for ALP); our adaptation to weighted graphs just changes the way majority is computed (summing up weights of neighbors instead of counting them); (b) Since LP is designed for undirected graphs, ALP actually considers the symmetrized version of $L(G)$ when being executed; in other words, an execution of ALP on G is equivalent to an execution of LP on a symmetrized weighted version of $L(G)$.

A final remark is that, if ALP is to be run with the weights v_T of (2), a preliminary computation of weighted PageRank on $L(G)$ should be performed; also this step can be carried out implicitly, without ever having to deal with $L(G)$.

V. RELATED WORK

Although node-clustering is traditionally much more developed and better understood (see [21] for an up-to-date survey), recently many authors advocated the adoption of link-clustering [27], [8], [12] as a way to overcome the problem of overlapping communities in complex networks. The advantage of this approach over the solution of soft or hierarchical node-clustering [15], [11] is that the latter is better suited for situations where the presence of a node in many communities is an exception rather than a rule; on the contrary, using link-clustering allows one to give multiple membership a more understandable meaning in the common situations when every single node is likely to belong to more than one cluster but each node-to-node relation can be explained as co-affiliation to some community (like in the well-known model of affiliation networks [14]). Of course, even in the latter situation co-affiliation can be due to many reasons (co-affiliation to many communities), one reason usually prevails.

The usage of line graphs to model link-clustering is especially promoted by Evans and Lambiotte [8] (who also take into consideration notions of weighting that deal with the problem of over-representing high-degree nodes), but they exploit the undirected version of line graphs instead of the directed one [10], and they do not distinguish between triangular and non-triangular arcs. It should be noted that the roles of (open and closed) triangles in social networks is well known and studied in the realm of SNA, under the name of *triads* [25].

As explained, our technique relies on some external node-clustering algorithm that uses a weighted version of $L(G)$, with the hope that triangular random walks highlight clear cuts between communities as they should. To test our hypothesis, we obviously need a clustering algorithm that can handle large weighted directed graphs; we tried three different clustering algorithms which satisfy our requirements and are considered the state of the art for massive complex networks: clustering via Potts' model as proposed in [19], the hierarchical Infomap algorithm presented in [20] and the Louvain method [3]. In our tests the latter proved to be the fastest among these candidates and produces also the best results in term of accuracy, so we will adopt it in our experiments. Actually, however, all the tested methods improve their performance on the versions of $L(G)$ that were weighted according to our criterion.

VI. EXPERIMENTS

The experiments that we are going to describe have been run using public datasets and relying heavily on the WebGraph [6] framework (in particular, the line-graph transformation was implemented as a part of it). The remaining tools are available as "Satellite Software" in the http://law.dsi.unimi.it/ website. In most of the experiments, we shall need a way to evaluate the clustering quality. More precisely, we suppose to be given a graph G with a measure of similarity σ between its arcs. The output of an arc-clustering algorithm is going to be a labelling function λ providing a label for every arc xy of the input graph. To evaluate the quality of the given arc-clustering λ with respect to the similarity σ, we shall use a variant of the Probabilistic Rand Index (PRI) [23]:

$$\mathrm{PRI}(\lambda,\sigma) = \sum_{\lambda(xy)=\lambda(x'y')} \sigma(xy,x'y') - \sum_{\lambda(xy)\neq\lambda(x'y')} \sigma(xy,x'y').$$

The cost of evaluating this quantity is prohibitive (quadratic in the number of arcs), thus we shall instead estimate its value by sampling pairs of arcs $\{xy, x'y'\}$ according to the following criteria: (i) the two arcs xy and $x'y'$ are sampled uniformly

at random (PRI_u); (ii) a node $x = x'$ is chosen uniformly at random, and we select two of its successors y and y' again at random (PRI_n); (iii) a node $x = x'$ is chosen at random proportionally to its degree, and we select two of its successors y and y' at random (PRI_d). While the first is an unbiased estimator of the PRI, the latter two aim at providing a more fine-grained understanding of the local quality of the clustering obtained. PRI is a quality measure that indirectly takes the number of clusters into account: an excessive fragmentation, for example, will produce bad PRI values, because similar arcs that are put in different clusters contribute negatively to the score. Nonetheless, we will also discuss the number of clusters obtained in our experiments.

b) Parameter tuning: For this set of experiments, we worked on the DBLP graph[7]; The DBLP graph is a scientific collaboration network where each vertex represents a scientist and two vertices are connected if they have worked together on an article. The current version (July 2011) of the DBLP dataset contains 986 324 authors and 2 684 847 publications, giving rise to 3 353 618 co-authorship edges. This network corresponds to the typical situation in which every author can belong to more than one scientific community (because typically, during their life, scientists work on many different and often scarcely related topics), but collaborations usually correspond to a specific topic. Based on this interpretation, we labelled each edge of DBLP with the concatenation of all titles of the co-authored papers, and the similarity between two edges is computed as the cosine distance between the corresponding term vectors (we normalized the words though a Porter's stemmer and used TF-IDF [1] for term weighting); this measure of similarity σ between edges is our ground truth.

In this experiment we used the weights v_T of (2) computed with different values of α and β to see how they impact on the quality of the clustering obtained with respect to similarity; we used ALP as a clustering algorithm, but in our experiments it seems that parameter can be tuned pretty much independently from the clustering algorithm employed. Most probably, it depends instead from the type of social network considered (e.g., as observed, whether triangles should be promoted or demoted); in all the graphs we are using here, however, the behavior was the same.

In Figure 4 we show the values of PRI_u for different combinations of α and β; we did a similar evaluation for PRI_n, PRI_d and for the number of communities obtained (the corresponding graphs are not shown).

- For $\alpha = 0$, the weights v_T of (2) become constant and the behavior of the clustering algorithm degrades (for the sake of readability, this is not shown in the figure);
- As long as $\alpha > 0$, its value does not seem to impact much on the local quality measures (PRI_n, PRI_d) but the overall quality PRI_u decreases for large α's: our interpretation for this behavior is that larger values of α produce a more fragmented clustering (as also witnessed by the number of communities obtained) because infrequent teleporting reduces transitivity.

As far as β is concerned, small values of β (i.e., more importance to triangles) produce the best results. As a rule of thumb, we think that α should be taken small (in the remaining experiments, we set $\alpha = 0.1$) at least for sparse networks; on denser graphs, larger values of α can be a better option to avoid that few communities flood all the arcs. As for β, we used $\beta = 0.01$ in our experiments, but the actual value should be adapted to the specific network under examination, as already discussed.

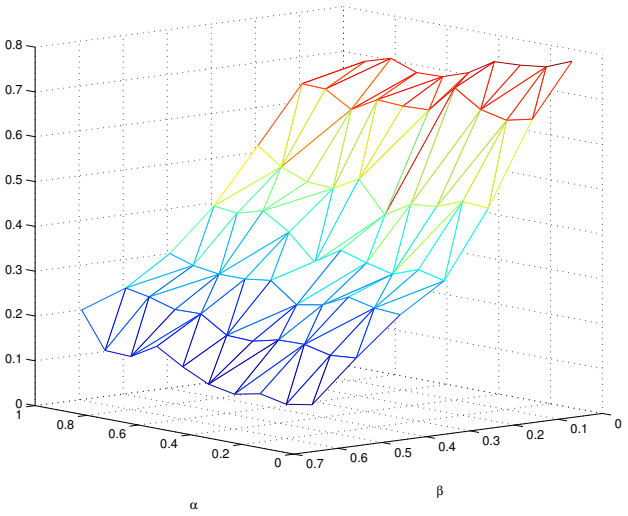


Fig. 4. PRI_u computed on DBLP (using the weights v_T of (2) and ALP for clustering) as a function of α and β.

c) Quality: We then faced the problem of directly evaluating the clustering quality, for the values of the parameters determined above ($\alpha = 0.1$ and $\beta = 0.01$). We performed our experiments on DBLP and on the Hollywood graph: the latter was obtained from the Internet Movie Database[8]; this undirected graph has, in its current version (July 2011), 2 180 759 nodes (actors and actresses) and 114 492 816 edges corresponding to having acted together in some movie. Here the edge xy is labelled with the multiset of directors that directed the movies co-acted by x and y, with the interpretation that a specific actor may have worked in many different movies, but directors tend often to collaborate with the same set of "trusted" actors. Similarity between arcs is once again computed using TF-IDF (here, the vocabulary is made by director IDs); the idea, this time, is to individuate the "clans" that typically pop up in the film industry around the figure of most directors. Note that, again, this idea would not fit with node clustering (because an actor is often part of more clans, but typically co-actorship individuates a clan in a quite specific way).

For this set of experiments, and for each of the two networks, we clustered the arcs in various ways (see below).

[7]http://www.informatik.uni-trier.de/~ley/db/.

[8]http://www.imdb.com/.

We considered the following combinations:

- we tried our weighting schemes w_T and v_T of (1) and (2), and for comparison the standard random surfer weights w_S and v_S (see Section III), as well as the unweighted version;
- for each weighting scheme above, we used two clustering algorithms: ALP (Section IV), that is fed directly with G (and computes $L(G)$ and its weights only implicitly) and the Louvain [3] algorithm, that is given the weighted version of $L(G)$ instead ([3] clusters the nodes of an arc-weighted graph);
- as baseline, we tried to cluster the arcs using the system proposed by [8] (that works on the undirected version of the link graph) and *LINK*, a link clustering technique proposed in [27][9]; both algorithms are specifically aimed at arc-clustering so they are the natural competitors of our method; unfortunately (as better explained below) we could run them only on the smallest of the two datasets, because of their lack of scalability;
- finally, as further baseline, we tried to cluster the arcs indirectly, through some of the best node clustering techniques; we transform a node clustering into an arc clustering with the following strategy: since a node clustering algorithm produces a labeling function $f : V_G \to \mathbb{N}$, we map each arc xy to the pair $(f(x), f(y)) \in \mathbb{N}^2$, and use the latter as arc label. If the original graph is symmetric, we can forget about the order of labels and assign an unique identifier to each unordered pair of labels.

The results obtained for DBLP [10] are shown in Table II, along with the computation time[11]: when using the Louvain [3] algorithm, we highlight the pre-computation time required to produce the weighted line graph to be fed to the algorithm; note also that for the PageRank-based weights v_T, there is some pre-computation time needed to obtain the PageRank vector (this is true also for ALP). As for Hollywood, the only arc-clustering method that can be applied is ALP and the results obtained are also shown in Table II—building explicitly the line graph is out of question and anyway it would be far too large to be handled by (the current implementation of) [3]; hence, our only baseline is Louvain run on the base graph G (we did not get any result from Infomap on the base graph, and we decided to stop it after 60h). Some comments are in order:

- Our weighting schemes aim at capturing local communities more than global ones, and indeed the local measures of quality (PRI_n and PRI_d) we obtain outperform significantly all other approaches; the best competitors, that still do not quite reach the same results, are Evans et al. [8] and LINK [27]. Both, however, do a rather poor job when the results are considered globally, but for opposite reasons: [27] seems to fragment the communities too much (many of them constitute of a single arc), whereas [8] produces too few communities (putting together too many "dissimilar" arcs). Apparently this problem presents itself also when we use our weighting scheme with [3], whereas ALP is able to produce a more balanced output, giving good results even on a global scale.
- Comparing our results with all the node-oriented approaches, it seems clear that arc-communities have a much more distinct structure than node-communities in the networks we examined.
- As far as the difference between the two types of weights, the gain in using the arc-stationary state v_T instead of the simple triangular weights w_T is marginal; yet PageRank computation is so fast that the effort is anyway worth.

d) Karate club (revisited): To visually appreciate the results of our clustering technique, we tried it on the karate club dataset; we set $\alpha = 0.1$ as usual, but this time the density of the network suggests using a larger β than we did with the other graphs. Figure 5 shows the outcome obtained for $\beta = 0.2$ (smaller values of β tend to fragment the network too much). The algorithm finds 6 communities, but two of them (the red and green arcs) are definitely dominant and correspond largely to the edges between homogeneous members. The two second-largest communities, in blue and violet, are rather dense internally but poorly linked to the other nodes. For comparison, in Figure 5 you can see the same network clustered with LINK, that individuates 22 communities.

e) Clustering of the word association network: For this experiment, we considered the Free Word Association network [16]; this is a directed graph describing the results of an experiment of free word association performed by more than 6 000 participants in the United States: its nodes correspond to words and arcs represent a cue-target pair (the arc xy means that the word y was output by some of the participants based on the stimulus x). This graph contains 10 617 words and 71 176 associations (arcs). We used ALP and Louvain to cluster it according to our two schemes (as usual, we set $\alpha = 0.1$ and $\beta = 0.01$). For comparison, we considered also the communities found by Evans et al. [8] and by LINK [27] on the same graph. In this case we do not have any ground truth to compare to, hence our analysis can only be based on some preliminary observations.

The number of communities found by ALP is 7 070 with w_T and 7 221 with v_T, showing that the use of PageRank tends in this case to obtain slightly smaller communities (the average size passes from 10.06 to 9.86). As for the other methods, [8] produces only 33 huge communities (the average size is 2 157), whereas [27] fragments the graph into 43 182 communities (the average size is 1.65).

An interesting observation is that of the 8 384 reciprocal arcs (an arc xy is reciprocal if also yx is an arc, the 11.8% of

[9]We used the LINK Python implementation that automatically optimizes its parameters. We also experimented with the software described in [12], but could not have it work on networks of more than about 100 nodes.

[10]All tests on DBLP were run only on the giant component of the graph because some of the baseline algorithms (in particular, LINK) requires the input graph to be connected; we verified, however, that the quality obtained by ALP is consistently the same even outside of the giant component.

[11]All experiments were performed on a Linux server equipped with Intel Xeon X5660 CPUs (2.80 GHz, 12 MB cache size) for overall 24 cores and 128 GB of RAM.

			clusters	PRI_u	PRI_n	PRI_d	computing time
DBLP	ALP (Section IV)	$\mathbf{v_T}$	613 203	**0.74**	**0.71**	**0.75**	1s+32s
		$\mathbf{w_T}$	592 562	**0.72**	**0.75**	**0.75**	32s
		v_S	48 025	0.02	0.16	0.18	24s
		w_S	38 498	0.02	0.08	0.03	22s
		-	38 498	0.02	0.08	0.03	22s
	Louvain [3]	$\mathbf{v_T}$	1 493	**0.01**	**0.69**	**0.53**	157s+337s
		$\mathbf{w_T}$	2 116	**0.02**	**0.71**	**0.53**	122s+334s
		v_S	230∗	0.01	0.44	0.39	137s+943s
		w_S	232	0.01	0.43	0.39	114s+914s
		-	250	0.01	0.16	0.15	92s+224s
	Evans et al. [8]	-	200	0.01	0.58	0.44	46min
	LINK [27]	-	1 415 245	0.28	0.31	0.51	50h
	Infomap [20]	-	62 680	0.05	0.27	0.29	874s
	Louvain (on G) [3]	-	6 442	0.01	0.28	0.28	13s
Hollywood	ALP (Section IV)	$\mathbf{v_T}$	383 780	**0.80**	**0.78**	**0.56**	1h+16h
		$\mathbf{w_T}$	424 094	**0.77**	**0.71**	**0.48**	13h
		v_S	255 247	0.00	0.03	0.03	3h
		w_S	277 859	0.00	0.02	0.01	3h
		-	277 859	0.00	0.02	0.01	3h
	Infomap [20]	-	-	-	-	-	> 60h
	Louvain (on G) [3]	-	23 807	0.01	0.18	0.19	242s

TABLE II

CLUSTERING QUALITY OBTAINED USING DIFFERENT TECHNIQUES ON THE DBLP AND HOLLYWOOD GRAPHS (IN BOLDFACE, THE TWO TRIANGULAR WEIGHTS SUGGESTED IN THIS PAPER, USING $\alpha = 0.1$ AND $\beta = 0.01$). THE UPPER GROUP REFERS TO THE APPLICATION OF THE ALP OR LOUVAIN ALGORITHM TO VARIOUS (WEIGHTED OR UNWEIGHTED) VERSIONS OF $L(G)$ (IN THE CASE OF LOUVAIN, THE LINE GRAPH MUST BE EXPLICITLY BUILT); THE MIDDLE GROUP CONSISTS OF ALGORITHMS THAT PRODUCE AN ARC-CLUSTERING ON G; THE BOTTOM GROUP, INSTEAD, PRODUCE A NODE-CLUSTERING ON G, THAT WE INTERPRET AS AN ARC-CLUSTERING.

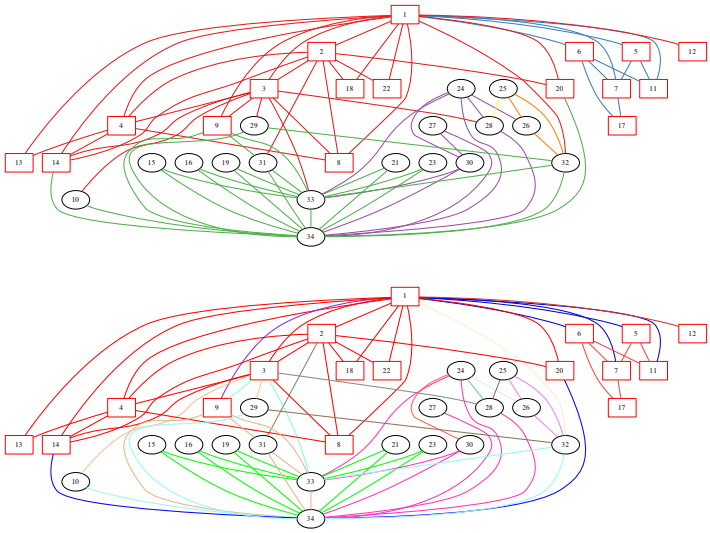


Fig. 5. Clustering of the karate club dataset: (top) using triangular weights (v_T) and the ALP clustering algorithm, (bottom) using LINK [27].

the arcs are such in this graph), only a minority are assigned by ALP the same label in the two directions (3 038 for w_T, 3 028 for v_T): this witnesses the fact that ALP does not behave "as if" the graph was symmetric.

On a purely anecdotal base, we present in Figure 6 the subgraph induced by the word "KEYBOARD" and its successors, as it is clustered by Louvain with v_T (top) and by Evans et al. [8] (bottom); note that the algorithm used is actually the same, so the difference is only in the weighting scheme. Although there is clearly a group of successors that are related to music and another one that is related to computers, [8] puts all arcs going out of "KEYBOARD" in the same community (even if the community of computer-related word is in fact recognized, because the internal arcs connecting the three words "COMPUTER", "TYPEWRITER" and "TYPE" have a different colour than the other ones). With our weighting scheme the arcs going toward the music group are clearly separated from the other.

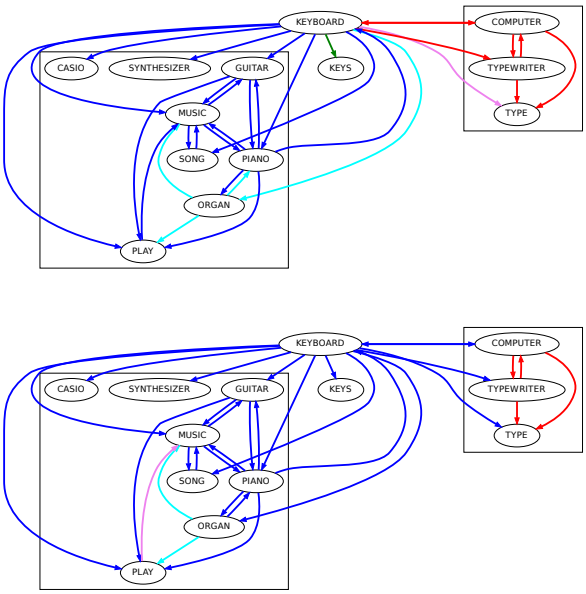


Fig. 6. Clustering of the word association network (subgraph around "KEYBOARD"): (top) using Louvain with v_T as weighting scheme, (bottom) using Evans et al. [8].

VII. Conclusions and Future Work

In this paper, we introduced a new kind of random process that helps in singling out arc communities in social networks; this can be seen as a Markov chain on the line graph whose arc-stationary state contains a big deal of information on the communities, and can be fruitfully used to gain a more accurate and fine-grained resolution, at least at a local level. In our experiments, using this information ended up in producing more reasonable and significant clusters, with a limited computational cost. These results are preliminary but very encouraging; we also believe that the weights proposed here can be beneficial for other types of mining tasks. Such tasks can be made reasonably scalable by exploiting the possibility (here explored with ALP) of writing implicit versions of mining algorithms that work on the weighted line graph without having to build it explicitly.

Acknowledgements

We thank Hawoon Jeong, Youngdo Kim, Dario Malchiodi and Federico Pedersini for their help in preparing the paper.

References

[1] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[2] Jeffrey Baumes, Mark K. Goldberg, Mukkai S. Krishnamoorthy, Malik Magdon-Ismail, and Nathan Preston. Finding communities by clustering a graph into overlapping subgraphs. In *IADIS AC'05*, pages 97–104, 2005.

[3] V.D. Blondel, J.L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008:P10008, 2008.

[4] Paolo Boldi, Violetta Lonati, Massimo Santini, and Sebastiano Vigna. Graph fibrations, graph isomorphism, and PageRank. *RAIRO Inform. Théor.*, 40:227–253, 2006.

[5] Paolo Boldi, Marco Rosa, Massimo Santini, and Sebastiano Vigna. Layered label propagation: A multiresolution coordinate-free ordering for compressing social networks. In Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar, M. P. Ravindra, Elisa Bertino, and Ravi Kumar, editors, *Proceedings of the 20th international conference on World Wide Web*, pages 587–596. ACM, 2011.

[6] Paolo Boldi and Sebastiano Vigna. The WebGraph framework I: Compression techniques. In *Proc. of the Thirteenth International World Wide Web Conference (WWW 2004)*, pages 595–601, Manhattan, USA, 2004. ACM Press.

[7] Gianna M. Del Corso, Antonio Gullì, and Francesco Romani. Fast pagerank computation via a sparse linear system. *Internet Mathematics*, 2:118–130, 2004.

[8] T. S. Evans and R. Lambiotte. Line graphs, link partitions, and overlapping communities. *Phys. Rev. E*, 80(1):016105, Jul 2009.

[9] Santo Fortunato and Claudio Castellano. Community structure in graphs. In Robert A. Meyers, editor, *Encyclopedia of Complexity and Systems Science*, pages 1141–1163. Springer, 2009.

[10] R. L. Hemminger and L. W. Beineke. Line graphs and line digraphs. In L. W. Beineke and R. J. Wilson, editors, *Selected Topics in Graph Theory*, pages 271–305. Academic Press Inc., 1978.

[11] Chen Jianbin, Fang Deying, and Shi Tong. A graph partition-based soft clustering algorithm. In *Proceedings of the 2008 Second International Symposium on Intelligent Information Technology Application - Volume 02*, pages 572–577, Washington, DC, USA, 2008. IEEE Computer Society.

[12] Youngdo Kim and Hawoong Jeong. The map equation for link community. *CoRR*, abs/1105.0257, 2011.

[13] Amy N. Langville and Carl D. Meyer. Deeper inside PageRank. *Internet Mathematics*, 1(3):355–400, 2004.

[14] Silvio Lattanzi and D. Sivakumar. Affiliation networks. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, STOC '09, pages 427–434, New York, NY, USA, 2009. ACM.

[15] Bo Long, Mark Zhang, Philip S. Yu, and Tianbing Xu. Clustering on complex graphs. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*, pages 659–664. AAAI Press, 2008.

[16] D. L. Nelson, C. L. McEvoy, and T. A. Schreiber. The university of south florida word association, rhyme, and word fragment norms. http://www.usf.edu/FreeAssociation/, 1998.

[17] Gergely Palla, Illes J. Farkas, Peter Pollner, Imre Derenyi, and Tamas Vicsek. Directed network modules. *New J.Phys.*, 9:186, 2007.

[18] Usha N. Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 76(3), 2007.

[19] Peter Ronhovde and Zohar Nussinov. Local resolution-limit-free potts model for community detection. *Phys. Rev. E*, 81(4):046114, Apr 2010.

[20] Martin Rosvall and Carl T. Bergstrom. Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PLoS ONE*, 6(4):e18209, 04 2011.

[21] Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

[22] E. Seneta. *Non-negative matrices and Markov chains*. Springer–Verlag, New York, 1981.

[23] Ranjith Unnikrishnan and Martial Hebert. Measures of similarity. In *7th IEEE Workshop on Applications of Computer Vision / IEEE Workshop on Motion and Video Computing (WACV/MOTION 2005)*, pages 394–400. IEEE Computer Society, 2005.

[24] Sebastiano Vigna. Spectral ranking, 2009.

[25] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*. Cambridge Univ Press, 1994.

[26] Wenpu Xing and Ali Ghorbani. Weighted pagerank algorithm. *Communication Networks and Services Research, Annual Conference on*, 0:305–314, 2004.

[27] Sune Lehmann Yong-Yeol Ahn, James P. Bagrow. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, August 2010.

[28] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.

Robustness of Social Networks: Comparative Results Based on Distance Distributions

Paolo Boldi Marco Rosa Sebastiano Vigna
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano, Italia

Abstract

Given a social network, which of its nodes have a stronger impact in determining its structure? More formally: which node-removal order has the greatest impact on the network structure? We approach this well-known problem for the first time in a setting that combines both web graphs and social networks, using datasets that are orders of magnitude larger than those appearing in the previous literature, thanks to some recently developed algorithms and software tools that make it possible to approximate accurately the number of reachable pairs and the distribution of distances in a graph. Our experiments highlight deep differences in the structure of social networks and web graphs, show significant limitations of previous experimental results, and at the same time reveal *clustering by label propagation* as a new and very effective way of locating nodes that are important from a structural viewpoint.

1 Introduction

In the last years, there has been an ever-increasing research activity in the study of real-world complex networks [WF94] (the world-wide web, the Internet autonomous-systems graph, coauthorship graphs, phone call graphs, email graphs and biological networks, to cite a few). These networks, typically generated directly or indirectly by human activity and interaction, appear in a large variety of contexts and often exhibit a surprisingly similar structure. One of the most important notions that researchers have been trying to capture is "node centrality": ideally, every node (often representing an individual) has some degree of influence or importance within the social domain under consideration, and one expects such importance to be reflected in the structure of the social network; centrality is a quantitative measure that aims at revealing the importance of a node.

Among the types of centrality that have been considered in the literature (see [Bor05] for a good survey), many have to do with shortest paths between nodes; for example, the *betweenness centrality* of a node v is the sum, over all pairs of nodes x and y, of the fraction of shortest paths from x to y passing through v. The role played by shortest paths is justified by one of the most well known features of complex networks, the so-called small-world phenomenon.

A small-world network [CH10] is a graph where the average distance between nodes is logarithmic in the size of the network, whereas the clustering coefficient is large (that is, neighbourhoods tend to be denser) than in a random Erdős-Rényi graph with the same size and average distance.[1] Here, and in the following, by "distance" we mean the length of the shortest path between two nodes. The fact that social networks (either electronically mediated or not) exhibit the small-world property is known at least since Milgram's famous experiment [Mil67] and is arguably the most popular of all features of complex networks.

[1] The reader might find this definition a bit vague, and some variants are often spotted in the literature: this is a general problem, also highlighted recently in [LADW05].

Based on the above observation that the small-world property is by far the most crucial of all the features that social networks exhibit, it is quite natural to consider centrality measures that are based on node distance, like betweenness. On the other hand, albeit interesting and profound, such measures are often computationally too expensive to be actually computed on real-world graphs; for example, the best known algorithm to compute betweenness centrality [Bra01] takes time $O(nm)$ and requires space for $O(n + m)$ integers (where n is the number of nodes and m is the number of arcs): both bounds are infeasible for large networks, where typically $n \approx 10^9$ and $m \approx 10^{11}$. For this reason, in most cases other strictly local measures of centrality are usually preferred (e.g., degree centrality).

One of the ideas that have emerged in the literature is that node centrality can be evaluated based on how much the removal of the node "disrupts" the graph structure [AJB00]. This idea provides also a notion of robustness of the network: if removing few nodes has no noticeable impact, then the network structure is clearly robust in a very strong sense. On the other hand, a node-removal strategy that quickly affects the distribution of distances probably reflects an importance order of the nodes.

Previous literature has used mainly the diameter or some analogous measure to establish whether the network structure changed. Recently, though, there have been some successful attempts to produce reliable estimates of the *neighbourhood function* of very large graphs [PGF02, BRV11a]; an immediate application of these approximate algorithms is the computation of the number of *reachable pairs* of the graph (the number of pairs $\langle x, y \rangle$ such there is a directed path from x to y) and its *distance distribution* (the distance distribution of a graph is a discrete distribution that gives, for every t, the fraction of pairs of nodes that are at distance t). From this data, a number of existing measures can be computed quickly and accurately, and new one can be conceived.

We thus consider a certain ordering of the nodes of a graph (that is supposed to represent their "importance" or "centrality"). We remove nodes (and of course their incident arcs) following this order, until a certain percentage ϑ of the arcs have been deleted[2]; finally, we compare the number of reachable pairs and distance distribution of the new graph with the original one. The chosen ordering is considered to be a reliable measure of centrality if the measured difference increases rapidly with ϑ (i.e., it is sufficient to delete a small fraction of important nodes to change the structure of the graph).

In this work, we applied the described approach to a number of complex networks, considering different orderings, and obtained the following results:

- In all complex networks we considered, the removal of a limited fraction of randomly chosen nodes does not change the distance distribution significantly, confirming previous results.
- We test strategies based on PageRank and on clustering (see Section 4.1 for more information about this), and show that they (in particular, the latter) disrupt quickly the structure of a web graph.
- Maybe surprisingly, none of the above strategies seem to have an impact when applied to social networks other than web graphs. This is yet another example of a profound structural difference between web graphs and social networks,[3] on the same line as those discussed in [BRV11a] and [CKL+09]. This observation, in particular, suggests that social networks tend to be much more robust and cohesive than web graphs, at least as far as distances are concerned, and that "scale-free" models, which are currently proposed for both type of networks, do not to capture this important difference.

[2] Observe that we delete nodes but count the percentage of arcs removed, and not of nodes: this choice is justified by the fact that otherwise node orderings that put large-degree nodes first would certainly be considered (unfairly) more disruptive.

[3] We remark that several proposals have been made to find features that highlight such structural differences in a computationwise-feasible way (e.g., assortative mixing [NP03]), but all instances we are aware of have been questioned by the subsequent literature, so no clear-cut results are known as yet.

2 Related work

The idea of grasping information about the structure of a network by repeatedly removing nodes out of it is not new: Albert, Jeong and Barabási [AJB00] study experimentally the variation of the diameter on two different models of *undirected* random graphs when nodes are removed either randomly or in "connectedness order" and report different behaviours. They also perform tests on some small real data set, and we will compare their results with ours in Section 6.

More recently, node-centrality measures that look at how some graph invariant changes when some vertices or edges are deleted (sometimes called "vitality" [BE05] or "induced" measures) have been studied for example in [Bor06] (identifying nodes that maximally disconnect the network) or in [BCK06] (related to the uncertainty of data).

Donato, Leonard, Millozzi and Tsaparas [DLMT08] study how the size of the giant component changes when nodes of high indegree or outdegree are removed from the graph. While this is an interesting measure, it does not provide information about what happens outside the component. They develop a library for semi-external visits that make it possible to compute in an exact way the strongly connected components on large graphs.

Finally, Fogaras [Fog03] considers how the *harmonic diameter*[4] (the harmonic mean of the distances) changes as nodes are deleted from a small (less than one million node) snapshot of the `.ie` domain, reporting a large increase (100%) when as little as 1000 nodes with high PageRank are removed. The harmonic diameter is estimated by a small number of visits, however, which gives no statistical guarantee on the accuracy of the results.

Our study is very different. First of all, we use graphs that are two orders of magnitude larger than those considered in [AJB00] or [Fog03]; moreover, we study the impact of node removal on the whole spectrum of distances. Second, we apply removal procedures to large social networks (previous literature used only web or Internet graphs), and the striking difference in behaviour shows that "scale-free" models fail to capture essential differences between these kind of networks and web graphs. Third, we document in a reproducible way all our experiments, which have provable statistical accuracy.

3 Computing the distance distribution

Given a directed graph G, its *neighbourhood function* $N_G(t)$ returns for each $t \in \mathbf{N}$ the number of pairs of nodes $\langle x, y \rangle$ such that y is reachable from x in no more than t steps. From the neighbourhood function, several interesting features of a graph can be estimated, and in this paper we are especially interested in the *distance distribution* of the graph G , represented by the cumulative distribution function $H_G(t)$, which returns the fraction of reachable pairs at distance at most t, that is, $H_G(t) = N_G(t)/\max_t N_G(t)$. The corresponding probability density function will be denoted by $h_G(-)$.

Recently, HyperANF [BRV11a] emerged as an evolution of the ANF tool [PGF02]. HyperANF can compute for the first time in a few hours the neighbourhood function of graphs with billions of nodes with a small error and good confidence using a standard workstation. The free availability of HyperANF opens new and interesting ways to study large graphs, of which this paper is an example.

4 Removal strategies and their analysis

In the previous section, we discussed how we can effectively approximate the distance distribution of a given graph G; we shall use such a distribution as the graph structural property of interest.

[4] Actually, the notion had been introduced before by Marchiori and Latora and named *connectivity length* [ML00], but we find the name "harmonic diameter" much more insightful.

Consider now a given total order $\prec$ on the nodes of G; we think of $\prec$ as a removal strategy in the following sense: when we want to remove ϑm arcs, we start removing the $\prec$-largest node (and its incident arcs), go on removing the second-$\prec$-largest node etc. and stop as soon as $\geq \vartheta m$ arcs have been removed. The resulting graph will be denoted by $G(\prec, \vartheta)$. Of course, $G(\prec, 0) = G$ whereas $G(\prec, 1)$ is the empty graph. We are interested in applying some measure of *divergence*[5] between the distribution H_G and the distribution $H_{G(\prec,\vartheta)}$. By looking at the divergence when ϑ varies, we can judge the ability of $\prec$ to identify nodes that will disrupt the network.

4.1 Some removal strategies

We considered several different strategies for removing nodes from a graph. Some of them embody actually significant knowledge about the structure of the graph, whereas others are very simple (or even independent of the graph) and will be used as baseline. Some of them have been used in the previous literature, and will be useful to compare our results.

As a first observation, some strategies requires a symmetric graph (a.k.a., undirected). In this case, we symmetrise the graph by adding the missing arcs[6].

The second obvious observation is that some strategies might depend on available metadata (e.g., URLs for web graphs) and might not make sense for all graphs.

Random. No strategy: we pick random nodes and remove them from the graph. It is important to test against this "nonstrategy" as we can show that the phenomena we observe are due to the peculiar choice of nodes involved, and not to some generic property of the graph.

Largest-degree first. We remove nodes in decreasing (out)degree order. This strategy is an obvious baseline, as *degree centrality* is the first shot at centrality in a network.

Near-Root. In web graphs, we can assume that nodes that are roots of web sites and their (quasi-)immediate successors (e.g., pages linked by the root) are most important in establishing the distance distribution, as people tend to link higher levels of web sites. This strategy removes essentially first root nodes, then the nodes that are children of a root on, and so on.

PageRank. PageRank [PBMW98] is an well-known algorithm that assigns ranks to nodes using a Markov chain based on the structure of the graph. It has been designed as an improvement over degree centrality, because nodes with high degree which however are connected to nodes of low rank will have a rather low rank, too (the definition is indeed recursive). There is a vast body of literature on the subject: see [BSV09, LM04] and the references therein.

Label propagation. Label propagation [RAK07] is a powerful technique for clustering symmetric graphs. Each node has a label (initially, the node number itself) and through a number of rounds each node changes its label by taking the label of the majority of its neighbours. At the end, node labels are used as cluster identifiers. Our removal strategy picks first, for each cluster in decreasing size order, the node with the highest number of neighbours in other clusters: intuitively, it is a representative of a set of tightly connected nodes (the cluster) which however has a very significant connection with the outside world (the other clusters) and thus we expect that its removal should seriously disrupt the distance distribution. Once we have removed all such nodes, we proceed again, cluster by cluster, using the same criterion (thus picking the second node of each cluster that has more connection towards other clusters), and so on.

[5] We purposedly use the word "divergence" between distributions, instead of "distance", to avoid confusion with the notion of distance in a graph.

[6] It is mostly a matter of taste whether to use directed symmetric graphs or simple undirected graphs. In our case, since we have to cope with both directed and undirected graph, we prefer to speak of directed graphs that are symmetric, that is, for every arc $x \to y$ there is a symmetric arc $y \to x$.

4.2 Measures of divergence

Once we changed the structure of a graph by deleting some of its nodes (and arcs), there are several ways to measure whether the structure of the graph has significantly changed. The first, basic raw datum we consider is the *number of pairs of nodes that are still reachable* divided by *the number of pairs initially reachable*, expressed as a percentage. Then, to estimate the change of the distance distribution we considered the following possibilities (here P denotes the original distance distribution, and Q the distribution after node removal):

Relative average-distance change. This is somehow the simplest and most natural measure: how much has the average distance changed? We use the measure

$$\delta(P, Q) = \frac{\mu_Q}{\mu_P} - 1$$

where μ denotes the average; in other words, we measure how much the average value changed. This measure is non-symmetric, but it is of course easy to obtain $\delta(P, Q)$ from $\delta(Q, P)$.

Relative harmonic-diameter change. This measure is analogous to the relative average-distance change, but the average on distances is *harmonic* and *computed on all pairs*, that is:

$$\frac{n(n-1)}{\sum_{x \neq y} \frac{1}{d(x,y)}} = n(n-1) \Big/ \sum_{t>0} \frac{1}{t}(N_G(t) - N_G(t-1)),$$

where n is the number of nodes of the graph. This measure, used in [Fog03], combines reachability information, as unreachable pairs contribute zero to the sum. It is easily computable from the neighbourhood function, as shown above.

Kullback-Leibler divergence. This is a measure of *information gain*, in the sense that it gives the number of additional bits that are necessary to code samples drawn from P when using an optimal code for Q. Also this measure is non-symmetric, but there is no way obtain the divergence from P to Q given that from Q to P.

ℓ norms. A further alternative is given by viewing distance distributions as functions $\mathbf{N} \to [0\,..\,1]$ and measure their distance using some ℓ-norm, most notably ℓ_1 or ℓ_2. Such distances are of course symmetric.

We tested, with various graphs and removal strategies, how the choice of distribution divergence influences the interpretation of the results obtained. In Figure 1 we show this for a single web graph and a single strategy, but the outcomes agree on all the graphs and strategies tested: the interpretation is that all divergences agree, and for this reason we shall use the (simple) measure δ applied to the average distance in the experimental section. The advantage of δ over the other measures is that it is very easy to interpret; for example, if δ has value, say, 0.3 it means that node removal has increased the average distance by 30%. We also discuss δ applied to the harmonic diameter.

5 Experiments

For our experiments, we considered a number of networks with various sizes and characteristics; most of them are either web graphs or (directed or undirected) social graphs of some kind (note that for web graphs we can rely on the URLs as external source of information). More precisely, we used the following datasets:

- *Hollywood*: One of the most popular *undirected* social graphs, the graph of movie actors: vertices are actors, and two actors are joined by an edge whenever they appeared in a movie together.

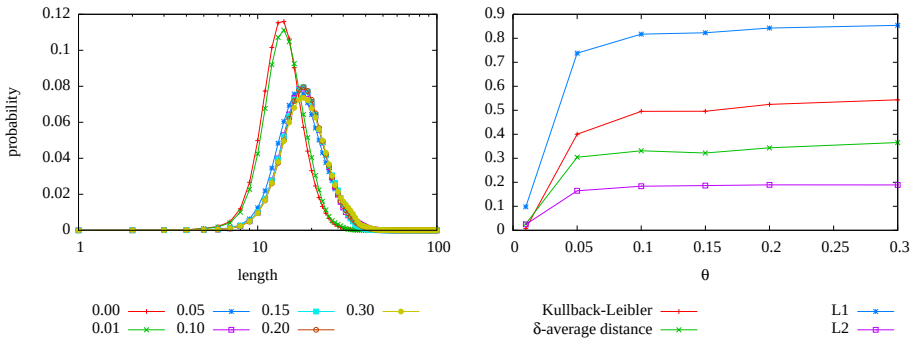


Figure 1: Testing various divergence measures on a web graph (a snapshot of the `.it` domain of 2004) and the near-root removal strategy. You can see how the distance distribution changes for different values of ϑ and the behaviour of divergence measures. We omitted to show the harmonic-diameter change to make the plot easier to read.

- *LiveJournal*: LiveJournal is a virtual community social site started in 1999: nodes are users and there is an arc from x to y if x registered y among his friends (it is not necessary to ask y permission, so the graph is *directed*). We considered the same 2008 snapshot of *LiveJournal* used in [CKL+09] for their experiments

- *Amazon*: This dataset describes similarity among books as reported by the Amazon store; more precisely the data was obtained in 2008 using the Amazon E-Commerce Service APIs using `SimilarityLookup` queries.

- *Enron*: This dataset was made public by the Federal Energy Regulatory Commission during its investigations: it is a partially anonymised corpus of e-mail messages exchanged by some Enron employees (mostly part of the senior management). We turned this dataset into a *directed* graph, whose nodes represent people and with an arc from x to y whenever y was the recipient of (at least) a message sent by x.

- For comparison, we considered two web graphs of different size: a 2004 snapshot of the `.it` domain ($\approx$ 40 million nodes), and a snapshot taken in May 2007 of the `.uk` domain ($\approx$ 100 million nodes).

We remark that all our graphs are available at the LAW web site.[7] HyperANF is available as free software at the WebGraph web site[8], and the class `RemoveHubs` that has been used to perform the experiments we describe is part of the LAW software.

We applied our removal strategies with different impact levels (e.g., percentage of removed arcs), namely 0.01, 0.05, 0.1, 0.15, 0.2 and 0.3. For each level we ran HyperANF at least seven times using 128 registers per counter: the percentage of reachable pair displayed in our tables has been obtained by averaging the neighbourhood functions obtained from the runs, with relative standard deviation smaller than 3.5% (e.g., the measure is within relative error 10.5% with 95% confidence). The starting number of reachable pairs is known with relative standard deviation smaller than 0.1%. The remaining derived measurements (average distances and harmonic diameters) have been computed separately on each run, and the resulting relative standard deviation is less than 4% for the average distance, and less than 20% for the harmonic diameter, except for about a dozen measurements, where

[7] `http://law.dsi.unimi.it/`. In particular, the graphs we used are the datasets named `hollywood-2009`, `ljournal-2008`, `amazon-2008`, `enron`, `it-2004` and `uk-2007-05`.

[8] `http://webgraph.dsi.unimi.it/`

it is less than 8.5% for the average distance, and less than 30% for the harmonic diameter.[9] Our tables and graphs slightly differs from those previously published [BRV11b] because we had time to generate more runs, and thus increase the precision of our results: some variation is also observed because of the relatively small number of runs (unavoidable, due to the large number of graphs to be analyzed).

6 Discussion

Table 1 and Figure 2 show that social networks suffer spectacularly less disconnection than web graphs when their nodes are removed using our strategies. Our most efficient removal strategy, label propagation, can disconnect almost all pairs of a web graph by removing 30% of the arcs, whereas it disconnects only about half (or less) of the pairs on social networks. This entirely different behaviour shows that web graphs have a path structure that passes through fundamental hubs.

Moreover, the average distance of the web graphs we consider increases by $50-80\%$ upon removal of 30% of the arcs, whereas in most social networks there is just an increase of a few percents (in any case, always less than 20%).[10]

Note that random removal can separate a good number of reachable pairs, but the increase in average distance is very marginal. This shows that considering both measures is important in evaluating removal strategies.

Of course, we cannot state that there is no strategy able to disrupt social networks as much as a web graph (simply because this strategy may be different from the ones that we considered), but the fact all strategies work very similarly in both cases (e.g., label propagation is by far the most disruptive strategy) suggests that the phenomenon is intrinsic.

There is a candidate easy explanation: shortest paths in web graphs pass frequently through home pages, which are linked more than other pages. But this explanation does not take into account the fact that clustering by label propagation is significantly more effective than the near-root removal strategy. Rather, it appears that there are fundamental hubs (not necessarily home pages) which act as shortcuts and through which a large number of shortest paths pass. Label propagation is able to identify such hubs, and their removal results in an almost disconnected graph and in a very significant increase in average distance.

These hubs are not necessarily of high outdegree: quite the opposite, rather, is true. The behaviour of web graphs under the largest-degree strategy is illuminating: we obtain the smallest reduction in reachable pairs and an almost unnoticeable change of the average distance, which means that nodes of high outdegree are not actually relevant for the global structure of the network.

Social networks are much more resistant to node removal. There is no strict clustering, nor definite hubs, that can be used to eliminate or elongate shortest paths. This is not surprising, as networks emerging from social interaction are much less engineered (there is no notion of "site" or "page hierarchy", for example) than web graphs.

The second important observation is that the removal strategies based on PageRank and label propagation are always the best (with the exception of the near-root strategy for `.uk`, which is better than PageRank). This suggests that label propagation is actually able to identify structurally important nodes in the graph—in fact, significantly better than any other method we tested.

Is the ranking provided by label propagation correlated to other rankings? Certainly not to the other rankings described in this paper, due to the different level of disruption it produces on the network. The closest ranking with similar behaviour is PageRank, but, for instance, Kendall's τ between PageRank and ranking by label propagation on the `.uk` dataset is ≈ -0.002 (complete uncorrelation).

[9] Unfortunately, estimating with precision the harmonic diameter is difficult due to the nonlinearity of its definition.

[10] We remark that in some cases the measure is negative or does not decrease monotonically. This is an artifact of the

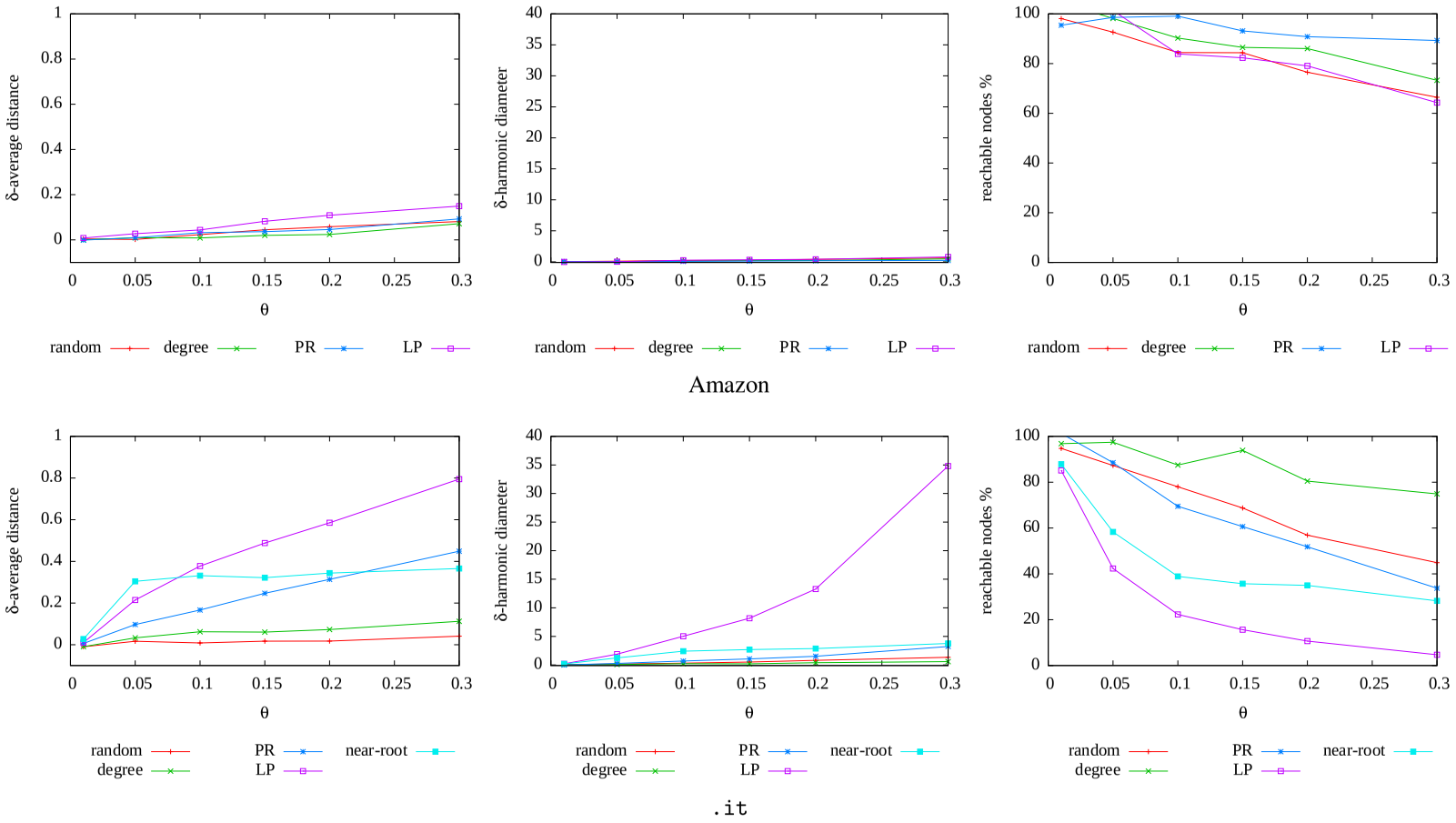


Figure 2: Typical behaviour of social networks (Amazon, upper) and web graphs (`.it`, lower) when a ϑ fraction of arcs is removed using various strategies. None of the proposed strategies completely disrupts the structure of social networks, but the effect of the label-propagation removal strategy on web graphs is very visible.

Graph	Strategy	0.01		0.05		0.1		0.15		0.2		0.3	
Amazon	Random	0.002	(98%)	0.000	(93%)	0.023	(84%)	0.045	(84%)	0.056	(77%)	0.081	(66%)
	Degree	0.000	(105%)	0.010	(98%)	0.008	(90%)	0.018	(87%)	0.023	(86%)	0.070	(73%)
	PR	−0.004	(95%)	0.008	(99%)	0.029	(99%)	0.034	(93%)	0.044	(91%)	0.092	(89%)
	LP	0.007	(104%)	0.025	(101%)	0.043	(84%)	0.081	(82%)	0.108	(79%)	0.149	(64%)
Enron	Random	0.007	(91%)	0.003	(84%)	−0.007	(75%)	−0.013	(65%)	0.003	(65%)	0.009	(57%)
	Degree	−0.010	(81%)	0.009	(73%)	0.035	(65%)	0.047	(61%)	0.062	(53%)	0.119	(43%)
	PR	0.001	(92%)	0.041	(72%)	0.051	(55%)	0.066	(38%)	0.129	(34%)	0.184	(25%)
	LP	−0.010	(90%)	−0.028	(76%)	−0.042	(70%)	−0.061	(52%)	−0.065	(52%)	−0.057	(40%)
Hollywood	Random	−0.005	(98%)	0.011	(101%)	0.005	(90%)	0.012	(86%)	−0.003	(73%)	0.014	(72%)
	Degree	0.003	(101%)	0.008	(106%)	0.002	(103%)	0.012	(94%)	0.020	(105%)	0.021	(92%)
	PR	0.007	(104%)	0.006	(97%)	0.019	(101%)	0.026	(99%)	0.027	(95%)	0.046	(94%)
	LP	−0.016	(91%)	−0.036	(76%)	−0.051	(63%)	−0.062	(52%)	−0.062	(48%)	−0.058	(43%)
LiveJournal	Random	−0.001	(97%)	−0.001	(95%)	0.004	(86%)	0.012	(87%)	0.023	(77%)	0.027	(66%)
	Degree	0.009	(104%)	0.018	(97%)	0.026	(94%)	0.037	(103%)	0.051	(99%)	0.074	(90%)
	PR	0.002	(98%)	0.023	(101%)	0.041	(96%)	0.055	(93%)	0.075	(97%)	0.111	(90%)
	LP	−0.004	(101%)	−0.021	(83%)	−0.022	(78%)	−0.034	(69%)	−0.026	(69%)	−0.041	(56%)
`.it`	Random	−0.012	(95%)	0.014	(87%)	0.007	(78%)	0.016	(69%)	0.017	(57%)	0.040	(45%)
	Degree	−0.011	(97%)	0.031	(97%)	0.062	(87%)	0.068	(94%)	0.080	(80%)	0.127	(75%)
	PR	0.005	(101%)	0.096	(89%)	0.164	(69%)	0.244	(61%)	0.308	(52%)	0.447	(34%)
	LP	0.010	(85%)	0.213	(42%)	0.378	(22%)	0.487	(16%)	0.583	(11%)	0.793	(5%)
	Near-Root	0.025	(88%)	0.301	(58%)	0.330	(39%)	0.320	(36%)	0.344	(35%)	0.365	(28%)
`.uk`	Random	0.001	(99%)	0.003	(82%)	0.033	(83%)	0.037	(74%)	0.056	(69%)	0.062	(49%)
	Degree	0.005	(101%)	0.011	(104%)	0.003	(97%)	−0.002	(94%)	0.013	(93%)	0.025	(98%)
	PR	0.023	(92%)	0.067	(80%)	0.100	(64%)	0.126	(55%)	0.175	(51%)	0.265	(38%)
	LP	0.051	(87%)	0.236	(39%)	0.272	(24%)	0.373	(18%)	0.439	(13%)	0.458	(6%)
	Near-Root	0.068	(80%)	0.244	(52%)	0.260	(48%)	0.261	(45%)	0.308	(45%)	0.278	(34%)

Table 1: For each graph and a sample of fractions of removed arcs we show the change in average distance (by the measure δ defined in Section 4.2) and the percentage of reachable pairs. PR stands for PageRank, and LP for label propagation.

Graph	Strategy	0.01		0.05		0.1		0.15		0.2		0.3	
Amazon	Random	0.036	(98%)	0.100	(93%)	0.215	(84%)	0.245	(84%)	0.397	(77%)	0.624	(66%)
	Degree	−0.033	(105%)	0.033	(98%)	0.121	(90%)	0.197	(87%)	0.204	(86%)	0.473	(73%)
	PR	0.057	(95%)	0.035	(99%)	0.077	(99%)	0.125	(93%)	0.172	(91%)	0.234	(89%)
	LP	−0.013	(104%)	0.028	(101%)	0.259	(84%)	0.321	(82%)	0.409	(79%)	0.795	(64%)
Enron	Random	0.132	(91%)	0.247	(84%)	0.397	(75%)	0.552	(65%)	0.572	(65%)	0.879	(57%)
	Degree	0.249	(81%)	0.435	(73%)	0.623	(65%)	0.800	(61%)	1.041	(53%)	1.703	(43%)
	PR	0.139	(92%)	0.472	(72%)	0.951	(55%)	1.758	(38%)	2.285	(34%)	3.741	(25%)
	LP	0.145	(90%)	0.311	(76%)	0.425	(70%)	0.833	(52%)	0.835	(52%)	1.388	(40%)
Hollywood	Random	0.032	(98%)	0.029	(101%)	0.125	(90%)	0.178	(86%)	0.373	(73%)	0.432	(72%)
	Degree	0.013	(101%)	−0.042	(106%)	−0.011	(103%)	0.087	(94%)	−0.014	(105%)	0.128	(92%)
	PR	−0.024	(104%)	0.055	(97%)	0.028	(101%)	0.049	(99%)	0.100	(95%)	0.138	(94%)
	LP	0.104	(91%)	0.281	(76%)	0.537	(63%)	0.814	(52%)	0.978	(48%)	1.256	(43%)
LiveJournal	Random	0.046	(97%)	0.059	(95%)	0.185	(86%)	0.162	(87%)	0.331	(77%)	0.587	(66%)
	Degree	−0.026	(104%)	0.057	(97%)	0.114	(94%)	0.018	(103%)	0.075	(99%)	0.203	(90%)
	PR	0.041	(98%)	0.027	(101%)	0.090	(96%)	0.162	(93%)	0.129	(97%)	0.261	(90%)
	LP	0.007	(101%)	0.200	(83%)	0.287	(78%)	0.413	(69%)	0.443	(69%)	0.745	(56%)
`.it`	Random	0.069	(95%)	0.173	(87%)	0.313	(78%)	0.503	(69%)	0.803	(57%)	1.345	(45%)
	Degree	0.033	(97%)	0.077	(97%)	0.220	(87%)	0.192	(94%)	0.389	(80%)	0.584	(75%)
	PR	−0.002	(101%)	0.268	(89%)	0.690	(69%)	1.057	(61%)	1.524	(52%)	3.221	(34%)
	LP	0.200	(85%)	1.885	(42%)	5.022	(22%)	8.178	(16%)	13.285	(11%)	34.809	(5%)
	Near-Root	0.186	(88%)	1.233	(58%)	2.415	(39%)	2.698	(36%)	2.867	(35%)	3.763	(28%)
`.uk`	Random	0.024	(99%)	0.232	(82%)	0.245	(83%)	0.416	(74%)	0.538	(69%)	1.158	(49%)
	Degree	−0.005	(101%)	−0.022	(104%)	0.046	(97%)	0.072	(94%)	0.092	(93%)	0.046	(98%)
	PR	0.122	(92%)	0.340	(80%)	0.716	(64%)	1.053	(55%)	1.320	(51%)	2.238	(38%)
	LP	0.216	(87%)	2.117	(39%)	4.152	(24%)	6.231	(18%)	9.229	(13%)	22.050	(6%)
	Near-Root	0.339	(80%)	1.327	(52%)	1.572	(48%)	1.712	(45%)	1.844	(45%)	2.663	(34%)

Table 2: For each graph and a sample of fractions of removed arcs we show the change in harmonic diameter (by the measure δ defined in Section 4.2) and the percentage of reachable pairs. PR stands for PageRank, and LP for label propagation.

It is interesting to compare our results against those in the previous literature. With respect to [AJB00], we test much larger networks. We can confirm that random removal is less effective that rank-based removal, but clearly the variation in diameter measured in [AJB00] has been made on a *symmetrised* version of the web graph. Symmetrisation destroys much of the structure of the network, and it is difficult to justify (you cannot navigate links backwards). We have evaluated our experiment using the variation in diameter instead of the variation in average distance (not shown here), but the results are definitely inconclusive. The behaviour is wildly different even between graphs of the same type, and shows no clear trend. This was expected, as the diameter is defined by a maximisation property, so it is very unstable.

We also evaluated the variation in harmonic diameter (see Table 2), to compare our results with those of [Fog03]. The harmonic diameter is very interesting, as it combines reachability and distance. The data confirm what we already stated: web graphs react to removal of 30% of their arcs by label propagation by increasing their harmonic diameter by an order of magnitude—something that does not happen with social networks. Table 2 is even more striking than Table 1 in showing that label propagation selects highly disruptive nodes in web graphs.

Our criterion for node elimination is a threshold on the number of *arcs* removed, rather than nodes, so it is not possible to compare our results with [Fog03] directly. However, for `.uk` PageRank at $\vartheta = 0.01$ removes 648 nodes, which produced in the `.ie` graph a relative increment of 100%, whereas we find 14%. This is to be expected, due to the very small size of the dataset used in [Fog03]: experience shows that connectedness phenomena in web graphs are very different in the "below ten million nodes" region. Nonetheless, the growth trend is visibile in both cases. However, the experiments in [Fog03] fail to detect both the disruptive behaviour at $\vartheta = .3$ and the striking difference in behaviour between largest-degree and PageRank strategy.

7 Conclusions and future work

We have explored experimentally the alterations of the distance distribution of some social networks and web graphs under different node-removal strategies. We have confirmed some of the experimental results that appeared in the literature, but at the same time shown some basic limitations of previous approaches. In particular, we have shown for the first time that there is a clear-cut structural difference between social networks and web graphs[11], and that it is important to test node-removal strategies until a significant fraction of the arcs have been removed.

Probably the most important conclusion is that "scale-free" models, which are currently proposed for both web graphs and social networks, do not to capture this important difference: for this reason, they can only make sense as long as they are adopted as baselines.

It might be argued that reachable pairs and distance distributions are too coarse as a feature. Nonetheless, we believe that they are the most immediate *global* feature that are approachable computationally. For instance, checking whether node removal alters the clustering coefficient would not be so interesting, because the clustering coefficient of each node depends only on the structure of the neighbourhood of each node. Thus, by removing first the nodes with high coefficient it would be trivial to make the clustering coefficient of the graph decrease quickly. Such trivial approaches cannot possibly work with reachable pairs or with distance distributions because they are properties that depend on the graph as a whole.

Finally, the efficacy of label propagation as a removal strategy suggests that it may be very interesting to study it as a form of *ranking*: an open question is whether it could be useful, for instance, as a query-independent ranking for information-retrieval applications.

---

probabilistic technique used to estimate the number of pairs—small relative errors are unavoidable.

[11] In this paper, like in all the other experimental research on the same topic, conclusions about social networks should be taken with a grain of salt, due to the heterogeneity of such networks and the lack of a large repertoire of examples.

References

[AJB00] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Error and attack tolerance of complex networks. *Nature*, 406:378–382, 2000.

[BCK06] Stephen P. Borgatti, Kathleen M. Carley, and David Krackhardt. On the robustness of centrality measures under conditions of imperfect data. *Social Networks*, 28(2):124–136, 2006.

[BE05] Ulrik Brandes and Thomas Erlebach. *Network Analysis: Methodological Foundations (Lecture Notes in Computer Science)*. Number 3418 in Lecture Notes in Computer Science. Springer-Verlag, 2005.

[Bor05] Stephen P. Borgatti. Centrality and network flow. *Social Networks*, 27(1):55–71, 2005.

[Bor06] Stephen P. Borgatti. Identifying sets of key players in a social network. *Comput. Math. Organ. Theory*, 12:21–34, April 2006.

[Bra01] Ulrik Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163–177, 2001.

[BRV11a] Paolo Boldi, Marco Rosa, and Sebastiano Vigna. HyperANF: Approximating the neighbourhood function of very large graphs on a budget. In Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar, M. P. Ravindra, Elisa Bertino, and Ravi Kumar, editors, *Proceedings of the 20th international conference on World Wide Web*, pages 625–634. ACM, 2011.

[BRV11b] Paolo Boldi, Marco Rosa, and Sebastiano Vigna. Robustness of social networks: Comparative results based on distance distributions. In *Social Informatics, Third International Conference, SocInfo 2011*, volume 6894 of *Lecture Notes in Computer Science*, pages 8–21. Springer, 2011.

[BSV09] Paolo Boldi, Massimo Santini, and Sebastiano Vigna. PageRank: Functional dependencies. *ACM Trans. Inf. Sys.*, 27(4):1–23, 2009.

[CH10] Reuven Cohen and Shlomo Havlin. *Complex Networks: Structure, Robustness and Function*. Cambridge University Press, 2010.

[CKL+09] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, Michael Mitzenmacher, Alessandro Panconesi, and Prabhakar Raghavan. On compressing social networks. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 219–228, New York, NY, USA, 2009. ACM.

[DLMT08] Debora Donato, Stefano Leonardi, Stefano Millozzi, and Panayiotis Tsaparas. Mining the inner structure of the web graph. *Journal of Physics A: Mathematical and Theoretical*, 41(22):224017, 2008.

[Fog03] Dániel Fogaras. Where to start browsing the web? In *Innovative Internet Community Systems, Third International Workshop, IICS 2003*, volume 2877 of *Lecture Notes in Computer Science*, pages 65–79. Springer, 2003.

[LADW05] Lun Li, David L. Alderson, John Doyle, and Walter Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Math.*, 2(4), 2005.

[LM04] Amy N. Langville and Carl D. Meyer. Deeper inside PageRank. *Internet Mathematics*, 1(3):355–400, 2004.

[Mil67] Stanley Milgram. The small world problem. *Psychology Today*, 2(1):60–67, 1967.

[ML00] Massimo Marchiori and Vito Latora. Harmony in the small-world. *Physica A: Statistical Mechanics and its Applications*, 285(3-4):539 – 546, 2000.

[NP03] Mark E. J. Newman and Juyong Park. Why social networks are different from other types of networks. *Phys. Rev. E*, 68(3):036122, 2003.

[PBMW98] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, Stanford University, Stanford, CA, USA, 1998.

[PGF02] Christopher R. Palmer, Phillip B. Gibbons, and Christos Faloutsos. Anf: a fast and scalable tool for data mining in massive graphs. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 81–90, New York, NY, USA, 2002. ACM.

[RAK07] Usha N. Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 76(3), 2007.

[WF94] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*. Cambridge Univ Press, 1994.

Injecting Uncertainty in Graphs for Identity Obfuscation

Paolo Boldi Francesco Bonchi Aristides Gionis Tamir Tassa

Università degli Studi
Milano, Italy
boldi@dsi.unimi.it

Yahoo! Research
Barcelona, Spain
{bonchi,gionis}@yahoo-inc.com

The Open University
Ra'anana, Israel
tamirta@openu.ac.il

ABSTRACT

Data collected nowadays by social-networking applications create fascinating opportunities for building novel services, as well as expanding our understanding about social structures and their dynamics. Unfortunately, publishing social-network graphs is considered an ill-advised practice due to privacy concerns. To alleviate this problem, several anonymization methods have been proposed, aiming at reducing the risk of a privacy breach on the published data, while still allowing to analyze them and draw relevant conclusions.

In this paper we introduce a new anonymization approach that is based on injecting *uncertainty* in social graphs and publishing the resulting *uncertain graphs*. While existing approaches obfuscate graph data by adding or removing edges entirely, we propose using a finer-grained perturbation that adds or removes edges *partially*: this way we can achieve the same desired level of obfuscation with smaller changes in the data, thus maintaining higher utility. Our experiments on real-world networks confirm that at the same level of identity obfuscation our method provides higher usefulness than existing randomized methods that publish standard graphs.

1. INTRODUCTION

Preserving the anonymity of individuals when publishing social-network data is a challenging problem that has recently attracted a lot of attention [2, 22]. The methods that have been proposed so far for anonymizing social graphs can be classified into three main categories: (1) methods that group vertices into super-vertices of size at least k, where k is the required level of anonymity; (2) methods that provide anonymity in the graph via deterministic edge additions or deletions; and (3) methods that add noise to the data in the form of random additions, deletions or switching of edges.

In this paper we introduce a new graph-anonymization method that does not fall in any of the above three categories. Our method injects uncertainty in the existence of the edges of the graph and publishes the resulting *uncertain graph*, that is, a graph where each edge e has an associated probability $\mathbf{p}(e)$ of being present. Injecting a limited amount of uncertainty in the data, in order to reach a desired level of identity obfuscation, is a natural approach [1]. For instance, the k-anonymity framework for relational data [25, 28] is typically based on injecting uncertainty by means of attribute generalization; for example, generalizing an exact numerical value to a range of values.

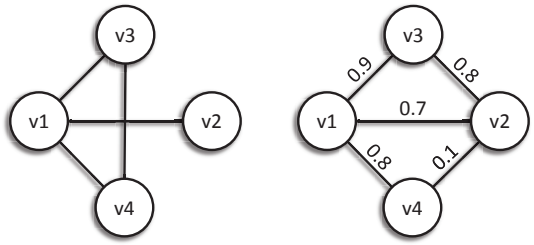


Figure 1: (a) A graph; (b) a possible obfuscation.

In the context of graph anonymization, our approach can be seen as a generalization of random-perturbation methods, which randomly delete existing edges and add non-existing edges [12]. From a probabilistic perspective, adding a non-existing edge e corresponds to changing its probability $\mathbf{p}(e)$ from 0 to 1, while removing an existing edge corresponds to changing its probability from 1 to 0. In our method, instead of considering only binary edge probabilities, we allow probabilities to take any value in $[0, 1]$, thus allowing for greater flexibility. The underlying intuition is that by using finer-grained perturbation operations, one can achieve the same desired level of obfuscation with smaller changes in the data, thus maintaining higher data utility.

An example of the proposed obfuscation method is shown in Figure 1: The graph (a) is the original graph that needs to be obfuscated; the published graph (b) is a possible obfuscation. While vertices v_1 and v_2 are connected in (a), in (b) they are connected with probability $\mathbf{p}(v_1, v_2) = 0.7$, representing a reduction of 0.3 in the certainty of existence of the edge (v_1, v_2). Vertices v_3 and v_4, which are connected in (a), are no longer connected in the published graph (b), i.e., $\mathbf{p}(v_3, v_4) = 0$. Vertices v_2 and v_3, which were not connected in (a), are connected with probability 0.8 in (b), corresponding to a partial creation of an edge.

A natural question that arises is how to query and analyze data that is published in the form of an uncertain graph. Hence, in order to prove the practical relevance of our proposal, not only we need to show that the uncertain graph maintains high utility, which we measure as similarity to the original graph in terms of characteristic properties, but also that the computation of these properties can be carried

Proceedings of the VLDB Endowment, Vol. 5, No. 11

out efficiently. An essential part of our discussion will be devoted to this. Fortunately, an increasing research effort was dedicated in recent years to the topic of querying and mining uncertain graphs [14, 15, 24, 36, 37, 38]: this body of research comes to our aid, providing evidence that useful analysis can be carried out on uncertain graphs.

In this work we achieve the following contributions:

- We introduce and formalize the idea of injecting uncertainty in graphs for identity obfuscation. In particular, we formally define the notion of (k, ε)-*obfuscation* for uncertain graphs (Section 3).
- We provide methods for assessing the level of obfuscation achieved by an uncertain graph with regards to the degree property (Section 4).
- We introduce our method for injecting uncertainty in a graph for (k, ε)-obfuscation (Section 5).
- In Section 6, we discuss several graph statistics and methods to compute them efficiently in uncertain graphs. These statistics are then used in Section 7 to assess the utility of the published uncertain graph.
- Our experimental assessment on three large real-world networks proves that at the same obfuscation levels, our method maintains higher data utility than existing random-perturbation methods.

In the next section we review the relevant literature, while in Section 8 we conclude the paper and suggest future work.

2. RELATED WORK

As we already mentioned, methods for anonymizing social networks can be broadly classified into three categories: generalization by means of clustering of vertices; deterministic alteration of the graph by edge additions or deletions; randomized alteration of the graph by addition, deletion or switching of edges.

In the first category, Hay et al. [10, 11] propose to generalize a network by clustering vertices and publishing the number of vertices in each partition together with the densities of edges within and across partitions. Campan and Truta [5] study the case in which vertices contain additional attributes, e.g., demographic information. They propose to cluster the vertices and reveal only the number of intra- and inter-cluster edges. The vertex properties are generalized in such a way that all vertices in the same cluster have the same generalized representation. Tassa and Cohen [29] consider a similar setting and propose a sequential clustering algorithm that issues anonymized graphs with higher utility than those issued by the algorithm of Campan and Truta.

Cormode et al. [7, 8] consider a framework where two sets of entities (e.g., patients and drugs) are connected by links (e.g., which patient takes which drugs), and each entity is also described by a set of attributes. The adversary relies upon knowledge of attributes rather than graph structure in devising a matching attack. To prevent matching attacks, their technique masks the mapping between vertices in the graph and real-world entities by clustering the vertices and the corresponding entities into groups. Zheleva and Getoor [33] consider the case where there are multiple types of edges, one of which is sensitive and should be protected. It is assumed that the network is published without the sensitive edges and the adversary predicts sensitive edges based on the observed non-sensitive edges.

In the second category of methods, Liu and Terzi [19] consider the case that a vertex can be identified by its degree. Their algorithms use edge additions and deletions in order to make the graph *k-degree anonymous*, meaning that for every vertex there are at least $k-1$ other vertices with the same degree.

Zhou and Pei [34] consider the case that a vertex can be identified by its radius-one induced subgraph. Adversarial knowledge stronger than the degree is also considered by Thompson and Yao [30], who assume that the adversary knows the degrees of the neighbors, the degrees of the neighbors of the neighbors, and so forth. Zou et al. [35] and Wu et al. [31] assume that the adversary knows the complete graph, and the location of the vertex in the graph; hence, the adversary can always identify a vertex in any copy of the graph, unless the graph has other vertices that are automorphically-equivalent.

In the last category of methods, Hay et al. [12] study the effectiveness of random perturbations for identity obfuscation. They concentrate on degree-based re-identification of vertices. Given a vertex v in the real network, they quantify the level of anonymity that is provided for v by the perturbed graph as $(\max_u\{\Pr(v \mid u)\})^{-1}$, where the maximum is taken over all vertices u in the released graph and $\Pr(v \mid u)$ stands for the belief probability that u is the image of the target vertex v. By performing experimentation on the Enron dataset, using various values for the number h of added and removed edges, they conclude that in order to achieve a meaningful level of anonymity for the vertices in the graph, h has to be tuned so high that the resulting features of the perturbed graph no longer reflect those of the original graph.

Ying et al. [32] compare random-perturbation methods to the method of k-degree anonymity [19]. They too use the a-posteriori belief probabilities to quantify the level of anonymity. Based on experimentation on two modestly-sized datasets (Enron and Polblogs) they conclude that the deterministic approach for k-degree anonymity preserves the graph structure better than random-perturbation methods.

In a more recent study, Bonchi et al. [4] take a different approach, by considering the entropy of the a-posteriori belief probability distributions as a measure of identity obfuscation. The rationale is that while using the a-posteriori belief probabilities is a local measure, the entropy is a global measure that examines the entire distribution of these belief probabilities. Bonchi et al. show that the entropy measure is more accurate than the a-posteriori belief probability, in the sense that the former distinguishes between situations that the latter perceives as equivalent. Moreover, the obfuscation level quantified by means of the entropy is always greater than the one based on a-posteriori belief probabilities. Finally, by means of a thorough experimentation on three large datasets, using several graph statistics and comparing also to Liu and Terzi [19], they demonstrate that random perturbation could be used to achieve meaningful levels of obfuscation while preserving most of the features of the original graph.

3. OBFUSCATION BY UNCERTAINTY

Let $G = (V, E)$ be an undirected graph, where V is the set of vertices and E is the set of edges. We write V_2 to denote the set of all $\binom{n}{2}$ unordered pairs of vertices from V, that is, $V_2 = \{(v_i, v_j) \mid 1 \leq i < j \leq n\}$. The goal is to anonymize

the graph G so that the identity of its vertices is obfuscated. We propose to publish G as an uncertain graph $\tilde{G} = (V, \mathbf{p})$, formally defined as follows.

DEFINITION 1. *Given a graph $G = (V, E)$, an uncertain graph on the vertices of G is a pair $\tilde{G} = (V, \mathbf{p})$, where $\mathbf{p} : V_2 \to [0, 1]$ is a function that assigns probabilities to unordered pairs of vertices.*

The original graph G and the uncertain graph $\tilde{G}$ have the same set of vertices V. For the sake of clarity, we write $v \in G$ when we speak about a vertex in G, and $v \in \tilde{G}$ when we speak about a vertex in $\tilde{G}$.

Since the mere description of an uncertain graph consists of $|V_2| = n(n-1)/2$ probability values, we propose to inject uncertainty only to a small subset of pairs of vertices. Namely, given a graph G, we create a subset $E_C \subseteq V_2$ of candidate edges, and then we inject uncertainty only to the pairs of vertices in E_C, while we implicitly assume that $\mathbf{p}(u, v) = 0$ for all $(u, v) \notin E_C$. The size of E_C will be set so that $|E_C| = c|E|$, for a small constant $c > 1$. In Section 5 we describe a strategy for selecting E_C, given G and a user-defined parameter c.

The uncertain graph $\tilde{G}$ induces a collection of *possible worlds* $\mathcal{W}(\tilde{G})$. A possible world $W \in \mathcal{W}(\tilde{G})$ is a graph $W = (V, E_W)$, where $E_W \subseteq E_C$. The edge probabilities in the uncertain graph $\tilde{G}$ imply that the probability of W is

$$\Pr(W) = \prod_{e \in E_W} \mathbf{p}(e) \cdot \prod_{e \in E_C \setminus E_W} (1 - \mathbf{p}(e)). \qquad (1)$$

Let us consider the knowledge that an adversary may extract from such an uncertain graph about a given target vertex in G. Following the literature, we assume that the adversary knows some vertex property P of his target vertex [4, 12, 19, 30, 31, 32, 34, 35]. Examples of such properties, as discussed in Section 2, are the degree, the degrees of the vertex and its neighbors, and the neighborhood subgraph induced by the target vertex and its neighbors.

Let Ω_P be the domain in which P takes values, e.g., if P is the degree property then $\Omega_P = \{0, \ldots, n-1\}$. Given an uncertain graph $\tilde{G}$ and a property P, for each $v \in \tilde{G}$ and $\omega \in \Omega_P$ we define the probability $X_v(\omega)$ that v originated from a vertex in G with property value ω. Specifically,

$$X_v(\omega) = \sum_{W \in \mathcal{W}(\tilde{G})} \Pr(W) \cdot \chi_{v,\omega}(W), \qquad (2)$$

where $\Pr(W)$ is given in Equation (1), and $\chi_{v,\omega}(W)$ is a 0–1 variable that indicates if the vertex v has the property value ω in the possible world W. In other words, $X_v(\omega)$ is the sum of probabilities of all possible worlds in which the vertex v has the given property value ω.

The probabilities $X_v(\omega)$ may be arranged in a $n \times |\Omega_P|$ matrix, where each row corresponds to one vertex $v \in \tilde{G}$ and it gives the corresponding probability distribution $X_v(\omega)$ over all possible values $\omega \in \Omega_P$. The columns of that matrix are proportional to the probability distributions that correspond to property values. More precisely, the normalized column corresponding to property $\omega \in \Omega_P$, i.e.,

$$Y_\omega(v) := \frac{X_v(\omega)}{\sum_{u \in \tilde{G}} X_u(\omega)} \qquad (3)$$

is the probability that v is the image in $\tilde{G}$ of a vertex that had the property ω in G.

$X_v(\omega)$	deg=0	deg=1	deg=2	deg=3
v_1:	0.006	0.092	0.398	0.504
v_2:	0.054	0.348	0.542	0.056
v_3:	0.020	0.260	0.720	0.000
v_4:	0.180	0.740	0.080	0.000

$Y_\omega(v)$	deg=0	deg=1	deg=2	deg=3
v_1:	0.023	0.064	0.229	0.900
v_2:	0.208	0.242	0.311	0.100
v_3:	0.077	0.180	0.414	0.000
v_4:	0.692	0.514	0.046	0.000

Table 1: The matrices $X_v(\omega)$ and $Y_\omega(v)$ for the uncertain graph in Figure 1(b) and the degree property.

EXAMPLE 1. *Consider the uncertain graph in Figure 1(b) and assume property P_1. Table 1 gives the corresponding matrix $X_v(\omega)$, in which each row gives the probability distribution regarding the degree of the corresponding vertex in G. For instance, the probability that v_1 has degree 2 is* $0.7 \cdot 0.9 \cdot (1-0.8) + 0.7 \cdot (1-0.9) \cdot 0.8 + (1-0.7) \cdot 0.8 \cdot 0.7 = 0.398$.

The columns of $X_v(\omega)$, after normalizing them, give the corresponding $Y_\omega(v)$ distributions for each value of the degree (shown also in Table 1). For instance, if we look for a vertex that has degree 3 in G, it is either v_1, with probability 0.9, or v_2, with probability 0.1.

To further stress the difference between the two probability distributions, $X_v(\omega)$ and $Y_\omega(v)$, let us consider an uncertain graph $\tilde{G}$ in which all edge probabilities are either 0 or 1 (i.e., a certain graph). Let ω be some property value in Ω_P and assume that $P^{-1}(\omega) = \{v_{i_1}, \ldots, v_{i_k}\}$ (namely, there are exactly k vertices with the property ω in the graph). Then, for all $v \in P^{-1}(\omega)$, $X_v(\omega) = 1$ (since each of them has the property ω with certainty) and $X_v(\omega') = 0$ for any other property $\omega' \neq \omega$ (since any vertex can have in any certain graph just one property). Furthermore, $X_v(\omega) = 0$ for all $v \notin P^{-1}(\omega)$. Let us now turn to consider the column in the matrix that corresponds to ω. Then $Y_\omega(v) = 1/k$ for each of the k vertices in $P^{-1}(\omega)$ and $Y_\omega(v) = 0$ for all other vertices since if we look for a specific vertex in the graph with property ω and that is the only information that we know about that sought-after vertex, then it can be any one of the vertices in $P^{-1}(\omega)$ with probability $1/k$.

We are ready to define our notion of privacy.

DEFINITION 2 ((k, ε)-OBFUSCATION). *Let P be a vertex property, $k \geq 1$ be a desired level of obfuscation, and $\varepsilon \geq 0$ be a tolerance parameter. The uncertain graph $\tilde{G}$ is said to k-obfuscate a given vertex $v \in G$ with respect to P if the entropy of the distribution $Y_{P(v)}$ over the vertices of $\tilde{G}$ is greater than or equal to $\log_2 k$:*

$$H(Y_{P(v)}) \geq \log_2 k.$$

The uncertain graph $\tilde{G}$ is a (k, ε)-obfuscation with respect to property P if it k-obfuscates at least $(1 - \varepsilon)n$ vertices in G with respect to P.

Namely, given the considered attack scenario, in which the adversary uses a background knowledge of property P of his target vertex, we wish to lower bound the entropy of the distribution it induces over the obfuscated graph vertices by $\log_2 k$ (in similarity to the privacy goal in k-anonymity). As

for the tolerance parameter ε, it serves the following purpose. Considering the fact that degree sequences in typical social networks have very skewed distribution, trying to obfuscate some very unique vertices (such as Barack Obama or CNN in `twitter` or `Facebook`) is on the one hand hopeless, and on the other hand not necessarily needed: these vertices do not represent "normal" users, and identifying them does not disclose anyone's personal information. In fact, as we will see later, our obfuscation algorithm guarantees that the ε-fraction of vertices for which the privacy requirement is not satisfied can be forced to be taken from some specific sub-population; for example, in the case of degree obfuscation they are vertices with high degree.

EXAMPLE 2. *Consider again the graph in Figure 1. Vertex v_1 has degree 3 in the original graph. Thus, in order to check the level of obfuscation of this vertex in the obfuscated graph we have to measure the entropy of the column* $\deg = 3$ *of Table $Y_\omega(v)$. That entropy is approximately* 0.469*, which is rather low, meaning that the identity of v_1 is not obfuscated enough in the uncertain graph in Figure 1(b). Vertex v_2 has degree 1 in the original graph. The entropy of the column* $\deg = 1$ *is* $\approx 1.688 > \log_2 3$*. Vertices v_3 and v_4 have degree 2, and the entropy of the corresponding column is* $\approx 1.742 \geq \log_2 3$*. Therefore, as three out of four vertices are 3-obfuscated, the graph in Figure 1(b) provides a (3,0.25)-obfuscation for the graph in Figure 1(a).*

4. QUANTIFYING THE OBFUSCATION

In this section we describe how to compute the level of obfuscation with regard to the degree property. When P is the degree, $\Omega_P = \{0, \ldots, n-1\}$, and, consequently, the matrix has n rows and n columns. We need to describe how to compute $X_v(\omega)$ for all $v \in \tilde{G}$ and $\omega \in \{0, \ldots, n-1\}$. Once the full matrix X_v is given, it is possible to derive the distributions Y_ω over the vertices of $\tilde{G}$ for all $\omega \in P(G)$ and then verify the k-obfuscation property.

Fix $v \in \tilde{G}$ and let $e_1, \ldots, e_{n-1}$ be the $n-1$ pairs of vertices that include v. For each $1 \leq i \leq n-1$, e_i is a Bernoulli random variable that equals 1 with some probability p_i. Letting d_v be the random variable corresponding to the degree of v, we have

$$d_v = \sum_{i=1}^{n-1} e_i. \tag{4}$$

Then for each possible degree $\omega \in \Omega_P$ of v, we have $X_v(\omega) = \Pr(d_v = \omega)$.

LEMMA 1. *The probability distribution of d_v may be computed exactly in time $O(n^2)$.*

PROOF. Let $d_v^\ell := \sum_{i=1}^{\ell} e_i$ denote the partial sum of the first ℓ Bernoulli random variables. We will show that once we have the distribution of $d_v^{\ell-1}$, we can compute that of d_v^ℓ in time $O(\ell)$. Hence, the distribution of $d_v = d_v^{n-1}$ can be computed in time $\sum_{\ell=1}^{n-1} O(\ell) = O(n^2)$. Indeed,

$$\Pr(d_v^\ell = j) = \Pr(d_v^{\ell-1} = j-1) \cdot p_\ell + \Pr(d_v^{\ell-1} = j) \cdot (1 - p_\ell).$$

Therefore, computing a single probability in the distribution of d_v^ℓ takes constant time (given the full distribution of $d_v^{\ell-1}$), and, consequently, computing the entire distribution of d_v^ℓ over all $0 \leq j \leq \ell$ takes time $O(\ell)$. □

It should be noted that since we choose to inject uncertainty only to a subset E_C of pairs of vertices, the sum in Equation (4) is taken only over the pairs of vertices in E_C that include the vertex v. Hence, if d is the average degree in G, the average number of addends in d_v is dc, where $c = |E_C|/|E|$.

In cases where the sum in Equation (4) has a large number of addends, we may adopt an alternative approach. Since d_v is the sum of independent random variables, it may be approximated by the normal distribution $N(\mu, \sigma^2)$, where $\mu = \sum_{i=1}^{n-1} E(e_i) = \sum_{i=1}^{n-1} p_i$ and $\sigma^2 = \sum_{i=1}^{n-1} Var(e_i) = \sum_{i=1}^{n-1} p_i(1-p_i)$ as implied by the Central Limit Theorem [16]. (The Central Limit Theorem becomes effective already for $n \approx 30$; for typical sizes of n in social networks, the normal approximation becomes very accurate.) Specifically, $\Pr(d_v = \omega) \approx \int_{\omega-1/2}^{\omega+1/2} \Phi_{\mu,\sigma}(x)dx$ for $\omega \in \Omega_P = \{0, \ldots, n-1\}$, where

$$\Phi_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{5}$$

5. INJECTING UNCERTAINTY

In this section we describe our algorithm, which, given a graph G, a desired level of obfuscation k, and a tolerance parameter ε, injects a minimal level of uncertainty to the graph so that it becomes (k, ε)-obfuscated with respect to a vertex property P.

5.1 Overview

As discussed in Section 3, we inject uncertainty in the graph by assigning probabilities to a subset $E_C \subseteq V_2$ of pairs of vertices, such that $|E_C| = c|E|$, for a small constant parameter c. The selection of E_C is described in a subsequent section. Once E_C is selected, only the pairs $e \in E_C$ will become uncertain edges in $\tilde{G}$. All other pairs $e \notin E_C$ will be certain non-edges, i.e., $\mathbf{p}(e) = 0$. To establish the uncertainty of each pair $e \in E_C$, we select a random perturbation $r_e \in [0, 1]$. If $e \in E$, it becomes an uncertain edge in $\tilde{G}$ with probability $\mathbf{p}(e) = 1 - r_e$; if $e \in E_C \setminus E$, it becomes an uncertain edge with probability $\mathbf{p}(e) = r_e$.

In order for the uncertain graph $\tilde{G}$ to preserve the characteristics of the original graph G, smaller values of the perturbation parameter r_e should be favored. A natural candidate for the generating distribution of r_e is the $[0, 1]$-truncated normal distribution,

$$R_\sigma(r) := \begin{cases} \frac{\Phi_{0,\sigma}(r)}{\int_0^1 \Phi_{0,\sigma}(x)dx} & 0 \leq r \leq 1 \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $\Phi_{\mu,\sigma}$ is the density function of a Gaussian distribution provided in Equation (5). As the standard deviation σ of the normal distribution decreases, a greater mass of R_σ will concentrate near $r = 0$ and then the amount of injected uncertainty will be smaller. Thus, small values of σ contribute towards better maintaining the characteristics of the original graph, but at the same time they provide lower levels of obfuscation. Larger values of σ have the opposite effect.

A key feature of our method is to select judiciously the perturbation r_e for each pair $e = (u, v) \in E_C$, depending on properties of the vertices u and v. Hence, the random variable r_e is drawn from $R_{\sigma(e)}$, where the parameter $\sigma(e)$ depends on the vertices that e connects. The perturbation will be larger for edges that connect more unique vertices,

which, consequently, require higher levels of uncertainty to "blend in the crowd," and smaller for edges that connect more "typical" vertices.

Additionally, in order to prevent identifying pairs $e \in E_C$ that are true edges in G (by turning every pair $e \in E_C$ to an edge if $\mathbf{p}(e) \geq 0.5$ and to a non-edge otherwise), the perturbation r_e is drawn from the *uniform* distribution in $[0, 1]$, rather than from the distribution R_σ, for a q-fraction of the pairs $e \in E_C$, with $0 < q \ll 1$.

5.2 Uniqueness Scores of Vertices

For certain properties of interest, such as degree, the majority of vertices in real-world graphs are already anonymous even without random perturbations. The reason is that for most values of the property P there are many vertices that have that value. Hence, we aim at controlling the amount of applied perturbation, so that larger perturbation is added at vertices that are less anonymized in the original graph. In particular, we suggest to calibrate the perturbation applied to a pair $e = (u, v) \in E_C$ according to the "uniqueness" of the two vertices u and v with respect to the property P. Namely, if both $P(u)$ and $P(v)$ are frequent values, then r_e should be very small; on the other hand, if $P(u)$ and $P(v)$ are outlier values, then r_e should be higher. We proceed to explain our method in detail.

Let $P : V \to \Omega_P$ be a property defined on the set of vertices V. Further, consider a distance function d between values in the range Ω_P of P. So, for each pair of values, $\omega, \omega' \in \Omega_P$, a distance $d(\omega, \omega') \geq 0$ is defined. For example, for the degree property P_1, the distance d is the modulus of the difference of two degrees, while for the radius-one subgraph property (P3), the distance d is the edit distance between two subgraphs.

Definition 3. *Let $P : V \to \Omega_P$ be a property on the set of vertices V of the graph G, let d be a distance function on Ω_P, and let $\theta > 0$ be a parameter. Then the θ-commonness of the property value $\omega \in \Omega_P$ is $C_\theta(\omega) := \sum_{v \in V} \Phi_{0,\theta}(d(w, P(v)))$, while the θ-uniqueness of $\omega \in \Omega_P$ is $U_\theta(\omega) := \frac{1}{C_\theta(\omega)}$.*

In the above definition the function Φ is the Gaussian distribution given by Equation (5). The commonness of the property value ω is a measure of how typical is the value ω among the vertices of the graph. It is obtained as a weighted average over all other property values ω', where the weight decays exponentially as a function of the distance between ω and ω'. The uniqueness is the inverse of the commonness. It should be noted that the commonness and uniqueness are meaningful only as relative measures, as they allow to assess how one property value is more common, or more unique, in G than another property value.

Commonness and uniqueness scores depend on the parameter θ, which determines the decay rate of the average weights as a function of the distance. We set $\theta = \sigma$ as larger amounts of uncertainty imply that property values may be spread on larger domains of Ω_P due to injecting uncertainty.

5.3 The Obfuscation Algorithm

Our algorithm for computing a (k, ε)-obfuscation of a graph with respect to a vertex property P is outlined as Algorithm 1. Targeting for high utility, the algorithm aims at injecting the minimal amount of uncertainty needed to achieve the required obfuscation. Computing the minimal amount of uncertainty is achieved via a binary search on the value of the uncertainty parameter σ.

Algorithm 1 (k, ε)-obfuscation

Input: $G = (V, E)$, vertex property P, obfuscation level k, tolerance ε, size multiplier c, and white noise level q.
Output: A (k, ε)-obfuscation $\tilde{G}$ of G with respect to P.

1: $\sigma_\ell \leftarrow 0$
2: $\sigma_u \leftarrow 1$
3: **repeat**
4: $\quad \langle \tilde{\varepsilon}, \tilde{G} \rangle \leftarrow$ GenerateObfuscation$(G, \sigma_u, P, k, \varepsilon, c, q)$
5: $\quad$ **if** $\tilde{\varepsilon} = \infty$ **then** $\sigma_u \leftarrow 2\sigma_u$
6: **until** $\tilde{\varepsilon} \neq \infty$
7: $\tilde{G}_{found} \leftarrow \tilde{G}$
8: **while** $\sigma_\ell + \delta < \sigma_u$ **do**
9: $\quad \sigma \leftarrow (\sigma_\ell + \sigma_u)/2$
10: $\quad \langle \tilde{\varepsilon}, \tilde{G} \rangle \leftarrow$ GenerateObfuscation$(G, \sigma_u, P, k, \varepsilon, c, q)$
11: $\quad$ **if** $\tilde{\varepsilon} = \infty$ **then** $\sigma_\ell \leftarrow \sigma$
12: $\quad$ **else** $\tilde{G}_{found} \leftarrow \tilde{G}$; $\sigma_u \leftarrow \sigma$
13: **return** $\tilde{G}_{found}$

The binary-search flow of Algorithm 1 is determined by the function GenerateObfuscation, which is shown as Algorithm 2. The function GenerateObfuscation returns a pair $\langle \tilde{\varepsilon}, \tilde{G} \rangle$ where $\tilde{\varepsilon} = \infty$ or $0 \leq \tilde{\varepsilon} \leq \varepsilon$. In the first case, the function could not find a (k, ε)-obfuscation with the given uncertainty parameter. In the latter case, $\tilde{G}$ is a $(k, \tilde{\varepsilon})$-obfuscation of G with respect to P, and thus, also a (k, ε)-obfuscation.

The obfuscation algorithm starts with an initial guess of an upper bound σ_u, which is iteratively doubled until a (k, ε)-obfuscated graph is found. Then, the binary-search process is performed using $\sigma_\ell = 0$ as the lower bound, and the upper bound σ_u that was found. The binary search terminates when the search interval is sufficiently short, and the algorithm outputs the best (k, ε)-obfuscation found (i.e., the last one that was successfully generated, because it will be the one obtained with the smallest σ).

The function GenerateObfuscation (Algorithm 2) aims at finding a (k, ε)-obfuscation of G using a given standard deviation parameter σ. First, it computes the σ-uniqueness level $U_\sigma(P(v))$ for each vertex $v \in G$. The more unique a vertex is, the harder it is to obfuscate it. Hence, in order to use the "uncertainty budget" σ in the most efficient way, the algorithm performs the following two pre-processing steps.

(Line 2): Since it is allowed not to obfuscate $\varepsilon|V|$ of the vertices, the algorithm selects the set H of $\lceil \frac{\varepsilon}{2}|V| \rceil$ vertices with largest uniqueness scores, which are the vertices that would require the largest amount of uncertainty, and excludes them from the subsequent obfuscation efforts. In later steps, the algorithm will inject uncertainty only to edges that are not adjacent to any of the vertices in H. (The algorithm could also receive H, or part of H, as an input, instead of fully selecting it on its own.)

(Line 3): The set of vertices not in H will need to be obfuscated. To obfuscate more unique vertices, higher uncertainty is necessary. Thus, edges need to be sampled with higher probability if they are adjacent to unique vertices. In order to handle this sampling process, our algorithm assigns a probability $Q(v)$ to every $v \in V$, which is proportional to the uniqueness level $U_\sigma(P(v))$ of v.

After that, the search for a (k, ε)-obfuscation starts: since the algorithm is randomized and there is a non-zero prob-

Algorithm 2 GenerateObfuscation

Input: $G = (V, E), P, k, \varepsilon, c, q$, and standard deviation σ.
Output: A pair $\langle \tilde{\varepsilon}, \tilde{G} \rangle$, where $\tilde{G}$ is a $(k, \tilde{\varepsilon})$-obfuscation (with $\tilde{\varepsilon} < \varepsilon$), or $\tilde{\varepsilon} = \infty$ if a (k, ε)-obfuscation was not found.

1: **for all** $v \in V$ **compute** the σ-uniqueness $U_\sigma(P(v))$
2: $H \leftarrow$ the set of $\lceil \frac{\varepsilon}{2}|V| \rceil$ vertices with largest $U_\sigma(P(v))$
3: **for all** $v \in V$ **do** $Q(v) \leftarrow U_\sigma(P(v)) / \sum_{u \in V} U_\sigma(P(u))$
4: $\tilde{\varepsilon} \leftarrow \infty$
5: **for** t times **do**
6: $E_C \leftarrow E$
7: **repeat**
8: randomly pick a vertex $u \in V \setminus H$ according to Q
9: randomly pick a vertex $v \in V \setminus H$ according to Q
10: **if** $(u, v) \in E$ **then** $E_C \leftarrow E_C \setminus \{(u, v)\}$
11: **else** $E_C \leftarrow E_C \cup \{(u, v)\}$
12: **until** $|E_C| = c|E|$
13: **for all** $e \in E_C$ **do**
14: **compute** $\sigma(e)$
15: draw w uniformly at random from $[0, 1]$
16: **if** $w < q$
17: **then** draw r_e uniformly at random from $[0, 1]$
18: **else** draw r_e from the random distribution $R_{\sigma(e)}$
19: **if** $e \in E$ **then** $\mathbf{p}(e) \leftarrow 1 - r_e$ **else** $\mathbf{p}(e) \leftarrow r_e$
20: $\varepsilon' \leftarrow |\{v \in V : \text{not } k\text{-obfuscated by } G' = (V, \mathbf{p})\}|/|V|$
21: **if** $\varepsilon' \leq \varepsilon$ **and** $\varepsilon' < \tilde{\varepsilon}$ **then** $\tilde{\varepsilon} \leftarrow \varepsilon'$; $\tilde{G} \leftarrow G'$
22: **return** $\langle \tilde{\varepsilon}, \tilde{G} \rangle$

ability of failure, t attempts to find a (k, ε)-obfuscation are performed (Lines 5-22; in our experiments we used $t = 5$).

Each attempt begins by randomly selecting a subset $E_C \subseteq V_2$, which will be subjected to uncertainty injection (Lines 6-12). The set E_C, whose target size is $|E_C| = c|E|$, is initialized to be E (Line 6). Then, the algorithm randomly selects two distinct vertices u and v, according to the probability distribution Q, such that none of them is in H (Lines 8-9). The pair of vertices (u, v) is removed from E_C if it is an edge, or added to E_C otherwise (Lines 10-11). The process is repeated until E_C reaches the required size $c|E|$. Since in typical graphs, the number of non-edges is significantly larger than the number of edges, i.e., $|E| \ll |V_2|/2$, the loop in Lines 7-12 ends very quickly, for small values of c, and the resulting set E_C includes most of the edges in E.

Next, in Line 14, we redistribute the uncertainty levels among all pairs $e \in E_C$ in proportion to their uniqueness levels. Specifically, we define for each $e = (u, v) \in E_C$ its σ-uniqueness level,

$$U_\sigma(e) := \frac{U_\sigma(P(u)) + U_\sigma(P(v))}{2},$$

and then set

$$\sigma(e) = \sigma|E_C| \cdot \frac{U_\sigma(e)}{\sum_{e' \in E_C} U_\sigma(e')}, \tag{7}$$

so that the average of $\sigma(e)$ over all $e \in E_C$ equals σ.

Given the edge uncertainty levels, $\sigma(e)$, we select for each pair of vertices $e \in E_C$ a random perturbation r_e. For the majority of the pairs (an $(1 - q)$-fraction, where the input parameter q is small) we select r_e from the random distribution $R_{\sigma(e)}$ (see Equation (6)). For the remaining q-fraction of pairs we select r_e from the uniform distribution on $[0, 1]$. If e is an actual edge ($e \in E$), it turns into an uncertain edge in $\tilde{G}$ with associated probability of $\mathbf{p}(e) = 1 - r_e$. If e is a non-edge in G ($e \in E_C \setminus E$), it turns into an uncertain edge in $\tilde{G}$ with probability $\mathbf{p}(e) = r_e$ (Line 19).

If the algorithm finds a (k, ε)-obfuscated graph in one of its t trials, it returns the obfuscated graph with minimal ε. If, on the other hand, all t attempts fail, the algorithm indicates the failure by returning $\tilde{\varepsilon} = \infty$.

6. UTILITY OF THE UNCERTAIN GRAPH

In order to prove the practical relevance of our proposal, we need to show that: (1) the uncertain graph maintains high utility, i.e., it is highly similar to the original graph in terms of characteristic properties; and (2) the computation of these properties can be carried out in reasonable time.

In the rest of this section, we discuss several graph statistics and show how to compute them in uncertain graphs. In our experimental assessment, we use those statistics to evaluate the utility of the proposed graph obfuscation.

Further evidence to the usefulness of publishing an uncertain graph is provided by the many recent papers on mining and querying uncertain graphs [14, 15, 24, 36, 37, 38].

6.1 Sampling

Given a standard (certain) graph G, let $S[G]$ be the value of a statistical measure S for G. Examples of such a statistical measure S are the average degree, the diameter, the clustering coefficient of G, and so on. In order to define the value of S in an uncertain graph $\tilde{G} = (V, \mathbf{p})$, the most natural choice is to consider the *expected value* of $S[\tilde{G}]$, namely,

$$E(S[\tilde{G}]) = \sum_{W \in \mathcal{W}(\tilde{G})} \Pr(W) \cdot S(W), \tag{8}$$

where $\Pr(W)$ is given in Equation (1). While for some statistics it is possible to compute the expected value in Equation (8) without explicitly performing a summation over the exponential number of possible worlds (as we will see in Section 6.2), for other statistics such a computation remains infeasible. Hence, we have to resort to approximation by sampling. Namely, we sample a subset of possible worlds $\mathcal{W}' \subseteq \mathcal{W}(\tilde{G})$ according to the distribution induced by the probabilities $\Pr(W)$, and then take the average $\overline{S}$ of the statistic S in the sampled worlds as an approximation of $E(S[\tilde{G}])$:

$$\overline{S} := \frac{1}{|\mathcal{W}'|} \sum_{W \in \mathcal{W}'} S(W). \tag{9}$$

Sampling a possible world according to the distribution $\Pr(W)$ is carried out by sampling independently each edge e with probability $\mathbf{p}(e)$.

The following lemma provides a probabilistic error bound for approximating the expected value by an average over a number of sampled worlds.

LEMMA 2. *Let $\tilde{G} = (V, \mathbf{p})$ be an uncertain graph and assume that S is a graph statistic that satisfies $a \leq S \leq b$. Let $r = |\mathcal{W}'|$ denote the number of sampled worlds and $\overline{S}$ be the average of the statistic S over those worlds, Equation (9). Then for every $\varepsilon > 0$,*

$$\Pr(|E(S[\tilde{G}]) - \overline{S}| \geq \varepsilon) \leq 2 \exp\left(-\frac{2\varepsilon^2 r}{(b - a)^2}\right). \tag{10}$$

PROOF. Let $\mathcal{W}' = \{W_i\}_{1 \le i \le r}$ be the set of r graphs that were sampled from $\tilde{G} = (V, \mathbf{p})$. Then $S_i = S[W_i]$, $1 \le i \le r$, are independent and identically distributed random variables. Since $E(S_i) = E(S[\tilde{G}])$ for all $1 \le i \le r$, it follows that also $E(\overline{S}) = E(S[\tilde{G}])$. Hence, inequality (10) follows directly from Hoeffding's inequality [13]. □

COROLLARY 1. *For given error bound ε and probability of failure δ, we have* $\Pr(|E(S[\tilde{G}]) - \overline{S}| \ge \varepsilon) \le \delta$, *provided that* $r \ge \frac{1}{2}\left(\frac{b-a}{\varepsilon}\right)^2 \ln\left(\frac{2}{\delta}\right)$.

In the next section, we define a number of scalar and vector statistics of interest; when possible, we also provide an explicit computation of $E(S[\tilde{G}])$.

6.2 Statistics Based on Degree

Let $d_1, \ldots, d_n$ denote the degree sequence in a graph G. The statistic S is called a degree-based statistic if $S = F(d_1, \ldots, d_n)$ for some function F. Examples of such statistics are:

- *Number of edges*: $S_{\text{NE}} = \frac{1}{2}\sum_{v \in V} d_v$.
- *Average degree*: $S_{\text{AD}} = \frac{1}{n}\sum_{v \in V} d_v$.
- *Maximal degree*: $S_{\text{MD}} = \max_{v \in V} d_v$.
- *Degree variance*:[1] $S_{\text{DV}} = \frac{1}{n}\sum_{v \in V} (d_v - S_{\text{AD}})^2$.

When $\tilde{G}$ is an uncertain graph, $d_1, \ldots, d_n$ are random variables. If F is a linear function, then we have

$$E(S[\tilde{G}]) = E(F(d_1, \ldots, d_n)) = F(E(d_1), \ldots, E(d_n)). \quad (11)$$

Hence, since the expected degree of a vertex $v \in V$ is equal to the sum of probabilities of its adjacent edges, the computation of the expected statistic is easy, in the case of a linear function. As the first two examples above, S_{NE} and S_{Ad}, correspond to a linear function F, we have:

$$E(S_{\text{NE}}[\tilde{G}]) = E\left(\frac{1}{2}\sum_{v \in V} d_v\right) = \frac{1}{2}\sum_{v \in V}\sum_{u \in V \setminus v} \mathbf{p}(u,v) = \sum_{e \in V_2} \mathbf{p}(e),$$

and

$$E(S_{\text{AD}}[\tilde{G}]) = E\left(\frac{1}{n}\sum_{v \in V} d_v\right) = \frac{1}{n}\sum_{v \in V}\sum_{u \in V \setminus v} \mathbf{p}(u,v) = \frac{2}{n}\sum_{e \in V_2} \mathbf{p}(e).$$

Things are less simple when F is non-linear, since then Equation (11) does not hold. This is the case with the latter two examples — the maximal degree ($F = \max$) and the degree variance (F is quadratic). For these statistics we adopt the sampling approach described in the previous section. Since the maximal degree is at most $n-1$, the statistic S_{MD} satisfies Corollary 1 with $a = 0$ and $b = n - 1$. Similarly, the statistic S_{DV} satisfies Corollary 1 with $a = 0$ and $b = (n-1)^2$. It should also be noted that we can compute $E(S_{\text{DV}}[\tilde{G}])$ precisely. However, the cost of evaluating the corresponding formulas, which we omit herein, is quadratic in the number of vertices.

We proceed to describe two additional statistics that are based on the degree distribution. In the following we use $\Delta(d)$, with $0 \le d \le n-1$, to denote the fraction of vertices in the graph G that have degree d.

[1]This is one of the measures of graph heterogeneity [27].

The first statistic, denoted by S_{PL}, is the power-law exponent of the degree distribution. For this statistic, we assume that the degree distribution follows a power law, $\Delta(d) \sim d^{-\gamma}$, and S_{PL} is an estimate of $-\gamma$. In our experiments, we focused on higher degrees where the power law fits better, and we fitted the exponent ignoring smaller degrees.

The second statistic is the degree distribution itself, $S_{\text{DD}} := (\Delta(0), \Delta(1), \ldots, \Delta(n-1))$. As opposed to all previous statistics, which were scalar, this one is a vector. In fact, each of the previous statistics may be derived from the degree distribution. To approximate $S_{\text{DD}}[G]$ we adopt once more the sampling approach: for every degree d, we approximate $\Delta(d)$ by the average $\overline{\Delta}(d)$ obtained over the sampled possible worlds.

6.3 Statistics Based on Shortest-path Distance

Other interesting measures characterizing a graph are those based on the shortest-path distance between pairs of vertices. Computing distance distributions on large graphs is far from trivial, as explained in the survey of Kang et al. [17]. While exact solutions using breadth-first search or Floyd's algorithm are out of question, there is still no consensus in the research community on which approximate technique is best [9]. Some methods are based on sampling, for example, performing a breadth-first search from a selected set of vertices [6, 18], and other are based on information diffusion [3, 17, 23]. While the former are simpler to implement, diffusion-based techniques have the advantage of being more general (they are natively designed for directed graphs, while most sampling methods only work for undirected ones) and scale more gracefully.

Defining the distance between pairs of vertices in uncertain graphs is not an easy task since, typically, the corresponding ensemble of possible worlds will include disconnected instances; in such disconnected possible worlds, some of the pairwise distances are infinite [24]. We directly avoid this problem by defining the distance-based measures S only on pairs of vertices that are path-connected.

We consider five measures:

- *Average distance*: S_{APD} is the average distance among all pairs of vertices that are path-connected.
- *Effective diameter*: S_{EDiam} is the 90-*th* percentile distance among all path-connected pairs of vertices, i.e., the minimal value for which 90% of the finite pairwise distances in the graph are no larger than. In our experiments, we used the variant that linearly interpolates between the 90-*th* percentile and the successive integer.
- *Connectivity length*: The statistic S_{CL} is defined as the harmonic mean of all pairwise distances in the graph, $S_{\text{CL}} = \frac{n(n-1)}{2}\left(\sum_{(u,v) \in V_2} \frac{1}{\text{dist}(u,v)}\right)^{-1}$ [20]. Note that by taking $\frac{1}{\text{dist}(u,v)} = 0$ for non path-connected pairs (u,v), the connectivity length can be defined as the average over all vertex pairs, independently on whether they lie in the same connected component.
- *Distribution of pairwise distances*: S_{PDD} is the distribution of pairwise distances in the graph, where $S_{\text{PDD}}[k]$ is the number of pairs of vertices whose distance equals k, for $1 \le k \le n-1$, and $S_{\text{PDD}}[\infty]$ is the number of pairs of vertices that are not path-connected.
- *Diameter*: S_{Diam} is the maximum distance among all path-connected pairs of vertices.

For computing the above measures we rely on sampling. It is easy to see that Lemma 2 and Corollary 1 hold for each of those statistics with $a = 1$ and $b = n - 1$.

To estimate the distance distribution in a given (certain) graph, we use HyperANF [3], a diffusion-based algorithm that provides a good tradeoff between accuracy guarantees and execution time. As the algorithm is probabilistic, the results that it gives may drift from the real ones, depending on the number of registers used for the evaluation. Such drifts affect the variance over the value obtained for each point of the distance distribution. To limit the effect of such probabilistic drifts, we repeat the execution of HyperANF and used jackknifing [26] to infer the standard error of the statistics that we compute; in our experiments this error ranges between 0.2% and 2%.

While the HyperANF approach is viable for the first four statistics described above, it falls short in estimating the diameter. Exact diameter estimation is difficult and even heuristic methods such as [9] would be too inefficient to be executed on many sampled worlds. As a result, we focus on estimating a lower bound S_{DiamLB} for S_{Diam}: such a lower bound is computed as the largest distance t for which the approximate distance distribution computed by HyperANF is nonzero; i.e., it is the largest distance t for which we estimate that there is at least one pair of vertices of distance t from each other.

6.4 Clustering Coefficient

The clustering coefficient S_{CC} measures the extent to which the edges of the graph "close triangles." More formally, given a graph G, let $T_3[G]$ be the number of cliques of size 3 in the graph G, and $T_2[G]$ be the number of connected triplets. The clustering coefficient $S_{\text{CC}}[G]$ of a graph G is then defined as $S_{\text{CC}}[G] = T_3[G]/T_2[G]$. Since $T_3[G] \leq T_2[G]$, the clustering coefficient is a number between 0 and 1.

EXAMPLE 3. *Let K_3 be the complete graph on three vertices. Then $T_3[K_3] = 1$ and $T_2[K_3] = 1$. Hence, $S_{\text{CC}}[K_3] = 1$. Consider next the graph G on three vertices u, v, w with two edges only — (u, v) and (u, w). Then $T_3[G] = 0$ and $T_2[G] = 1$, whence $S_{\text{CC}}[G] = 0$.*

Given an uncertain graph $\tilde{G}$, we can estimate the expected clustering coefficient $E(S_{\text{CC}}[\tilde{G}])$ by sampling (see Section 6.1). Since the clustering coefficient takes values between 0 and 1, we can apply Lemma 2 with $a = 0$ and $b = 1$. Thus, we can estimate $E(S_{\text{CC}}[\tilde{G}])$ within an error of at most ε and probability of success at least $1 - \delta$ by sampling at least $r = \frac{1}{2\varepsilon^2} \ln(\frac{2}{\delta})$ possible worlds.

7. EXPERIMENTAL ASSESSMENT

The objective of our experimental assessment is to show that the proposed technique is able to provide the required obfuscation levels while maintaining high data utility. In particular, we set the following concrete subgoals. For given values of k and ε, we want to assess:

1. the level of noise (specified by the value of σ) needed to achieve (k, ε)-obfuscation;
2. the running time of the obfuscation algorithm;
3. the error in the statistics of the obfuscated graph with respect to the original graph;
4. how the proposed method compares with random-perturbation methods for the same levels of obfuscation.

Table 2: Values of σ that yielded a (k, ε)-obfuscation obtained by Alg. 1. In all cases $q = 0.01$ and $c = 2$, except for the two cases marked (*) where $c = 3$.

Dataset	k	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-4}$
dblp	20	$5.9605 \cdot 10^{-8}$	$1.6153 \cdot 10^{-5}$
	60	$2.9802 \cdot 10^{-7}$	$3.2206 \cdot 10^{-3}$
	100	$1.8775 \cdot 10^{-5}$	$1.0711 \cdot 10^{-2}$
flickr	20	$2.2948 \cdot 10^{-5}$	$2.6343 \cdot 10^{-2}$
	60	$1.0397 \cdot 10^{-3}$	$7.3275 \cdot 10^{-2}$ (*)
	100	$5.8624 \cdot 10^{-3}$	$2.9273 \cdot 10^{-1}$ (*)
Y360	20	$5.9605 \cdot 10^{-8}$	$5.9605 \cdot 10^{-8}$
	60	$5.9605 \cdot 10^{-8}$	$1.0133 \cdot 10^{-6}$
	100	$5.9605 \cdot 10^{-8}$	$1.1146 \cdot 10^{-5}$

Table 3: Computation (real) time in edges/sec.

Dataset	k	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-4}$
dblp	20	1069.34	1550.78
	60	1000.64	1279.39
	100	888.908	1166.87
flickr	20	1004.93	926.45
	60	1019.05	300.39 (*)
	100	862.155	271.84 (*)
Y360	20	2113.51	1900.32
	60	1762.21	1665.80
	100	1643.84	1664.75

For our experiments, we use three large real-world datasets. dblp is a co-authorship graph extracted from a recent snapshot of the DBLP database considering only journal publications.[2] Vertices represent authors, and there is an undirected edge between two authors if they have authored a journal paper together.

flickr is a popular online community for sharing photos, with millions of users.[3] In addition to many photo-sharing facilities, users are creating a social network by explicitly marking other users as their *contacts*.

Y360: Yahoo! 360 was a social-networking and personal-communication portal. In the Y360 dataset, edges represents the friendship relationship among users.

The graphs sizes vary from 226 413 vertices of dblp, 588 166 of flickr, to 1 226 311 of Y360, with different densities; Y360 is the largest but also the sparsest dataset. The main statistics (as defined in Section 6) of the three datasets are reported in Table 4.

7.1 Parameter Tuning and Running Time

In our first set of experiments, we considered three obfuscation levels, $k \in \{20, 60, 100\}$, and two possible tolerance values, $\varepsilon \in \{10^{-3}, 10^{-4}\}$. We experimented with different values for q and c (with $q \in \{0.01, 0.05, 0.1\}$ and $c \in \{2, 3\}$), but here we present only the case $q = 0.01$ and $c = 2$ (except for two instances that will be discussed below). In Table 2, we report the minimal values of σ, as found by Algorithm 1, that yielded a (k, ε)-obfuscation for given values of k and ε.

As expected, larger k or smaller ε required larger values of σ, because more noise was needed in order to reach the desired level of obfuscation. In some cases, Algorithm 1 failed to find a proper upper bound for σ in the loop in

[2] http://www.informatik.uni-trier.de/~ley/db/
[3] http://www.flickr.com/

Table 4: The sample mean on a sample of size 100, with $\varepsilon = 10^{-4}$. The last column is the average (over all statistics) of the relative absolute difference between the sample mean and the real value of the statistics.

	graph	S_{NE}	S_{AD}	S_{MD}	S_{DV}	S_{PL}	S_{APD}	S_{DiamLB}	S_{EDiam}	S_{CL}	S_{CC}	rel.err.
dblp	real	716 460	6.33	238	76.79	−0.046	7.34	25	8.78	6.96	0.38	
	$k = 20$	713 952	6.31	233	76.18	−0.046	7.01	22.59	7.16	6.68	0.35	0.049
	$k = 60$	735 766	6.50	652	122.8	−0.014	6.05	20.52	6.29	5.76	0.23	0.429
	$k = 100$	754 776	6.67	975	187.6	−0.008	5.67	19.12	6.00	5.41	0.16	0.705
flickr	real	5 801 442	19.73	6 660	6 200	−0.002	5.03	21	5.43	4.80	0.12	
	$k = 20$	5 921 470	20.14	5 847	6 924	−0.002	4.84	20.51	4.80	4.64	0.05	0.112
	$k = 60$	6 944 481	23.61	4 534	12 847	−0.002	4.59	17.66	4.47	4.42	0.04	0.322
	$k = 100$	7 640 446	25.98	6 121	18 438	−0.001	4.50	16.81	4.33	4.37	0.06	0.415
Y360	real	2 618 645	4.27	258	112.6	−0.027	8.21	31	8.94	7.77	0.04	
	$k = 20$	2 605 027	4.25	257	109.5	−0.028	8.06	31.53	9.19	7.66	0.03	0.026
	$k = 60$	2 605 952	4.25	256	110.0	−0.028	8.05	30.04	8.95	7.64	0.03	0.025
	$k = 100$	2 609 937	4.26	259	111.9	−0.027	8.01	31.64	8.99	7.60	0.03	0.023

Table 5: The relative sample standard error of the mean (SEM) on a sample of size 100, with $\varepsilon = 10^{-4}$ (the other parameters are set as in Table 2). For every statistics, the value shown is the sample standard deviation, divided by the square root of the sample size and normalized by the sample mean. The last column is the average of the relative sample standard errors over all of the statistics.

	k	S_{NE}	S_{AD}	S_{MD}	S_{DV}	S_{PL}	S_{APD}	S_{DiamLB}	S_{EDiam}	S_{CL}	S_{CC}	average
dblp	20	0.00010	0.00010	0.0120	0.00100	0.0110	0.0040	0.041	0.10	0.020	0.013	0.019
	60	0.00024	0.00024	0.0260	0.00350	0.0170	0.0035	0.058	0.16	0.019	0.018	0.028
	100	0.00029	0.00029	0.0170	0.00430	0.0170	0.0033	0.055	0.15	0.018	0.024	0.027
flickr	20	0.00016	0.00016	0.0067	0.00074	0.0037	0.0036	0.060	0.15	0.016	0.045	0.028
	60	0.00018	0.00018	0.0100	0.00068	0.0030	0.0039	0.084	0.17	0.018	0.054	0.033
	100	0.00017	0.00017	0.0064	0.00059	0.0032	0.0039	0.082	0.18	0.018	0.035	0.032
Y360	20	0.00004	0.00004	0.0024	0.00025	0.0035	0.0036	0.043	0.13	0.021	0.045	0.027
	60	0.00004	0.00004	0.0049	0.00031	0.0032	0.0046	0.051	0.15	0.021	0.061	0.031
	100	0.00005	0.00005	0.0120	0.00044	0.0044	0.0035	0.052	0.16	0.018	0.057	0.032

Lines 3-6. In those cases, increasing the parameter c to 3 resolved the problem.

The obfuscation algorithm was implemented in Java and run on an Intel Xeon X5660 CPUs, 2.80 GHz, 12 MB cache size. Table 3 reports the running times (expressed in edges per second) of the same experiments for which we reported in Table 2 the values of σ. As explained above, we used in all cases $q = 0.01$ and $c = 2$, except for the two cases marked by $(*)$ in which $c = 3$. We note that using smaller values of c has the benefit of keeping the graph size under control; such a benefit is of special importance for large networks. Smaller values of c also reduce the runtime of Algorithm 2, where the main loop (Lines 13-19) is over $c|E|$ edges. This effect is evident in Table 3, where the performance drops substantially in the two cases where $c = 3$. As expected, the performance slightly decreases when k increases or ε decreases, due to the increased efforts to achieve a higher obfuscation level. We note that the smaller computation times required for `Y360` are due to the fact that this dataset turns out to be easier to obfuscate than the others (as witnessed also by the small final values of σ as reported in Table 2).

The parameter q just introduces some amount of "white noise" in the graph. Using higher values of q enhances obfuscation but it also reduces the utility of the final released graph. Due to space limitations, we present only results for $q = 0.01$. A more elaborated set of plots, for different settings of q and other obfuscation parameters, will be given in an extended version of this paper[4].

[4]A complete set of plots, along with the code of Algorithm 1, is available at `http://boldi.dsi.unimi.it/obfuscation/`.

7.2 Data Utility

Next, we computed statistics of interest on the obfuscated graphs, using the sampling method (Section 6.1).[5] For every obfuscated graph, we sampled 100 possible worlds and for each of them we computed all the scalar statistics listed above. The mean values obtained are shown in Table 4. Those values are very concentrated, as witnessed by Table 5, that reports the relative sample standard error of the mean (also called SEM; it is obtained as the sample standard deviation divided by the square root of the sample size and normalized by the sample mean); the last column reports the average computed over all the statistics. As can be seen, all statistics are very well concentrated; on average, the fluctuations for all statistics are of about 3% (last column of Table 5), but most of them (see, for example, S_{NE} or S_{AD}) exhibit a much higher level of concentration. There is a weak dependence on k and also on ε (the latter dependence is not shown here).

We proceed to comparing the sample mean of the statistics obtained with their real values on the original graph (see again Table 4). The quality of the estimation decreases when obfuscation becomes larger: in the last column of the table, we computed the average statistical error over all scalar statistics, that is, the relative absolute difference between the estimate and the real value. With small values of k, e.g., $k = 20$, the error is always well below 15%; larger values of k introduce larger errors, up to 70.5% when $k = 100$

[5]For S_{NE} and S_{AD} we use the exact formulas (Sec. 6.2). The results are almost identical to those obtained by sampling.

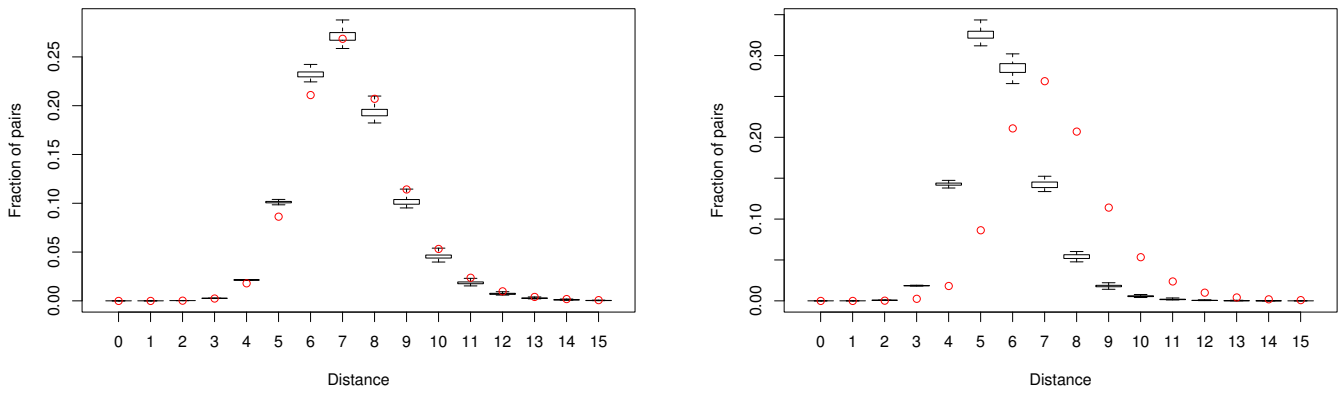


Figure 2: The distribution of pairwise distances S_{PDD}; the small (red) dots correspond to the distribution in the real `dblp` graph; the boxplots give the distributions for the case $k = 20$, $\varepsilon = 10^{-3}$ (left) and $k = 100$, $\varepsilon = 10^{-4}$ (right). As usual, the two whiskers represent the smallest and largest values observed across the samples, whereas the box represents the range between the lower and the upper quartiles.)

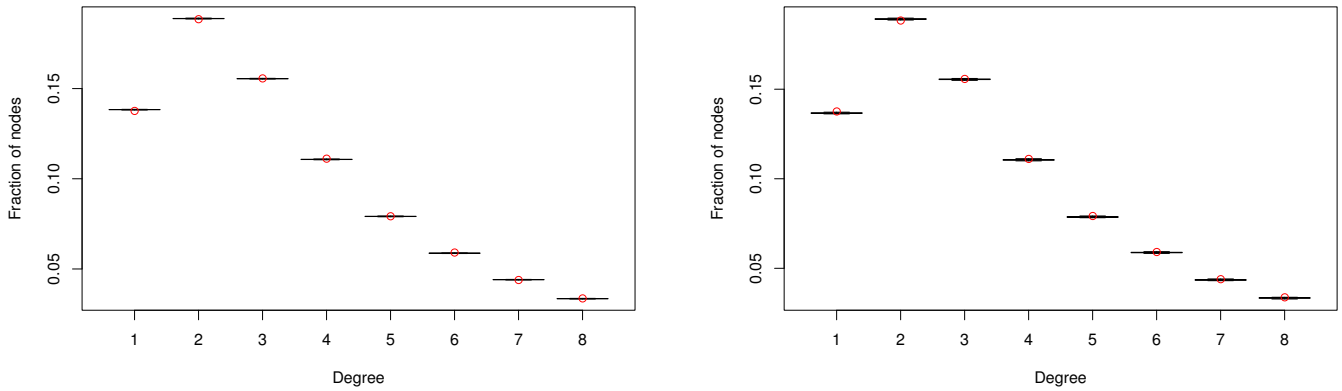


Figure 3: The distribution of degrees S_{DD}; the small (red) dots correspond to the distribution in the real `dblp` graph; the boxplots give the distributions for the case $k = 20$, $\varepsilon = 10^{-3}$ (left) and $k = 100$, $\varepsilon = 10^{-4}$ (right).

in the `dblp` dataset. Observe that some statistics (e.g., degree variance or clustering coefficient) are more affected by error than others.

The behavior described for scalar statistics is also observed with vector statistics. For example, Figure 2 shows S_{PDD} (the distribution of the pairwise distances) in the original `dblp` and in two obfuscated versions. Here, two extreme cases are presented: For $k = 20$ and $\varepsilon = 10^{-3}$ the distribution obtained is qualitatively very similar (as witnessed also by the scalar distance-based statistics in Table 4); conversely, for $k = 100$ and $\varepsilon = 10^{-4}$, the estimated distribution is quite far from the original one. In Figure 3 we present a similar plot for the degree distribution: for every degree, we considered the distribution of the frequency of that degree across all possible worlds. In this case, the approximation is very concentrated and its mean almost coincides with the real degree frequency, even for $k = 100$ and $\varepsilon = 10^{-4}$.

7.3 Comparative Evaluation

We finally compare our proposed method with random-perturbation methods that publish a standard graph (in articular the methods described by Bonchi et al. [4]):

- *random sparsification:* given a parameter p, each edge $e \in E$ is removed from the graph with probability p;
- *random perturbation:* given a parameter p, first each edge $e \in E$ is removed from the graph with probability p, then each non-existing edge in $V_2 \setminus E$ is added with probability $\frac{p|E|}{\binom{|V|}{2}-|E|}$.

To make the comparison possible, we must first determine which value of the parameter p used in these obfuscation algorithms corresponds to which pair (k, ε) of obfuscation parameters. The appropriate values can be deduced by the anonymity level plots of the sparsified or perturbed graph obtained with a certain value of p: of course, any such graph will correspond to many pairs of parameters (k, ε); for example, given any fixed ε, an appropriate k can be determined by disregarding the εn vertices with smallest anonymity and letting k be the least anonymity of the remaining vertices.

Figure 4 shows the obfuscation levels obtained for some of the parameter combinations on `dblp` and `flickr`. The plot shows, for every obfuscation level k, the number of vertices that have obfuscation level less than or equal to k. The two rectangles appearing in the plot highlight the obfuscation requirements (k, ε). Figure 4 shows, for example,

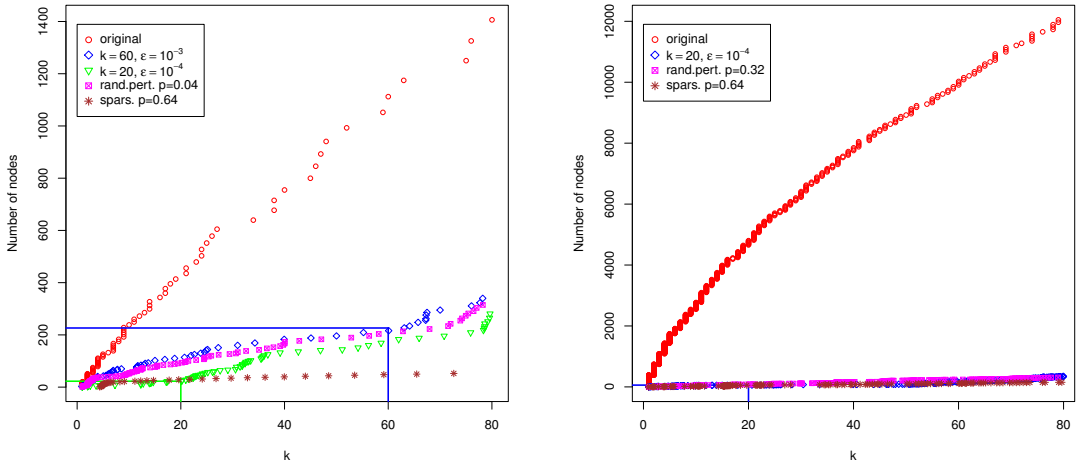


Figure 4: Comparison of the anonymity levels obtained for `dblp` (left) and `flickr` (right) using obfuscation, random perturbation and sparsification, for the parameter choices described in Section 7.3. The plot shows, for every obfuscation level k, the number of vertices that have obfuscation level less than or equal to k.

Table 6: Comparison between obfuscation by uncertainty and obfuscation by random sparsification and perturbation.

	graph	S_{NE}	S_{AD}	S_{MD}	S_{DV}	S_{PL}	S_{APD}	S_{DiamLB}	S_{EDiam}	S_{CL}	S_{CC}	rel. err.
dblp	original	716 460	6.33	238	76.79	−0.046	7.34	25	8.78	6.96	0.38	
	rand.pert. ($p = 0.04$)	716 393	6.33	230	71.26	−0.048	7.09	18.55	7.25	6.85	0.36	0.071
	obf. ($k = 60$, $\varepsilon = 10^{-3}$)	713 819	6.31	236	75.86	−0.046	7.15	22.75	7.21	6.82	0.36	0.043
	rand.spars. ($p = 0.64$)	257 890	2.28	93	11.40	−0.124	10.24	36.72	10.60	25.77	0.13	0.921
	obf. ($k = 20$, $\varepsilon = 10^{-4}$)	713 952	6.31	233	76.18	−0.046	7.01	22.59	7.16	6.68	0.35	0.050
flickr	original	5 801 442	19.73	6 660	6 200	−0.002	5.03	21	5.43	4.80	0.12	
	rand.pert. ($p = 0.64$)	5 801 229	19.73	2 407	820.3	−0.0059	4.55	7.02	4.15	4.49	0.030	0.497
	rand.spars. ($p = 0.32$)	3 944 902	13.41	4 526	2 871	−0.003	5.24	19.56	4.91	6.69	0.079	0.286
	obf. ($k = 20$, $\varepsilon = 10^{-4}$)	5 921 470	20.14	5 847	6 924	−0.002	4.84	20.51	4.81	4.64	0.050	0.112

that a random perturbation of `dblp` with $p = 0.04$ matches obfuscation $(k = 60, \varepsilon = 10^{-3})$.

We here present the comparative results in the following cases:[6]

- `dblp` with random perturbation using $p = 0.04$, matching $k = 60$ and $\varepsilon \approx 10^{-3}$;
- `dblp` with sparsification using $p = 0.64$, matching $k = 20$ and $\varepsilon \approx 10^{-4}$;
- `flickr` with random perturbation using $p = 0.32$ and with sparsification using $p = 0.64$, both corresponding to $k = 20$ with $\varepsilon \approx 10^{-4}$.

For each of the two obfuscation techniques presented in [4], we produced 50 samples; note that in those probabilistic methods, the obfuscation is a certain graph. Then we computed the statistics on each sample, and proceeded in the same way as we did for the obfuscated graph.

Table 6 shows the results of the comparison. In all cases, the quality of the statistics as computed with our obfuscation method is much better; in one case, the relative error is 5% instead of the 92% imposed by sparsification to obtain the same level of obfuscation. Therefore, we can safely conclude that our experimental assessment on real-world graphs confirms the initial and driving intuition underlying this paper: by using finer-grained perturbation operations, such as only perturbing *partially* the existence of an edge, one can achieve the same desired level of obfuscation with smaller changes in the data than when completely removing or adding edges, thus maintaining higher data utility.

[6]The values of p used here ($p \in \{0.04, 0.32, 0.64\}$) are the same as those used by Bonchi et al. [4].

8. CONCLUSIONS AND FUTURE WORK

We introduce a new approach for identity obfuscation in graph data. In the proposed approach, the desired obfuscation is obtained by injecting uncertainty in the social graph and publishing the resulting uncertain graph. Our proposal can be seen as a generalization of random perturbation methods for identity obfuscation in graphs, as it enables finer-grained perturbations than fully removing or fully adding edges. Such increased flexibility in spreading the noise over the edges of the graph enables achieving the same level of obfuscation with smaller changes in the data, as confirmed by our experiments on real-world graphs.

While the results that we achieve are most encouraging, this work represents only a first step in a promising research direction. As it is often the case, new privacy-enabling techniques create novel attacks that, in turn, propel stronger protection mechanisms. Therefore, in our future investigation we plan to extend and strengthen this line of research by further assessing its limits and merits.

One interesting research direction is to investigate how to extend our uncertainty-based approach in order to release networks with additional information, besides the mere graph data, such as vertex attributes [22], communication logs among users, information-propagation traces, and other types of social dynamics. Another case of particular interest is that of a sequential release of a social network. In a recent paper, Medforth and Wang [21] demonstrated the risks of publishing a sequence of releases of the same network. In particular, they described the *degree-trail attack*, by which the vertex belonging to a target user can be re-identified from a sequence of published graphs, by comparing the degrees of the vertices in the published graphs with the degree evolution of the target. The applicability of the degree-trail attack to our probabilistic graph release is an open research question.

Acknowledgments. This research was partially supported by the Torres Quevedo Program of the Spanish Ministry of Science and Innovation, co-funded by the European Social Fund, and by the Spanish Centre for the Development of Industrial Technology under the CENIT program, project CEN-20101037, "Social Media" (`http://www.cenitsocialmedia.es/`). Part of the work was done while P. Boldi and T. Tassa were visiting Yahoo! Research.

9. REFERENCES

[1] O. Abul, F. Bonchi, and M. Nanni. *Never walk alone*: Uncertainty for anonymity in moving objects Databases. In *ICDE*, pages 376–385, 2008.

[2] L. Backstrom, C. Dwork, and J. M. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *WWW*, pages 181–190, 2007.

[3] P. Boldi, M. Rosa, and S. Vigna. HyperANF: Approximating the neighbourhood function of very large graphs on a budget. In *WWW*, pages 625–634, 2011.

[4] F. Bonchi, A. Gionis, and T. Tassa. Identity obfuscation in graphs through the information theoretic lens. In *ICDE*, pages 924–935, 2011.

[5] A. Campan and T. Truta. A clustering approach for data and structural anonymity in social networks. In *PinKDD*, pages 33–54, 2008.

[6] E. Cohen. Size-estimation framework with applications to transitive closure and reachability. *Journal of Computer and System Sciences*, 55(3):441–453, 1997.

[7] G. Cormode, D. Srivastava, S. Bhagat, and B. Krishnamurthy. Class-based graph anonymization for social network data. *PVLDB*, 2(1):766–777, 2009.

[8] G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. *PVLDB*, 1(1):833–844, 2008.

[9] P. Crescenzi, R. Grossi, L. Lanzi, and A. Marino. A comparison of three algorithms for approximating the distance distribution in real-world graphs. In *TAPAS*, pages 92–103, 2011.

[10] M. Hay, G. Miklau, D. Jensen, D. F. Towsley, and C. Li. Resisting structural re-identification in anonymized social networks. *VLDB Journal*, 19(6):797–823, 2010.

[11] M. Hay, G. Miklau, D. Jensen, D. F. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. In *PVLDB*, 1(1):102–114, 2008.

[12] M. Hay, G. Miklau, D. Jensen, P. Weis, and S. Srivastava. Anonymizing social networks. *University of Massachusetts Technical Report*, 07(19), 2007.

[13] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13-30, 1963.

[14] R. Jin, L. Liu, and C. C. Aggarwal. Discovering highly reliable subgraphs in uncertain graphs. In *KDD*, pages 992–1000, 2011.

[15] R. Jin, L. Liu, B. Ding, and H. Wang. Distance-constraint reachability computation in uncertain graphs. *PVLDB*, 4(9):551–562, 2011.

[16] O. Kallenberg. *Foundations of Modern Probability*. Springer Series in Statistics, second edition, 2002.

[17] U. Kang, C. E. Tsourakakis, A. P. Appel, C. Faloutsos, and J. Leskovec. HADI: Mining radii of large graphs. *ACM Transactions on Knowledge Discovery from Data*, 5(2):8, 2011.

[18] R. J. Lipton and J. F. Naughton. Query size estimation by adaptive sampling. *Journal of Computer and System Sciences*, 51(1):18–25, 1995.

[19] K. Liu and E. Terzi. Towards identity anonymization on graphs. In *SIGMOD Conference*, pages 93–106, 2008.

[20] M. Marchiori and V. Latora. Harmony in the small-world. *Physica A*, 285:539–546, 2000.

[21] N. Medforth and K. Wang. Privacy risk in graph stream publishing for social network data. In *ICDM*, pages 437–446, 2011.

[22] A. Narayanan and V. Shmatikov. De-anonymizing social networks. In *IEEE Symposium on Security and Privacy*, pages 173–187, 2009.

[23] C. R. Palmer, P. B. Gibbons, and C. Faloutsos. Anf: a fast and scalable tool for data mining in massive graphs. In *KDD*, pages 81–90, 2002.

[24] M. Potamias, F. Bonchi, A. Gionis, and G. Kollios. k-Nearest neighbors in uncertain graphs. *PVLDB*, 3(1):997–1008, 2010.

[25] P. Samarati and L. Sweeney. Generalizing data to provide anonymity when disclosing information (abstract). In *PODS*, page 188, 1998.

[26] J. Shao and D. Tu. *The jackknife and bootstrap*. Springer series in statistics, 1995.

[27] T. Snijders. The degree variance: An index of graph heterogeneity. *Social Networks*, 3(3):163–174, 1981.

[28] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal on Uncertainty Fuzziness and Knowledge-based Systems*, 10(5):571-588, 2002.

[29] T. Tassa and D. Cohen. Anonymization of centralized and distributed social networks by sequential clustering. *IEEE Transactions on Knowledge and Data Engineering*, 2012.

[30] B. Thompson and D. Yao. The union-split algorithm and cluster-based anonymization of social networks. In *ASIACCS*, pages 218–227, 2009.

[31] W. Wu, Y. Xiao, W. Wang, Z. He, and Z. Wang. k-Symmetry model for identity anonymization in social networks. In *EDBT*, pages 111–122, 2010.

[32] X. Ying, K. Pan, X. Wu, and L. Guo. Comparisons of randomization and k-degree anonymization schemes for privacy preserving social network publishing. In *SNA-KDD*, pages 1–10, 2009.

[33] E. Zheleva and L. Getoor. Preserving the privacy of sensitive relationship in graph data. In *PinKDD*, pages 153–171, 2007.

[34] B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In *ICDE*, pages 506-515, 2008.

[35] L. Zou, L. Chen, and M. T. Özsu. K-automorphism: A general framework for privacy preserving network publication. *PVLDB*, 2(1):946–957, 2009.

[36] Z. Zou, H. Gao, and J. Li. Discovering frequent subgraphs over uncertain graph databases under probabilistic semantics. In *KDD*, pages 633–642, 2010.

[37] Z. Zou, J. Li, H. Gao, and S. Zhang. Finding top-k maximal cliques in an uncertain graph. In *ICDE*, pages 649–652, 2010.

[38] Z. Zou, J. Li, H. Gao, and S. Zhang. Mining frequent subgraph patterns from uncertain graph data. *IEEE Transactions on Knowledge and Data Engineering*, 22(9):1203–1218, 2010.

Axioms for Centrality

Paolo Boldi Sebastiano Vigna*
Dipartimento di informatica, Università degli Studi di Milano, Italy

August 9, 2013

Abstract

Given a social network, which of its nodes are more central? This question has been asked many times in sociology, psychology and computer science, and a whole plethora of *centrality measures* (a.k.a. *centrality indices*, or *rankings*) were proposed to account for the importance of the nodes of a network. In this paper, we try to provide a mathematically sound survey of the most important classic centrality measures known from the literature and propose an *axiomatic* approach to establish whether they are actually doing what they have been designed for. Our axioms suggest some simple, basic properties that a centrality measure should exhibit.

Surprisingly, only a new simple measure based on distances, *harmonic centrality*, turns out to satisfy all axioms; essentially, harmonic centrality is a correction to Bavelas's classic *closeness centrality* [4] designed to take unreachable nodes into account in a natural way.

As a sanity check, we examine in turn each measure under the lens of information retrieval, leveraging state-of-the-art knowledge in the discipline to measure the effectiveness of the various indices in locating web pages that are relevant to a query. While there are some examples of this comparisons in the literature, here for the first time we take into consideration centrality measures based on distances, such as closeness, in an information-retrieval setting. The results match closely the data we gathered using our axiomatic approach.

Our results suggest that centrality measures based on distances, which have been neglected in information retrieval in favour of spectral centrality measures in the last years, are actually of very high quality; moreover, harmonic centrality pops up as an excellent general-purpose centrality index for arbitrary directed graphs.

1 Introduction

In the last years, there has been an ever-increasing research activity in the study of real-world complex networks [51] (the world-wide web, the autonomous-systems graph within the Internet, coauthorship graphs, phone call graphs, email graphs and biological networks, to cite but a few). These networks, typically generated directly or indirectly by human activity and interaction (and therefore hereafter dubbed "social"), appear in a large variety of contexts and often exhibit a surprisingly similar structure. One of the most important notions that researchers have been trying to capture in such networks is "node centrality": ideally, every node (often representing an individual) has some degree of influence or importance within the social domain under consideration, and one expects such importance to be reflected in the structure of the social network; centrality is a quantitative measure that aims at revealing the importance of a node.

Among the types of centrality that have been considered in the literature (see [12] for a good survey), many have to do with distances between nodes.[1] Take, for instance, a node in an undirected

*The authors have been supported by the EU-FET grant NADINE (GA 288956).

[1] Here and in the following, by "distance" we mean the length of a shortest path between two nodes.

connected network: if the sum of distances to all other nodes is large, the node under consideration is *peripheral*; this is the starting point to define Bavelas's *closeness centrality* [4] which is the reciprocal of peripherality (i.e., the reciprocal of the sum of distances to all other nodes).

The role played by shortest paths is justified by one of the most well-known features of complex networks, the so-called *small-world* phenomenon. A small-world network [17] is a graph where the average distance between nodes is logarithmic in the size of the network, whereas the clustering coefficient is larger (that is, neighbourhoods tend to be denser) than in a random Erdős-Rényi graph with the same size and average distance.[2] The fact that social networks (whether electronically mediated or not) exhibit the small-world property is known at least since Milgram's famous experiment [38] and is arguably the most popular of all features of complex networks. For instance, the average distance of the Facebook graph was recently established to be just 4.74 [3].

The purpose of this paper is to pave the way for a formal well-grounded assessment of centrality measures, based on some simple guiding principles; we seek notions of centrality that are at the same time *robust* (they should be applicable to arbitrary directed graphs, possibly non-connected, without modifications) and *understandable* (they should have a clear combinatorial interpretation).

With these principles in mind, we shall present and compare the most popular and well-known centrality measures proposed in the last decades. The comparison will be based on a set of *axioms*, each trying to capture a specific trait.

In the last part of the paper, as a sanity check, we compare the measures we discuss in an information-retrieval settings, using the classic GOV2 collection to extract documents satisfying a query and ranking the resulting induced subgraph of relevant documents based solely on centrality.

The results are somehow surprising, and suggest that simple measures based on distances, and in particular *harmonic centrality* (which we introduce formally in this paper) can give better results than some of the most sophisticated indices used in the literature. These unexpected outcomes are the main contribution of this paper, together with the set of axiom we propose, which provide a conceptual framework for understanding centrality measures in a formal way. We also try to give an orderly account of centrality in social and network sciences, gathering scattered results and folklore knowledge in a systematic way.

2 A Historical Account

In this section we sketch the historical development of centrality, focusing on the ten classical centrality measures that we decided to include in this paper: the overall growth of the field is of course much more complex, and the literature contains a myriad of alternative proposals that will not be discussed here.

Centrality is a fundamental tool in the study of social networks: the first efforts to define formally centrality indices were attempted in the late 1940s by the Group Networks Laboratory at M.I.T. directed by Alex Bavelas [4], in the framework of communication patterns and group collaboration [30, 5]; those pioneering experiments all concluded that centrality was related to group efficiency in problem-solving, and agreed with the subjects' perception of leadership. In the following decades, various measures of centrality were employed in a multitude of contexts (to understand political integration in Indian social life [18], to examine the consequences of centrality in communication paths for urban development [45], to analyse their implications to the efficient design of organizations [7, 34], or even to explain the wealth of the Medici family based on their central position with respect to marriages and financial transactions in the 15th century Florence [42]). We can certainly say that the problem of singling out influential individuals in a social group is a holy grail that sociologists have been trying to capture for at least fifty years.

[2] The reader might find this definition a bit vague, and some variants are often spotted in the literature: this is a general well-known problem, also highlighted recently, for example in [32].

Although all researchers agree that centrality is an important structural attribute of social networks, and that it is directly related to other important group properties and processes, there is no consensus on exactly *what* centrality is or on its conceptual foundations, and there is very little agreement on the proper procedures for its measurement [15, 21]: as Freeman observed, "several measures are often only vaguely related to the intuitive ideas they purport to index, and many are so complex that it is difficult or impossible to discover what, if anything, they are measuring" [21].

Freeman acutely remarks that the implicit starting point of all centrality measures is the same: the central node of a star should be deemed more important than the other vertices; paradoxically, it is precisely the unanimous agreement on this requirement that may have produced quite different approaches to the problem. In fact, the center of a star is at the same time

1. the node with largest degree;
2. the node that is closest to the other nodes (e.g., that has the smallest average distance to other nodes);
3. the node through which most shortest paths pass;
4. the node with the largest number of incoming paths of length k, for every k;
5. the node that maximizes the dominant eigenvector of the graph matrix;
6. the node with highest probability in the stationary distribution of the natural random walk on the graph.

These observations lead to corresponding (competing) views of centrality. Degree is probably the oldest kind of measure of importance ever used, being equivalent to majority voting in elections (where $x \to y$ is interpreted as "x voted for y").

The most classical notion of *closeness*, instead, was introduced by Bavelas [4] for undirected, connected networks as the reciprocal of the sum of distances from a given node. Closeness was originally aimed at establishing how much a vertex can communicate without relying on third parties for his messages to be delivered.[3] In the seventies, Nan Lin proposed to adjust the definition of closeness so to make it usable on directed networks that are not necessarily strongly connected [33].

Centrality indices based on the count of shortest paths were formally developed independently by Anthonisse [2] and Freeman [22], who introduced *betweenness* as a measure of the probability that a random shortest path passes through a given node or edge.

Katz's index [27] is based instead on a weighted count of *all* paths coming into a node: more precisely, the weight of a path of length t is β^t, for some *attenuation factor* β, and the score of x is the sum of the weights of all paths coming into x. Of course, β must be chosen so that all the summations converge.

While the above notions of centrality are combinatorial in nature, and based on the discrete structure of the underlying graph, another line of research studies *spectral* techniques (in the sense of linear algebra) to define a measure of centrality.

The earliest known proposal of this kind is due to Seeley [46], who normalized to sum one the row of an adjacency matrix representing the "I like him" relations among a group of children, and assigned a centrality score using the resulting dominant eigenvector. This is actually equivalent to studying the stationary distribution of the Markov chain defined by the natural random walk on the graph. Few years later, Wei [52] proposed the dominant eigenvector of suitable matrices to rank sport teams.

Curiously enough, the most famous among spectral centrality scores is also one of the most recent, PageRank [43]: PageRank was a centrality measure specifically geared toward web graphs, and it was

[3] The notion can also be generalized to a weighted summation of node contributions multiplied by some *discount* functions applied to their distance to a given node [16].

introduced precisely with the aim of implementing it in a search engine (specifically, Google, that the authors of PageRank founded in 1997).

In the same span of years, Jon Kleinberg defined another centrality measure (actually, a ranking algorithm) called HITS [28] (for "Hyperlink-Induced Topic Search"). The idea[4] is that every node of a graph is associated with two importance indices: one (called "authority score") measures how reliable (important, authoritative...) a node is, and another (called "hub score") measures how good the node is in pointing to authoritative nodes, with the two scores mutually reinforcing each other. The result is again the dominant eigenvector of a suitable matrix. SALSA [31] is a more recent and strictly related score based on the same idea, with the difference that it applies some normalization to the matrix.

3 Definitions and conventions

In this paper we consider directed graphs defined by a set N of n nodes and a set $A \subseteq N \times N$ of arcs; we write $x \to y$ when $\langle x, y \rangle \in A$ and call x and y the source and target of the arc, respectively. An arc with the same source and target is called a *loop*.

The *transpose* of a graph is obtained by reversing all arc directions (i.e., it has an arc $y \to x$ for all arcs $x \to y$ of the original graph). A *symmetric graph* is a graph such that $x \to y$ whenever $y \to x$; such a graph is fixed by transposition, and can be identified with a undirected graph, that is, a graph whose arcs are a subset of unordered pairs of nodes (usually called "edges"). A *successor* of x is a node y such that $x \to y$, and a *precedessor* of x is a node y such that $y \to x$. The *outdegree* $d^+(x)$ of a node x is the number of its successors, and the *indegree* $d^-(x)$ is the number of its predecessors.

A *path* (of length k) is a sequence $x_0, x_1, \ldots, x_{k-1}$, where $x_j \to x_{j+1}$, $0 \leq j < k$. A *walk* (of length k) is a sequence $x_0, x_1, \ldots, x_{k-1}$, where $x_j \to x_{j+1}$ or $x_{j+1} \to x_j$, $0 \leq j < k$. A (strongly) connected *component* of a graph is a maximal subset in which every pair of nodes is connected by a walk (path). Components form a partition of the nodes of a graph. A graph is *(strongly) connected* if there is a single (strongly) connected component, that is, for every choice of x and y there is a walk (path) from x to y. A strongly connected component is *terminal* if its nodes have no arc towards other components.

The *distance* $d(x, y)$ from x to y is the length of a shortest path from x to y, or ∞ if no such path exists. The nodes *reachable* from x are the nodes y such that $d(x, y) < \infty$. The nodes *coreachable* from x are the nodes y such that $d(y, x) < \infty$. A node has *trivial* (co)reachable set if the latter contains only the node itself.

The notation $\bar{A}$, where A is a nonnegative matrix, will be used throughout the paper to denote the matrix obtained by ℓ_1-normalizing the rows of A, that is, dividing each element of a row by the sum of the row (null rows are left unchanged). If there are no null rows, $\bar{A}$ is *stochastic*, that is, it is nonnegative and the row sums are all equal to one.

We use Iverson's notation: if P is a predicate, $[P]$ has value 0 if P is false and 1 if P is true [29]; finally, we denote with H_i the i-th harmonic number $\sum_{1 \leq k \leq i} 1/k$.

3.1 Geometric measures

We call *geometric* those measures assuming that importance is a function of the distances. These are actually some of the oldest measures defined in the literature.

[4] To be true, Kleinberg's algorithm works in two phases; in the first phase, one selects a subgraph of the starting webgraph based on the pages that match the given query; in the second phase, the centrality score is computed on the subgraph. Since in this paper we are looking at HITS simply as a centrality index, will simply apply it to the graph under examination.

3.1.1 Indegree

Indegree, the number of incoming arcs $d^-(x)$, can be considered a geometric measure: it is simply the number of nodes at distance one[5]. It is probably the oldest kind of measure of importance ever used, as it is equivalent to majority voting in elections (where $x \to y$ if x voted for y). Indegree has a number of obvious shortcomings (e.g., it is easy to spam), but it is actually a good baseline, and in some cases turned out to provide better results than more sophisticated methods (see, e.g., [19]).

3.1.2 Closeness

Closeness was introduced by Bavelas in the late forties [6]; the closeness of x is defined by

$$\frac{1}{\sum_y d(y,x)}. \tag{1}$$

The intuition behind closeness is that nodes that are more central have smaller distances, and thus a smaller denominator, resulting in a larger centrality. We remark that for this definition to make sense, the graph must be strongly connected. Lacking that condition, some of the denominators will be ∞, resulting in a rank of zero for all nodes which cannot coreach the whole graph.

It was not probably in Bavelas's intentions to apply the measure to directed graphs, and even less to graph with infinite distances, but nonetheless closeness is sometimes "patched" by simply not including unreachable nodes, that is,

$$\frac{1}{\sum_{d(y,x)<\infty} d(y,x)},$$

and assuming that nodes with an empty coreachable set have centrality 0 by definition: this is actually the definition we shall use in the rest of the paper. These apparently innocuous adjustments, however, introduce a strong bias toward nodes with a small coreachable set.

3.1.3 Lin's index

Nan Lin [33] tried to repair the definition of closeness for graphs with infinite distances by weighting closeness using the square of the number of coreachable nodes; his definition for the centrality of a node x with a nonempty coreachable set is

$$\frac{\left|\{y \mid d(y,x) < \infty\}\right|^2}{\sum_{d(y,x)<\infty} d(y,x)}.$$

The rationale behind this definitions is the following: first, we consider closeness not the inverse of a sum of distances, but rather the inverse of the *average* distance, which entails a first multiplication by the number of coreachable nodes. This change normalizes closeness across the graph. Now, however, we want nodes with a larger coreachable set to be more important, given that the average distance is the same, so we multiply again by the number of coreachable nodes. Nodes with an empty coreachable set have centrality 1 by definition.

Lin's index was (somewhat surprisingly) ignored in the following literature. Nonetheless, it seems to provide a reasonable solution for the problems caused by the definition of closeness.

[5]Most centrality measures proposed in the literature were actually described only for undirected, connected graphs. Since the study of web graphs and online social networks has posed the problem of extending centrality concepts to networks that are directed, and possibly not strongly connected, in the rest of this paper we consider measures depending on the *incoming* arcs of a node (e.g., incoming paths, left dominant eigenvectors, distances from all nodes to a fixed node). If necessary, these measures can be called "negative", as opposed to the "positive" versions obtained by taking the transpose of the graph.

3.1.4 Harmonic centrality

As we noticed, the main problem of closeness lies in the presence of pairs of unreachable nodes. We thus get inspiration from Marchiori and Latora [35], who, faced with the problem of providing a sensible notion of "average shortest path" for a generic directed network, propose to replace the average distance with the *harmonic mean of all distances*. Indeed, in case a large number of pairs of nodes are not reachable, the average distance between reachable pairs can be misleading: a graph might have a very low average distance, while it is almost completely disconnected (e.g., a perfect matching has average distance exactly one). The harmonic mean has the useful property of handling ∞ cleanly (assuming, of course, that $\infty^{-1} = 0$). For example, a perfect matching has harmonic mean of distances $n - 1$.

In general, for each graph-theoretical notion based on arithmetic averaging or maximization there is an equivalent notion based on the harmonic mean. If we consider closeness the reciprocal of a denormalized average of distances, it is natural to consider also the reciprocal of a denormalized harmonic mean of distances. We thus define the *harmonic centrality* of x as[6]

$$\sum_{y \neq x} \frac{1}{d(y,x)} = \sum_{d(y,x)<\infty, y \neq x} \frac{1}{d(y,x)}. \tag{2}$$

The difference with (1) might seem minor, but actually it is a radical change. Harmonic centrality is strongly correlated to closeness centrality in simple networks, but naturally also accounts for nodes y that cannot reach x. Thus, it can be fruitfully applied to graphs that are not strongly connected.

3.2 Spectral measures

Spectral measures compute the left dominant eigenvector of some matrix derived from the graph, and depending on how the matrix is modified before the computation we can obtain a number of different measures. Existence and uniqueness of such measures is usually derivable by the theory of nonnegative matrices started by Perron and Frobenius [8]; we will however avoid to discuss such issues, as there is a large body of established literature about the topic. All observations in this section are true for strongly connected graphs; the modifications for graphs that are not strongly connected can be found in the cited references.

3.2.1 The left dominant eigenvector

The first and most obvious spectral measure is the left dominant eigenvector of the adjacency matrix. Indeed, the dominant eigenvector can be thought as the fixed point of an iterated computation in which every node starts with the same score, and then updates its score with the sum of its predecessors. The vector is then normalized, and the process repeated until convergence.

The usage of dominant eigenvectors to find important nodes in matrices of entities can be traced at least back to Wei's Master thesis [52]. Wei's thesis was then popularized by Kendall, and the technique is actually known in the literature about ranking of sport teams as "Kendall–Wei ranking".[7]

Dominant eigenvectors do not react very well to the lack of strong connectivity. Depending on the dominant eigenvalue of the strongly connected components, the dominant eigenvector might or might not be nonzero on non-terminal components (a detailed characterization can be found in [8]).

[6] We remark that Tore Opsahl already in a March 2010 blog posting observed that in an undirected graph with several disconnected components the inverse of the harmonic mean of distances offers a better notion of centrality than closeness, as it weights less elements that belong to smaller components.

[7] It was rediscovered as a generic way of ranking graphs by Bonacich [11].

3.2.2 Seeley's index

The dominant eigenvector rationale can be slighly amended with the observation that the update rule we described can be thought of as if each node gives away its score to its successors: or even, that each node has a *reputation* and is giving its reputation to its successors so that they can build their own.

Once we take this viewpoint, it is clear that it is not very sensible to give away the same amount of reputation to everybody: it is more reasonable to *divide* equally reputation among our successors. From a linear-algebra viewpoint, this amounts to normalizing each row of the adjacency matrix using the ℓ_1 norm.

This approach was advocated by Seeley [46] for computing the popolarity among groups of children, given a graph representing whether each child liked or not another one. The matrix resulting from the ℓ_1-normalization process is actually stochastic, so the score can be interpreted as the probability distribution of the stationary state of a Markov chain. In particular, if the underlying graph is symmetric Seeley's index collapses to the degree (modulo normalization) because of the very well-known characterization of the stationary distribution of the natural random walk on a symmetric graph.

Also Seeley's index does not react very well to the lack of strong connectivity, but in a more predictable way: the only nodes with a nonzero rank are those belonging to terminal components.

3.2.3 Katz's index

Katz introduced his celebrated index [27] using a summation over all paths coming into a node, but weighting each path so that the summation would actually be finite. Due to the interplay between the powers of the adjacency matrix and the number of paths connecting two nodes, Katz's index can be expressed as

$$\boldsymbol{k} = \mathbf{1} \sum_{i=0}^{\infty} \beta^i A^i.$$

For the summation above to be finite, the *attenuation factor* β must be smaller than $1/\lambda$, where λ is the dominant eigenvalue of A.

Katz immediately noted that the index was actually expressible using linear algebra operations:

$$\boldsymbol{k} = \mathbf{1}(1 - \beta A)^{-1}.$$

It took some more time to realize that, due to Brauer's theorem on the displacement of eigenvalues [14], Katz's index is actually the left dominant eigenvector of a *perturbed matrix*

$$\beta\lambda A + (1 - \beta\lambda)\boldsymbol{e}^T\mathbf{1}, \tag{3}$$

where $\boldsymbol{e}$ is a right dominant eigenvector such that $\mathbf{1}\boldsymbol{e}^T = \lambda$ [50]. An easy generalization (actually suggested by Hubbell [25]) replaces the vector $\mathbf{1}$ with some preference vector $\boldsymbol{v}$, so that paths are also weighted differently depending on their starting node.[8]

If the underlying graph is strongly connected, the limit of Katz's index when $\beta \to 1/\lambda$ is exactly the dominant eigenvector [50]. This is also true under the much more general condition that the dominant eigenvalue of A is *semisimple* [37], but in that case the limit is a specific dominant eigenvector that depends on the preference vector $\boldsymbol{v}$.

[8] We must note that the original definition of Katz's index is $\mathbf{1}A\sum_{i=0}^{\infty}\beta^i A^i = \mathbf{1}/\beta\sum_{i=0}^{\infty}\beta^{i+1}A^{i+1} = (\mathbf{1}/\beta)\sum_{i=0}^{\infty}\beta^i A^i - \mathbf{1}/\beta$. This additional multiplication by A is somewhat common in the literature, even for PageRank; clearly, it alters the order induced by the ranking only when there is a nonuniform preference vector. Our discussion can be easily adapted for this version.

3.2.4 PageRank

PageRank [43] is one of the most discussed and quoted spectral indices in use today, mainly because of its alleged use in Google's ranking algorithm.[9]

By definition, PageRank is the left dominant eigenvector (i.e., the stationary distribution) $\boldsymbol{p}$ of the Markov chain

$$\alpha\bar{A} + (1-\alpha)\mathbf{1}^T\boldsymbol{v},$$

where again $\bar{A}$ is the ℓ_1-normalized adjacency matrix of the graph, and $\boldsymbol{v}$ is a *preference vector* (which must be a distribution). The reader will immediately notice the similarity with (3): indeed, we can work backwards and rewrite PageRank as

$$\boldsymbol{p} = \boldsymbol{v}\big(1-\alpha\bar{A}\big)^{-1},$$

leading to

$$\boldsymbol{p} = \boldsymbol{v}\sum_{i=0}^{\infty}\alpha^i\bar{A}^i,$$

which shows immediately that Katz's index and PageRank differ only by the ℓ_1 normalization applied to A, similarly to the difference between the dominant eigenvector and Seeley's index.

Analogously to what happens with Katz's index, the limit of PageRank when α goes to 1 is exactly the dominant eigenvector of $\bar{A}$, that is, Seeley's index [10, 24]. The statement is always true, because in stochastic matrices the dominant eigenvalue is always semisimple [8], but if the graph is not strongly connected the limit is a specific dominant eigenvector that depends on the preference vector $\boldsymbol{v}$ [10].

3.2.5 HITS

Kleinberg introduced his celebrated HITS algorithm [28] using the web metaphore of "mutual reinforcement": a page is authoritative if it is pointed by many good *hubs*—pages which contain good list of authoritative pages—, and a hub is good if it points to authoritative pages. This suggests an iterative process that computes at the same time an authoritativeness score $\boldsymbol{a}_i$ and a "hubbiness" score $\boldsymbol{h}_i$ starting with $\boldsymbol{a}_0 = \mathbf{1}$, and then applying the update rule

$$\boldsymbol{h}_{i+1} = \boldsymbol{a}_i A^T \qquad \boldsymbol{a}_{i+1} = \boldsymbol{h}_{i+1} A.$$

This process converges to the left dominant eigenvector of the matrix $A^T A$, which gives the final authoritativeness score, which is the score we label with "HITS" throughout the paper.[10]

Inverting the process, and considering the left dominant eigenvector of the matrix AA^T, gives the final hubbiness score. The two vectors are actually the left and right *singular vectors* associated with the largest *singular value* in the singular-value decomposition of A. Note also that hubbiness is the positive version of authoritativeness.

3.2.6 SALSA

Finally, we consider SALSA, a measure introduced by Lempel and Moran [31] always using the metaphore of mutual reinforcement between authoritativeness and hubbiness, but ℓ_1-normalizing the matrices A and A^T. We start with $\boldsymbol{a}_0 = \mathbf{1}$ and proceed with

$$\boldsymbol{h}_{i+1} = \boldsymbol{a}_i\overline{A^T} \qquad \boldsymbol{a}_{i+1} = \boldsymbol{h}_{i+1}\bar{A}.$$

[9] The reader should be aware, however, that the literature about the actual effectiveness of PageRank in information retrieval is rather scarce, and comprises mainly negative results such as [40] and [19].

[10] As discussed in [20], the dominant eigenvector may not be unique; equivalently, the limit of the recursive definition given above may depend on the way the authority and hub scores are initialized. Here we consider the result of the iterative process starting with $\boldsymbol{a}_0 = \mathbf{1}$.

We remark that this normalization process is analogous to the one that moves us from the dominant eigenvector to Seeley's index, or from Katz's index to PageRank.

Similarly to what happens with Seeley's index on symmetric graphs, SALSA does not need such an iterative process to be computed.[11] First, one computes the connected components of the symmetric graph induced by the matrix $A^T A$; in this graph, x and y are adjacent if x and y have some common predecessor in the original graph. Then, the score of a node is the ratio between its indegree and the sum of the indegrees of nodes in the same component, multiplied by the ratio between the component size and n. Thus, contrarily to HITS, a single linear scan of the graph is sufficient to compute SALSA, albeit the computation of the intersection graph requires time proportional to $\sum_x d^+(x)^2$.

3.3 Path-based measures

Path-based measures exploit not only the existence of a shortest paths, but actually take into examination all shortest paths (or all paths) coming into a node. We remark that indegree can be considered a path-based measure, as it is the equivalent to the number of incoming paths of length one.

3.3.1 Betweeness

Betweenness centrality was introduced by Anthonisse [2] for edges, and then rephrased by Freeman for nodes [22]. The idea is to measure the probability that a random shortest path passes through a given node: if σ_{yz} is the number of shortest paths going from y to z, and $\sigma_{yz}(x)$ is the number of such paths that pass through x, we define the *betweeness* of x as

$$\sum_{y,z \neq x, \sigma_{yz} \neq 0} \frac{\sigma_{yz}(x)}{\sigma_{yz}}.$$

The intuition behind betweeness is that if a large fraction of shortest paths passes through x, than x is an important point of passage for the network. Indeed, removing nodes in betweeneess order causes a very quick disruption of the network [9].

3.3.2 Spectral measures as path-based measures

It is a general observation that all spectral measures can actually be interpreted as path-based measures, as they depend on taking the limit of some summations of powers of A, or on the limit of powers of A, and in both cases we can express these algebraic operations in terms of suitable paths.

For instance, the left dominant eigenvector of a nonnegative matrix can be computed with the power method by taking the limit of $\mathbf{1}A^k/\|\mathbf{1}A^k\|$ for $k \to \infty$. Since, however, $\mathbf{1}A^k$ is a vector associating with each node the number of paths of length k coming into the node, we can see that dominant eigenvector expresses the relative growth of the number of paths coming into each node as their length increases.

Analogously, Seeley's index can be computed (modulo a normalization factor) by taking the limit of $\mathbf{1}\bar{A}^k$ (in this case, the ℓ_1 norm cannot grow, so we do not need to renormalize at each iteration). The vector $\mathbf{1}\bar{A}^k$ has the following combinatorial interpretation: it assigns to each x the sums of the *weights* of the paths coming into x, where the weight of a path $x_0, x_1, \ldots, x_t$ is

$$\prod_{i=0}^{t-1} \frac{1}{d^+(x_i)}. \tag{4}$$

[11] This property, which appears to be little known, is proved in Proposition 2 of the original paper [31].

When we switch to the attenuated versions of the previous indices (that is, Katz's index and PageRank), we switch from limits to infinite summations and at the same time modify the weight of a path of length t with β^t or α^t. Actually, the Katz index of x was originally defined as the summation over all t of the number of paths of length t coming into x multiplied by β^t, and PageRank is the summation over all paths coming into x of the weight (4) multiplied by α^t.

The reader can easily work out similar definitions for HITS and SALSA, which depend on a suitable definition of alternate "back-and-forth path" (see, e.g., [13])

4 Axioms for Centrality

The comparative evaluation of centrality measures is a challenging, difficult, arduous task, for many different reasons. The datasets that are classically used in social sciences are very small (typically, some tens of nodes) and it is hard to draw conclusions out of them. Nonetheless, some attempts were put forward, like [48]; sometimes, the attitude was actually to provide evidence that different measures highlight different kinds of centralities and are therefore equally incomparably interesting [23]. Whether the latter attitude is the only sensible conclusion or not is debatable. While it is clear that the notion of centrality, in its vagueness, can be interpreted differently giving rise to many good but incompatible measures, we will provide evidence that some measures tend to reward nodes that are in no way central.

If results obtained on small corpora may be misleading, a comparison on larger corpora is much more difficult to deal with, due to the lack of ground truth and to the unavailability of implementations of efficient algorithms to compute the measures under consideration (at least in the cases where efficient, possibly approximate, algorithms do exist). Among the few attempts that try a comparison on large networks we cite [49] and [41], that nevertheless focus only on web graphs and on a very limited number of centrality indices.

In this paper, we propose to understand (part of) the behaviour of a centrality measure using a set of axioms. While, of course, it is not sensible to prescribe a set of axioms that *define* what centrality should be (in the vein of Shannon's definiton of entropy [47] or Altman and Tennenholtz axiomatic definition of Seeley's index [1][12]), as different indices serve different purposes, it is reasonable to set up some *necessary* axioms that an index should satisfy to behave predictably and follow our intuition.

The other interesting aspect of defining axioms is that, even if one does not believe they are really so discriminative or necessary, they provide a very specific, formal, provable piece of information about a centrality measure that is much more precise than folklore intuitions like "this centrality is really correlated to indegree" or "this centrality is really fooled by cliques". We believe that a theory of centrality should exactly provide this kind of compact, meaningful, reusable information (in the sense that it can be used to prove other properties). This is indeed what happens, for example, in topology, where the information that a space is T_0, rather than T_1, is a compact way to provide a lot of information about the structure of the space.

Definining such axioms is a delicate matter. First of all, the semantics of the axioms must be very clear. Second, the axioms must be evaluable in an exact way on the most common centrality measures. Third, they should be formulated avoiding the trap of small, finite (counter)examples, on which many centrality measures collapse (e.g., using an asymptotic definition). We assume from the beginning that the centrality measures under examination are invariant by isomorphism, that is, that they depend just on the structure of the graph, and not on particular labelling chosen for each node.

To meet these constraints, we propose to study the reaction of centrality measures to *change of size*, to *(local) change of density* and their *monotonicity with respect to arc additions*. We expect that nodes belonging to larger groups, when all other parameters are fixed, should be more important,

[12]The authors claim to formalize PageRank [44], but they do not consider the damping factor (equivalently, they are setting $\alpha = 1$), so they are actually formalizing Seeley's venerable index [46].

and that nodes with a denser neighbourhood (i.e., having more friends), when all other parameter are fixed, should also be more important. We also expect that adding an arc should increase the importance of the target.

How can we actually determine if this happens in an exact way, and possibly in an asymptotic setting? To do so, we need to do something entirely new—evaluating *exacly* (i.e., in algebraic closed form) all measures of interest on all nodes of some representative classes of networks.

4.1 The size axiom

An obvious approach to reduce to a minimum the amount of computation is using strongly connected *vertex-transitive*[13] graphs as basic building blocks: these graphs have as much symmetry as possible, which entails a simplification of the computations. Finally, since we want to compare density, the obvious choice is to pick the *densest* strongly connected vertex-transitive graph, the clique, and the *sparsest* strongly connected, the directed cycle. Choosing two graphs at the extreme of the density spectrum guarantees that best possible highlight of the reaction of centrality measures to densities. Moreover, k-cliques and directed p-cycles obviously exist for every k and p (this might not happen for more complicated structures, e.g., a cubic graph).

Let us consider a graph made by a k-clique and a p-cycle (see the figure in Table 1).[14] Because of invariance by isomorphism, all nodes of the clique has the same score, and all nodes of the cycle have the same score. But which nodes are more important? Probably everybody would answer that if $p = k$ the elements on the clique are more important, and indeed this axiom is so trivial that is satisfied by almost any measure we are aware of. But we are interested in assessing the sensitivity to *size*, and thus we state our first axiom:

Definition 1 (Size axiom) *Consider the graph $S_{k,p}$ made by a k-clique and a directed p-cycle. A centrality measure satisfies the* size axiom *if for every k there is a P_k such that for all $p \geq P_k$ in $S_{k,p}$ the centrality of a node of the p-cycle is strictly larger than the centrality of a node of the k-clique, and if for every p there is a K_p such that for all $k \geq K_p$ in $S_{k,p}$ the centrality of a node of the k-clique is strictly larger than the centrality of a node of the p-cycle.*

Intuitively, when $p = k$ we do expect nodes of the cycle to be less important than nodes of the clique. (Note that because of vertex transitivity and invariance by isomorphism we can speak of the "centrality of a node of the p-cycle", without specifying which node.) The rationale behind the case $k \to \infty$ is rather obvious: the denser community is also getting larger, and thus its members are expected to become even more important.

On the other hand, if the cycle becomes very large (more precisely, when its size goes to infinity), its nodes are still part of a very large (albeit badly connected) community, and we expect them to achieve at some size greater importance than the node of a fixed-size community, no matter how dense it can be.

Since one might devise some centrality measures that satisfy the size axiom for p and not for k, which we would not certaintly want to pass our screening, stating both properties in Definition 1 gives us a finer granularity and avoids pathological cases.

4.2 The density axiom

Designing an axiom for density is a more delicate issue, since we must be able to define an increase of density "with all other parameters fixed", including size. Let us start ideally from a graph made

[13] A graph is vertex-transitive if for every nodes x and y there is an automorphism exchanging x and y.

[14] The graph is of course disconnected. It is a common theme of this work that centrality measures should work also on graphs that are not strongly connected, for the very simple reason that we meet this kind of graphs in the real world, the web being an obvious example.

by a directed k-cycle and a directed p-cycle, and connect a vertex x of the k-cycle with a vertex y of the p-cycle through a bidirectional arc, the *bridge*. If $k = p$, the vertices x and y are symmetric, and thus must have necessarily the same ranking. Now, we increase the density of the k-cycle as much as possible, turning it into a k-clique (see the figure in Table 2). Note that this change of density is local to x, as the degree of y has not changed. We are thus *strictly increasing the local density around x, leaving all other parameters fixed*, and in these circumstances we expect that the ranking x increases.

Definition 2 (Density axiom) *Consider the graph $D_{k,p}$ made by a k-clique and a p-cycle ($p, k \geq 3$) connected by a bidirectional bridge $x \leftrightarrow y$, where x is a node of the clique and y is a vertex of the cycle. A centrality measure satisfies the* density axiom *if for $k = p$ the centrality of x is strictly larger than the centrality of y.*

Note that our axiom does not specify any constraint when $k \neq p$. While studying the behaviour of the graph $D_{k,p}$ of the previous definition when $k \neq p$ shades some lights of the inner behaviour of centrality measures, it is essential, in an axiom asserting the sensitivity to density, that size is not involved.

In our proofs for the density axiom, we actually let k and p be independent parameters (even if the axiom requires $k = p$) because in this way we can compute the *watershed*, that is, the value of k (expressed as a function of p) at which the axiom becomes true (if any). The watershed can give some insight as to how badly a measure can miss to satisfy the density axiom.

4.3 The monotonicity axiom

Finally, we propose a seemingly trivial axiom that specifies strictly monotonic behaviour upon the addition of an arc:

Definition 3 (Monotonicity axiom) *Consider an arbitrary graph G and a pair of nodes x, y such that $x \not\to y$. A centrality measure satisfies the* monotonicity axiom *if when we add $x \to y$ the centrality of y increases.*

Actually, in some sense this axiom *is* trivial: it is satisfied by essentially all centrality measures we consider *on strongly connected graphs*. Thus, it is an excellent test to verify that a measure is able to handle correctly partially disconnected graphs.

We remark that the reader might be tempted to define a *weak* monotonicity axiom which just require the rank of y to be nondecreasing. However, the constant ranking associating one to every node of every network would satisfy such an axiom, which makes it not very interesting for our goals.

5 Proofs and Counterexamples

We have finally reached the core of this paper: given that we are considering eleven centralities and three axioms, we have to verify 33 statements. For the size and density axioms, we compute in closed form the values of all measures, from which we can derived the desired results, whereas for the monotonicity axiom we provide directly proofs or counterexamples.

We remark that in all our tables we use the proportionality symbol $\propto$ to mark values that have been rescaled by a common factor to make them more readable.

5.1 Size

Table 1 provides scores for the graph $S_{p,k}$, from which we can check whether the size axiom is satisfied. The scores are all immediately computable from the basic definitions, because as we noticed $S_{k,p}$ is highly symmetrical and so there are actually only two scores—the score of a node of the clique

and the score of a node of the cycle. Note that in the case of some spectral centrality measure there are actually several possible solutions, in which case we use the one returned by the power method starting from the uniform vector.

5.2 Density

Table 2 provides scores for the graph $D_{p,k}$. Being the graph strongly connected, there is no uniqueness issue. While the computation of geometric and path-based centrality measures, being just a matter of finite summations, is tedious but rather straightforward, spectral indices require some more care. In the case of $D_{k,p}$, we have to write down parametric equations expressing the matrix computation that defines the centrality, and solve them. As noted before, we prefer to perform the computation with two independent parameters k and p (even if the axiom requires $k = p$) because in this way we can compute the watershed.

In all cases, we can always use the bounds imposed by symmetry to write down just a small number of variables: c for the centrality of an element of the clique, ℓ for the clique bridge ("left"), r for the cycle bridge ("right"), and some function $t(d)$ of the distance from the cycle bridge for the nodes of the cycle (with $0 < d < p$), with the condition $t(0) = r$.

5.2.1 The left dominant eigenvector

In this case, the equations are given by the standard eigenvalue problem of the adjacency matrix:

$$\begin{aligned}\lambda\ell &= r + (k-1)c\\ \lambda c &= \ell + (k-2)c\\ \lambda r &= \ell + \frac{r}{\lambda^{p-1}},\end{aligned}$$

subject to the condition that we choose the λ with maximum absolute value. Note that in the case of the last equation we "unrolled" the equations about the elements of the cycle, $\lambda t(d+1) = t(d)$. Solving the system and choosing $c = 1/(\lambda - k + 1)$ gives the solutions found in Table 2.

Since for nonnegative matrices the dominant eigenvalue is monotone in the matrix entries, $\lambda \geq k-1$, because the k-clique has dominant eigenvalue equal to $k-1$. On the other hand, $\lambda \leq k$ by row-sum bounds, and the eigenvalue equations have no solution for $\lambda = k-1$, so we conclude that $k-1 < \lambda \leq k$.

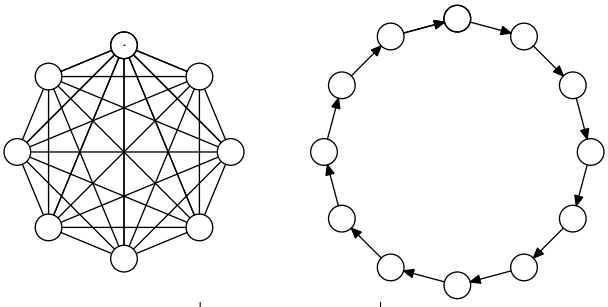
5.2.2 Katz's index

In this case, the equations can be obtained by the standard technique of "taking one summand out", that is, writing

$$\boldsymbol{k} = \mathbf{1}\sum_{i=0}^{\infty}\beta^i A^i = \mathbf{1} + \mathbf{1}\sum_{i=1}^{\infty}\beta^i A^i = \mathbf{1} + \left(\mathbf{1}\sum_{i=0}^{\infty}\beta^i A^i\right)\beta A = \mathbf{1} + \boldsymbol{k}\beta A.$$

The equations are then

$$\begin{aligned}\ell &= 1 + \beta r + \beta(k-1)c\\ c &= 1 + \beta\ell + \beta(k-2)c\\ r &= 1 + \beta\ell + \beta\left(\frac{1-\beta^{p-1}}{1-\beta} + \beta^{p-1}r\right),\end{aligned}$$



Centrality	k-clique	p-cycle
Degree	$k-1$	1
Harmonic	$k-1$	H_{p-1}
Closeness	$\frac{1}{k-1}$	$\frac{2}{p(p-1)}$
Lin	$\frac{k^2}{k-1}$	$\frac{2p}{p-1}$
Betweenness	0	$\frac{(p-1)(p-2)}{2}$
Dominant $\propto$	1	0
Seeley $\propto$	1	1
Katz	$\frac{1}{1-(k-1)\beta}$	$\frac{1}{1-\beta}$
PageRank $\propto$	1	1
HITS $\propto$	1	0
SALSA $\propto$	1	1

Table 1: Centrality scores for the graph $S_{k,p}$. H_i denotes the i-th harmonic number. The parameter β is Katz's attenuation factor.

where again we "unrolled" the equations about the elements of the cycle, as we would just have $t(d+1) = 1 + \beta t(d)$, so

$$t(d) = \frac{1-\beta^d}{1-\beta} + \beta^d r.$$

The explicit values of the solutions are quite ugly, so we present them in Table 2 as a function of the centrality of the clique bridge ℓ.

5.2.3 PageRank

To simplify the computation, we use $\mathbf{1}$, rather than $\mathbf{1}/(k+p)$, as preference vector (the result obtained is obviously the same up to proportionality). We use the same technique employed in the computation of Katz's index, leading to

$$\begin{aligned}
\ell &= 1 - \alpha + \frac{1}{2}\alpha r + \alpha c \\
c &= 1 - \alpha + \frac{\alpha}{k}\ell + \alpha\frac{k-2}{k-1}c \\
r &= 1 - \alpha + \frac{\alpha}{k}\ell + \alpha\left(1 - \alpha^{p-1} + \frac{1}{2}\alpha^{p-1} r\right),
\end{aligned}$$

noting once again that unrolling the equation of the cycle $t(1) = 1 - \alpha + \alpha r/2$ and $t(d+1) = 1 - \alpha + \alpha t(d)$ for $d > 1$ we get

$$t(d) = 1 - \alpha^d + \frac{1}{2}\alpha^d r.$$

The explicit values for PageRank are even uglier than those of Katz's index, so again we present them in Table 2 as a function of the centrality of the clique bridge ℓ.

5.2.4 Seeley's index

This is a freebie, as we can just compute PageRank's limit when $\alpha \to 1$.

5.2.5 HITS

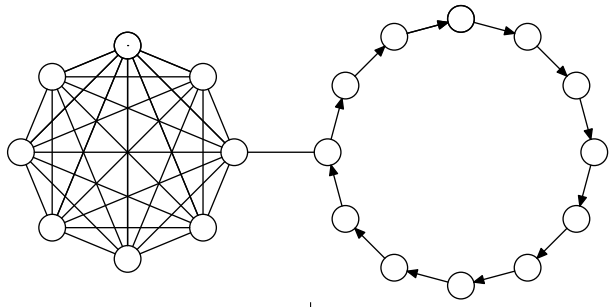
In this case we write down an eigenvalue problem for $A^T A$. We have

$$\begin{aligned}
\mu c &= (k-1)c + (k-2)^2 c + (k-2)\ell + r \\
\mu \ell &= k\ell + (k-1)(k-2)c + t \\
\mu r &= 2r + (k-1)c \\
\mu t &= t + \ell.
\end{aligned}$$

By normalizing the result so that $c = \mu^2 - \mu(k+1) + k - 1$, and noting that the dominant eigenvalue of $A^T A$ is the square of the dominant eigenvalue of A, we obtain the complex but somewhat readable values shown in Table 2. Note that p has no role in the solution, because $A^T A$ can be decomposed into two independent blocks, one of which is an identity matrix corresponding to all elements of the cycle except for the first two.

5.2.6 SALSA

It is easy to check that the components of the intersection graph of precedessors are given by the clique together with the cycle bridge and its successor, and then by one component for each node of the cycle. The computation of the scores is then trivial using the non-iterative rules.



Centrality	Clique	Clique bridge	Cycle bridge	Cycle ($d > 0$ from the bridge)	Watershed
Degree	$k-1$	k	2	1	—
Harmonic	$k-2+H_{p+1}$	$k-1+H_p$	$1+\frac{k-1}{2}+H_{p-1}$	$\frac{1}{d+1}+\frac{k-1}{d+2}+H_{p-1}$	—
Closeness	$\frac{1}{k-1+2p+p(p-1)/2}$	$\frac{1}{k-1+p+p(p-1)/2}$	$\frac{1}{2k-1+p(p-1)/2}$	$\frac{1}{k(d+2)-1+p(p-1)/2}$	$k \leq p$
Betweenness	0	$2p(k-1)$	$2k(p-1)+\frac{(p-1)(p-2)}{2}$	$2k(p-2)+\frac{(p-1)(p-2)}{2}$	$k \leq \frac{p^2+p+2}{4}$
Dominant	$\frac{1}{\lambda-k+1}$	$1+\frac{1}{\lambda-k+1}$	$1+\lambda$	$\frac{1+\lambda}{\lambda^d}$	—
Seeley $\propto$	$k-1$	k	2	1	—
Katz $\propto$	$\frac{1+\beta\ell}{1-\beta(k-2)}$	ℓ	$\frac{1}{1-\beta}+\frac{\beta}{1-\beta^p}\ell$	$\frac{1}{1-\beta}+\frac{\beta^{d+1}}{1-\beta^p}\ell$	—
PageRank $\propto$	$\frac{(k-1)(k-\alpha k+\alpha\ell)}{k(k-1-\alpha(k-2))}$	ℓ	$2+2\frac{\alpha\ell-k}{k(2-\alpha^p)}$	$1+\alpha^d\frac{\alpha\ell-k}{k(2-\alpha^p)}$	—
HITS $\propto$	$\lambda^4-\lambda^2(k+1)+k-1$	$(k-1)(k-2)(\lambda^2-1)$	$\lambda^6-(k^2-2k+4)\lambda^4$ $+(3k^2-7k+6)\lambda^2-(k-1)^2$	$[d=1](k-1)(k-2)$	—
SALSA $\propto$	$(k-1)(k+2)$	$k(k+2)$	$2(k+2)$	$k+2+[d\neq 1](k^2-2k+2)$	—

Table 2: Centrality scores for the graph $D_{k,p}$. The parameter β is Katz's attenuation factor, whereas $\alpha \in [0\,..\,1]$ is PageRank's damping factor. The value $k-1 < \lambda \leq k$ is the dominant eigenvalue of the adjacency matrix A. Lin's centrality is omitted because it is proportional to closeness (the graph being strongly connected).

Armed with our closed-form description of the scores, we have now to prove whether the density axiom actually holds, that is, whether $\ell > r$ when $k = p$. In Table 2 we report the *watershed*, that is, the point at which the axiom becomes true. When there is no watershed, the axiom is true for every $k, p \geq 3$. Note that the determination of the watershed is trivial in almost all cases. We will now discuss the remaining cases.

Theorem 1 *HITS satisfies the density axiom.*

Proof. As we have seen, we can normalize the solution to the HITS equation so that

$$\begin{aligned}\ell &= (k-1)(k-2)(\mu-1)\\ r &= \mu^3 - (k^2-2k+4)\mu^2 + (3k^2-7k+6)\mu - (k-1)^2\end{aligned}$$

Moreover, the characteristic polynomial can be computed explicitly from the set of equations and some simple observations on the eigenvectors for the eigenvalue 1:

$$p(\mu) = \big(\mu^4 - (k^2-2k+6)\mu^3 + (5k^2-12k+15)\mu^2 - (6k^2-16k+14)\mu + k^2 - 2k + 1\big)(1-\mu)^{k+p-4}.$$

The largest eigenvalue μ_0 satisfies the inequation $(k-1)^2 \leq \mu_0 \leq k^2 - 2k + 5/4$ for every $k \geq 9$, as shown below (the statement of the theorem can be verified in the remaining cases by explicit computation, as it does not depend on p). Using the stated upper and lower bounds on μ_0, we can say that

$$\begin{aligned}\ell - r &= (k-1)(k-2)(\mu-1) - (\mu^3 - (k^2-2k+4)\mu^2 + (3k^2-7k+6)\mu - (k-1)^2)\\ &= -\mu^3 + (k^2-2k+4)\mu^2 - (2k^2-4k+4)\mu + k - 1\\ &\geq -\left(k^2-2k+\frac{5}{4}\right)^3 + (k^2-2k+4)(k-1)^4 - (2k^2-4k+4)\left(k^2-2k+\frac{5}{4}\right) + k - 1\\ &= \frac{1}{4}k^4 - k^3 - \frac{19}{16}k^2 + \frac{43}{8}k - \frac{253}{64},\end{aligned}$$

which is positive for $k > 4$.

We are left to prove the bounds on μ_0. The lower bound can be easily obtained by monotonicity of the dominant eigenvalue in the matrix entries, because the dominant eigenvalue of a k-clique is $k-1$. For the upper bound, first we observe that μ_0 can be computed explicitly (as it is the solution of a quartic equation) and using its expression in closed form it is possible to show that $\lim_{k\to\infty} \mu_0 = (k-1)^2$. This guarantees that the bound $\mu_0 \leq k^2 - 2k + 5/4$ is true ultimately. To obtain an explicit value of k after which the bound holds true, observe that $k^2 - 2k + 5/4 = \mu_0$ implies $q(k) = p(\mu_0) = 0$, where $q(k) = p(k^2 - 2k + 5/4)$. Computing the Sturm sequence associated to $q(k)$ one can prove that $q(k)$ has no zeroes for $k \geq 9$, hence our lower bound on k. ∎

Theorem 2 *Katz's index satisfies the density axiom.*

Proof. Recall that the equations for Katz's index are

$$\begin{aligned}\ell &= 1 + \beta r + \beta(k-1)c\\ c &= 1 + \beta\ell + \beta(k-2)c\\ r &= 1 + \beta\ell + \beta\left(\frac{1-\beta^{p-1}}{1-\beta} + \beta^{p-1}r\right).\end{aligned}$$

First, we remark that as $\beta \to 1/\lambda$ Katz's index tends to the dominant eigenvector, so ultimately $\ell > r$. Thus, by continuity, we just need to show that $\ell = r$ never happens in the range of our parameters. If we solve the equations above for c, ℓ and r and impose $\ell = r$, we obtain

$$p = \frac{\ln \dfrac{\beta^2 + k - 2}{k - 1}}{\ln \beta}.$$

Now observe that

$$\beta \leq \frac{\beta^2 + k - 2}{k - 1}$$

is always true for $\beta \leq 1$ and $k \geq 3$. This implies that under the same conditions $p \leq 1$, which concludes the proof. ∎

Theorem 3 *PageRank with constant preference vector satisfies the density axiom.*

Proof. The proof is similar to that of Theorem 2. Recall that the equations for PageRank are

$$\begin{aligned}
\ell &= 1 - \alpha + \frac{1}{2}\alpha r + \alpha c \\
c &= 1 - \alpha + \frac{\alpha}{k}\ell + \frac{\alpha(k-2)}{k-1}c \\
r &= 1 - \alpha + \frac{\alpha}{k}\ell + \alpha\left(1 - \alpha^{p-1} + \frac{1}{2}\alpha^{p-1} r\right).
\end{aligned}$$

First, we remark that as $\alpha \to 1$ PageRank tends to Seeley's index, so ultimately $\ell > r$. By continuity, we thus just need to show that $\ell = r$ never happens in our range of parameters. If we solve the equations above for c, ℓ and r and impose $\ell = r$, we obtain

$$p = 1 + \frac{\ln\left(-\dfrac{2\alpha^2 - (k^2 - 4k + 6)\alpha + k^2 - 3k + 2}{(k^2 - 3k + 2)\alpha^2 - (2k^2 - 3k)\alpha + k^2 - k}\right)}{\ln \alpha}.$$

Now observe that $2\alpha^2 - (k^2 - 4k + 6)\alpha + k^2 - 3k + 2 \geq 0$ for $k \geq 3$. Thus, a solution for p exists only when the denominator is negative. However, in that region

$$-\frac{2\alpha^2 - (k^2 - 4k + 6)\alpha + k^2 - 3k + 2}{(k^2 - 3k + 2)\alpha^2 - (2k^2 - 3k)\alpha + k^2 - k} \geq 1.$$

This implies that under the same conditions $p \leq 1$, which concludes the proof. ∎

5.3 Monotonicity

For the monotonicity axiom, we discuss briefly the nontrivial cases.

5.3.1 Harmonic

If you add an arc $x \to y$ the harmonic centrality of y can only increase, because this addition can only reduce the distances (possibly even turning some of them from infinite to finite), so it will increase their reciprocals (strictly increasing the one from x).

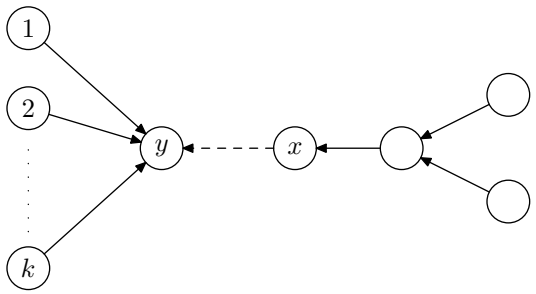


Figure 1: A counterexample showing that Lin's index fails to satisfy the monotonicity axiom.

5.3.2 Closeness

If you consider a one-arc graph $z \to y$ and add an arc $x \to y$, the closeness of y decreases from 1 to $1/2$.

5.3.3 Lin

Consider the graph in Figure 1: the Lin centrality of y is $(k+1)^2/k$. After adding an arc $x \to y$, the centrality becomes $(k+5)^2/(k+9)$, which is smaller than the previous value when $k > 3$.

5.3.4 Betweenness

If you consider a graph made of two isolated nodes x and y, the addition of the arc $x \to y$ leaves the betweenness of x and y unchanged.

5.3.5 Katz

The score of y after adding $x \to y$ can only increase, because the set of paths coming into y now contains new elements[15].

5.3.6 Dominant eigenvector, Seeley's index, HITS

If you consider a clique and two isolated nodes x, y, the rank given by the dominant eigenvector, Seeley's index and HITS to x and y is zero, and it remains unchanged when the arc $x \to y$ is added.

5.3.7 SALSA

Consider the graph in Figure 2: the indegree of y is 1, and its component in the intersection graph of predecessors is trivial, so its SALSA centrality is $(1/1) \cdot (1/6) = 1/6$. After adding an arc $x \to y$, the indegree of y becomes 2, but now its component is $\{ y, z \}$; so the sum of indegrees within the component is $2 + 3 = 5$, hence the centrality of y becomes $(2/5) \cdot (2/6) = 2/15 < 1/6$.

5.3.8 PageRank

The case of PageRank turns out to be definitely nontrivial:

Theorem 4 *PageRank satisfies the monotonicity axiom if* $\alpha \in (0\,..\,1)$.

[15] It should be noted, however, that this is true only for the values of the parameter β that still make sense after the addition.

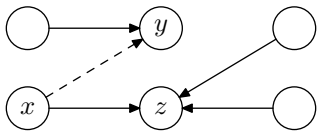


Figure 2: A counterexample showing that SALSA fails to satisfy the monotonicity axiom.

Proof. For this proof, we define PageRank as $\boldsymbol{v}\big(1-\alpha\bar{A}\big)^{-1}$ (i.e., without the normalizing factor $1-\alpha$), so to simplify our calculations. By linearity, the result for the standard definition follows immediately.

Consider two nodes x and y of a graph G such that there is no arc from x to y, and let d be the outdegree of x. Given the normalized matrix $\bar{A}$ of G, and the normalized matrix $\bar{A}'$ of the graph G' obtained by adding to G the arc $x \to y$, we have

$$\bar{A} - \bar{A}' = \boldsymbol{\chi}_x \boldsymbol{\delta},$$

where $\boldsymbol{\chi}_x$ is the characteristic vector of x, and $\boldsymbol{\delta}$ is the difference between the rows corresponding to x in $\bar{A}$ and $\bar{A}'$, which contains $1/d(d+1)$ in the positions corresponding to the successors of x in G, and $-1/(d+1)$ in the position corresponding to y (note that if $d = 0$, we have just the latter entry).

We now use the Sherman–Morrison formula to write down the inverse of $1-\alpha\bar{A}'$ as a function of $1-\alpha\bar{A}$. More precisely,

$$\begin{aligned}\big(1-\alpha\bar{A}'\big)^{-1} = \Big(1-\alpha\big(\bar{A}-\boldsymbol{\chi}_x^T\boldsymbol{\delta}\big)\Big)^{-1} &= \big(1-\alpha\bar{A}+\alpha\boldsymbol{\chi}_x^T\boldsymbol{\delta}\big)^{-1} \\ &= \big(1-\alpha\bar{A}\big)^{-1} - \frac{\big(1-\alpha\bar{A}\big)^{-1}\alpha\boldsymbol{\chi}_x^T\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}}{1+\alpha\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}\boldsymbol{\chi}_x^T}.\end{aligned}$$

We now multiply by the preference vector $\boldsymbol{v}$, obtaining the explicit PageRank correction:

$$\begin{aligned}\boldsymbol{v}\big(1-\alpha\bar{A}'\big)^{-1} = \boldsymbol{v}\big(1-\alpha\bar{A}\big)^{-1} - \boldsymbol{v}&\frac{\big(1-\alpha\bar{A}\big)^{-1}\alpha\boldsymbol{\chi}_x^T\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}}{1+\alpha\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}\boldsymbol{\chi}_x^T} \\ &= \boldsymbol{r} - \frac{\alpha\boldsymbol{r}\boldsymbol{\chi}_x^T\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}}{1+\alpha\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}\boldsymbol{\chi}_x^T}\boldsymbol{r} - \frac{\alpha r_x\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}}{1+\alpha\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1}\boldsymbol{\chi}_x^T}.\end{aligned}$$

We now note that $\big(1-\alpha\bar{A}\big)^{-1}\boldsymbol{\chi}_x^T$ is the vector of positive contributions to the PageRank of x, modulo the normalization factor $1-\alpha$. As such, it is made of positive values adding up to at most $1/(1-\alpha)$. When the vector is multiplied by $\boldsymbol{\delta}$, in the worst case ($d = 0$) we obtain $1/(1-\alpha)$, so given the conditions on α it is easy to see that the denominator is positive. This implies that we can gather all constants in a single positive constant c and just write

$$\boldsymbol{v}\big(1-\alpha\bar{A}'\big)^{-1} = \big(\boldsymbol{v}-c\boldsymbol{\delta}\big)\big(1-\alpha\bar{A}\big)^{-1}.$$

The above equation rewrites the rank-one correction due to the addition of the arc $x \to y$ as a formal correction of the preference vector. We are interested in the difference

$$\big(\boldsymbol{v}-c\boldsymbol{\delta}\big)\big(1-\alpha\bar{A}\big)^{-1} - \boldsymbol{v}\big(1-\alpha\bar{A}\big)^{-1} = -c\boldsymbol{\delta}\big(1-\alpha\bar{A}\big)^{-1},$$

as we can conclude our proof by just showing that its y-th coordinate is strictly positive.

We now note that being $\left(1-\alpha\bar{A}\right)$ strictly diagonally dominant, the (nonnegative) inverse $B = \left(1-\alpha\bar{A}\right)^{-1}$ has the property that the entries b_{ii} on the diagonal are strictly larger than off-diagonal entries b_{ki} on the same column [36, Remark 3.3], and in particular they are nonzero. Thus, if $d = 0$

$$\left[-c\boldsymbol{\delta}\left(1-\alpha\bar{A}\right)^{-1}\right]_y = \frac{c}{d+1}b_{yy} > 0,$$

and if $d \neq 0$

$$\left[-c\boldsymbol{\delta}\left(1-\alpha\bar{A}\right)^{-1}\right]_y = \frac{c}{d+1}b_{yy} - \sum_{x\to z}\frac{c}{d(d+1)}b_{zy} > \frac{c}{d+1}b_{yy} - \sum_{x\to z}\frac{c}{d(d+1)}b_{yy} = 0. \blacksquare$$

6 Roundup

All our proofs are summarized in Table 3, where we distilled our results into simple yes/no answers to the question: does a given centrality measure satisfy the axioms?

It was surprising for us to discover that *only harmonic centrality satisfies all axioms*.[16] All spectral centrality measures are sensitive to density. Row-normalized spectral centrality measures (Seeley's index, PageRank and SALSA) are insensitive to size, whereas the remaining ones are only sensitive to the increase of k (or p in the case of betweenness). All non-attenuated spectral measures are also non-monotone. Both Lin's and closeness centrality fail density tests[17]. Closeness has indeed the worst possible behaviour, failing to satisfy all our axioms. While this result might seem counterintuitive, it is actually a consequence of the known tendence of very far nodes to dominate the score, hiding the contribution of closer nodes, whose presence is more correlated to local density.

All centralities satisfying the density axiom have no watershed: the axiom is satisfied for all $p, k \geq 3$. The watershed for closeness (and Lin's index) is $k \leq p$, meaning that they just miss it, whereas the watershed for betweeness is a quite pathological condition ($k \leq (p^2+p+2)/4$): you need a clique whose size is *quadratic* in the size of the cycle before the node of the clique on the bridge becomes more important than the one on the cycle (compare this with closeness, where $k = p+1$ is sufficient).

We remark that our results on geometric indices do not change if we replace the directed cycle with a symmetric (i.e., undirected) cycle. It is possible that the same is true also of spectral rankings, but the geometry of the paths of the undirected cycle makes it extremely difficult to carry on the analogous computations in that case.

7 Sanity check via information retrieval

Information retrieval has developed in the last fifty years a large body of research about extracting knowledge from data. In this section we want to leverage the work done in that field to check that our axioms actually describe interesting features centrality measures. We are in this sense following the same line of thought as in [40]: in that paper, the authors tried to establish in a methodologically sound way which of degree, HITS and PageRank works better as a feature in web retrieval. Here we ask the same question, but we include for the first time also geometric indices, which had never been considered before in the literature about information retrieval, most likely because it was not possible to compute them efficiently on large networks.

[16] It is interesting to note that it is actually the only centrality satisfying the size axiom—in fact, you need a cycle of $\approx e^k$ nodes to beat a k-clique.

[17] We note that since $D_{k,p}$ is strongly connected, closeness and Lin's centrality differ just by a multiplicative constant.

Centrality	Size	Density	Monotonicity
Degree	only k	yes	yes
Harmonic	yes	yes	yes
Closeness	no	no	no
Lin	only k	no	no
Betweenness	only p	no	no
Dominant	only k	yes	no
Seeley	no	yes	no
Katz	only k	yes	yes
PageRank	no	yes	yes
HITS	only k	yes	no
SALSA	no	yes	no

Table 3: For each centrality and each axiom, we report whether it is satisfied.

The community working on information retrieval developed a number of standard datasets with associated queries and ground truth about which documents are relevant for every query; those collections are typically used to compare the (de)merits of new retrieval methods; since many of those collections are made of hyperlinked documents, it is possible to use them to assess centrality measures, too.

In this paper we consider the somewhat classical TREC GOV2 collection (aboud 25 million web documents) and the 149 associated queries. For each query (*topic*, in TREC parlance), we have solved the corresponding Boolean conjunction of the terms, obtaining a subset of matching web pages. Each subset induces a graph (whose nodes are the pages satisfying the conjunctive query), which can then be ranked using any centrality measure. Finally, the pages in the graph are listed in rank order as results of the query, and standard relevance measures can be applied to see how much they correspond to the available ground truth about the assessed relevance of pages to queries.

There are a few methodological remarks that are necessary before discussing the results:

- The results we present are for GOV2; there are other publicly available collections with queries and relevant documents that can be used to this purpose.
- As observed in earlier works [40], centrality scores in isolation have a very poor performance when compared with text-based ranking functions, but can improve the results of the latter. We purposely avoid measuring performance in conjunction with text-based ranking because this would introduce further parameters. Moreover, our idea is using information-retrieval techniques to judge centrality measures, not improving retrieval performance *per se* (albeit, of course, a better centrality measure could be used to improve the quality of retrieved documents).
- Because of the poor performance, even for the best documents about half of the queries have null score. Thus, the data we report must be taken with a grain of salt—confidence intervals would be largely overlapping (i.e., our experiments have limited statistical significance).
- Some methods are claimed to work better if *nepotistic links* (that is, links between pages of the same host) are excluded from the graph. We therefore report also results on the procedure applied to GOV2 with all intra-host links removed.
- There are several ways to build a graph associated with a query. Here we choose the simplest possible way—we solve the query in conjunctive form and build the induced subgraph. Variants may include enlarging the resulting graph with successors/predecessors, possibly by sampling [39].

- There are many measures of effectiveness that are used in information retrieval; among those, we focus here on the Precision at 10 (P@10, i.e., fraction of relevant documents retrieved among the first ten) and on the NDCG@10 [26].

The results obtained are presented in Table 4: even if obtained in a completely different way, they confirm the information we have been gathering with our axioms. Harmonic centrality has the best overall scores. When we eliminate nepotistic links, the landscape changes drastically—SALSA and PageRank lead now the results—but the best performances are *worse* than those obtained using the whole structure of the web. Note that, again consistently with the information gathered up to now, closeness performs very badly and betweenness performs essentially like using no ranking at all (i.e., showing the documents in some arbitrary order).

There are two new centrality measures appearing in Table 4 which deserve an explanation. When we first computed these tables, we were very puzzled: HITS is supposed to work very badly on disconnected graphs (it fails monotonicity), whereas it was the second best ranking after harmonic centrality. Also, when you eliminate nepotistic links the graphs become highly disconnected and all rankings tend to correlate with one another simply because most nodes obtain a null score. How is it possible that PageRank and SALSA work so well (albeit less than harmonic centrality on the whole graph) with so little information?

Our suspect was that *these ranking were actually picking up some much more elementary signal than their definition could make you think*. In a highly disconnected graph, the values assigned by such algorithms depends mainly on the indegree and on some additional ranking provided by coreachable (or weakly reachable) nodes.

We thus devised two somewhat paradoxically simple centrality measures, Windegree and Salsina. Windegree is simply the indegree weighted (i.e., multiplied) by the number of coreachable nodes. Salsina is the indegree multiplied by the number of weakly reachable nodes (which is somewhat similar to the way you compute SALSA). Both rankings have been designed to satisfy *all* our axioms. As it is evident from Table 4, such simple rankings outperform in this test most of the very sophisticated rankings proposed in the literature: this shows on one hand that it is possible to extract information from the graph underlying a query in very simple ways that do not involve any spectral technique, and on the other hand that designing centralities around our axioms actually pays off. We consider this fact a further confirmation that the traits of centrality represented by our axioms are important.

8 Conclusions and future work

We have presented a set of axioms that try to capture part of the intended behaviour of centrality measures. We have proved or disproved all our axioms for twelve classical centrality measures and for *harmonic centrality*, a small variant to Bavelas's closeness that we define formally in this paper for the first time. The results are surprising and confirmed by some information-retrieval experiments: harmonic centrality is a very simple measure providing a good notion of centrality. It is almost identical to closeness centrality on undirected, connected networks, but provides a centrality notion for arbitrary directed graphs.

There is of course a large measure of arbitrariness in what we have done: other researchers could come up with other axioms. We believe that this is actually a *feature*—building an ecosystem of interesting axioms is just a healthy way of understanding centrality better and less anecdotally. Promoting the growth of such an ecosystem is one of the goals of this work.

As a final note, the experiments on information retrieval that we have reported are just a start. Testing with different collections (and possibly with different ways of generating the graph associated to a query) may lead to different results. Nonetheless, we believe we have made the important point that *geometric measures are relevant not also to social networks, but also to information retrieval*. In the literature comparing exogenous (i.e., link-based) rankings one can find different instances of

All links

	NDCG@10	P@10
BM25	0.5842	0.5644
Harmonic	0.1438	0.1416
Windegree	0.1373	0.1356
HITS	0.1364	0.1349
Salsina	0.1357	0.1349
Lin	0.1307	0.1289
Katz $3/4\lambda$	0.1228	0.1242
Katz $1/2\lambda$	0.1222	0.1228
Indegree	0.1222	0.1208
Katz $1/4\lambda$	0.1204	0.1181
SALSA	0.1194	0.1221
Closeness	0.1093	0.1114
PageRank $1/2$	0.1091	0.1094
PageRank $1/4$	0.1085	0.1107
Dominant	0.1061	0.1027
PageRank $3/4$	0.1060	0.1094
Betweenness	0.0595	0.0584
—	0.0588	0.0577

Inter-host links only

	NDCG@10	P@10
BM25	0.5842	0.5644
SALSA	0.1384	0.1282
PageRank $1/4$	0.1347	0.1295
Salsina	0.1318	0.1255
PageRank $1/2$	0.1315	0.1268
PageRank $3/4$	0.1313	0.1255
Katz $1/2\lambda$	0.1297	0.1262
Windegree	0.1295	0.1262
Harmonic	0.1293	0.1262
Katz $1/4\lambda$	0.1289	0.1255
Lin	0.1286	0.1248
Indegree	0.1283	0.1248
Katz $3/4\lambda$	0.1278	0.1242
HITS	0.1179	0.1107
Closeness	0.1168	0.1121
Dominant	0.1131	0.1067
Betweenness	0.0588	0.0577
—	0.0588	0.0577

Table 4: Normalized discounted cumulative gain (NDCG) and precision at 10 retrieved documents (P@10) for the GOV2 collection using all links and using only inter-host links. The tables include, for reference, the results obtained using a state-of-the-art text ranking function, BM25, and a final line obtained by applying no ranking function at all (documents are sorted by the document identifier).

spectral rankings and indegree, but up to know that venerable measures based on distances have been neglected. We suggest that it is time to change this attitude.

9 Acknowledgements

We thank David Gleich for useful pointers leading to the proof of the monitonicity axiom for PageRank, and Edith Cohen for useful discussions on the behaviour of centrality indices. Marco Rosa participated to the first phases of development of this paper.

References

[1] Alon Altman and Moshe Tennenholtz. Ranking systems: the PageRank axioms. In *Proceedings of the 6th ACM conference on Electronic commerce*, pages 1–8. ACM, 2005.

[2] Jac M. Anthonisse. The rush in a graph. Technical report, Amsterdam: University of Amsterdam Mathematical Centre, 1971.

[3] Lars Backstrom, Paolo Boldi, Marco Rosa, Johan Ugander, and Sebastiano Vigna. Four degrees of separation. In *ACM Web Science 2012: Conference Proceedings*, pages 45–54. ACM Press, 2012. Best paper award.

[4] A. Bavelas. A mathematical model for group structures. *Human Organization*, 7:16–30, 1948.

[5] A. Bavelas, D. Barrett, and American Management Association. *An experimental approach to organizational communication*. Publications (Massachusetts Institute of Technology. Dept. of Economics and Social Science).: Industrial Relations. American Management Association, 1951.

[6] Alex Bavelas. Communication patterns in task-oriented groups. *Journal of the Acoustical Society of America*, 1950.

[7] Murray A. Beauchamp. An improved index of centrality. *Behavioral Science*, 10(2):161–163, 1965.

[8] Abraham Berman and Robert J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Classics in Applied Mathematics. SIAM, 1994.

[9] Paolo Boldi, Marco Rosa, and Sebastiano Vigna. Robustness of social and web graphs to node removal. *Social Network Analysis and Mining*, 2013.

[10] Paolo Boldi, Massimo Santini, and Sebastiano Vigna. PageRank: Functional dependencies. *ACM Trans. Inf. Sys.*, 27(4):1–23, 2009.

[11] Phillip Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1):113–120, 1972.

[12] Stephen P. Borgatti. Centrality and network flow. *Social Networks*, 27(1):55–71, 2005.

[13] Allan Borodin, Gareth O. Roberts, Jeffrey S. Rosenthal, and Panayiotis Tsaparas. Link analysis ranking: algorithms, theory, and experiments. *ACM Transactions on Internet Technology (TOIT)*, 5(1):231–297, 2005.

[14] Alfred Brauer. Limits for the characteristic roots of a matrix. IV: Applications to stochastic matrices. *Duke Math. J.*, 19:75–91, 1952.

[15] Robert L. Burgess. Communication networks and behavioral consequences. *Human Relations*, 22(2):137–159, 1969.

[16] Edith Cohen and Haim Kaplan. Spatially-decaying aggregation over a network. *Journal of Computer and System Sciences*, 73(3):265–288, 2007.

[17] Reuven Cohen and Shlomo Havlin. *Complex Networks: Structure, Robustness and Function*. Cambridge University Press, 2010.

[18] B.S. Cohn and M. Marriott. Networks and centres of integration in Indian civilization. *Journal of Social Research*, 1:1–9, 1958.

[19] Nick Craswell, David Hawking, and Trystan Upstill. Predicting fame and fortune: Pagerank or indegree. In *In Proceedings of the Australasian Document Computing Symposium, ADCS2003*, pages 31–40, 2003.

[20] Ayman Farahat, Thomas Lofaro, Joel C. Miller, Gregory Rae, and Lesley A. Ward. Authority rankings from HITS, PageRank, and SALSA: Existence, uniqueness, and effect of initialization. *SIAM Journal on Scientific Computing*, 27:1181–1201, 2006.

[21] L. Freeman. Centrality in social networks: Conceptual clarification. *Social Networks*, 1(3):215–239, 1979.

[22] Linton C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.

[23] N.E. Friedkin. Theoretical foundations for centrality measures. *The American Journal of Sociology*, 96(6):1478–1504, 1991.

[24] Roger A. Horn and Stefano Serra-Capizzano. A general setting for the parametric Google matrix. *Internet Math.*, 3(4):385–411, 2006.

[25] Charles H. Hubbell. An input-output approach to clique identification. *Sociometry*, 28(4):377–399, 1965.

[26] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002.

[27] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[28] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, September 1999.

[29] Donald E. Knuth. Two notes on notation. *American Mathematical Monthly*, 99(5):403–422, May 1992.

[30] H. J. Leavitt. Some effects of certain communication patterns on group performance. *J Abnorm Psychol*, 46(1):38–50, January 1951.

[31] Ronny Lempel and Shlomo Moran. SALSA: the stochastic approach for link-structure analysis. *ACM Trans. Inf. Syst.*, 19(2):131–160, 2001.

[32] Lun Li, David L. Alderson, John Doyle, and Walter Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Math.*, 2(4), 2005.

[33] Nan Lin. *Foundations of Social Research*. McGraw-Hill, New York, 1976.

[34] Kenneth Mackenzie. Structural centrality in communications networks. *Psychometrika*, 31(1):17–25, 1966.

[35] Massimo Marchiori and Vito Latora. Harmony in the small-world. *Physica A: Statistical Mechanics and its Applications*, 285(3-4):539 – 546, 2000.

[36] J.J. McDonald, M. Neumann, H. Schneider, and M.J. Tsatsomeros. Inverse m-matrix inequalities and generalized ultrametric matrices. *Linear Algebra and its Applications*, 220:321–341, 1995.

[37] Carl D. Meyer. *Matrix analysis and applied linear algebra*. Society for Industrial and Applied Mathematics, pub-SIAM:adr, 2000.

[38] Stanley Milgram. The small world problem. *Psychology Today*, 2(1):60–67, 1967.

[39] Marc Najork, Sreenivas Gollapudi, and Rina Panigrahy. Less is more: sampling the neighborhood graph makes salsa better and faster. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 242–251. ACM, 2009.

[40] Marc Najork, Hugo Zaragoza, and Michael J. Taylor. HITS on the web: how does it compare? In Wessel Kraaij, Arjen P. de Vries, Charles L. A. Clarke, Norbert Fuhr, and Noriko Kando, editors, *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007*, pages 471–478. ACM, 2007.

[41] Marc A. Najork, Hugo Zaragoza, and Michael J. Taylor. HITS on the web: how does it compare? In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 471–478. ACM, 2007.

[42] John F. Padgett and Christopher K. Ansell. Robust Action and the Rise of the Medici, 1400-1434. *The American Journal of Sociology*, 98(6):1259–1319, 1993.

[43] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, Stanford University, Stanford, CA, USA, 1998.

[44] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, Stanford University, Stanford, CA, USA, 1998.

[45] Forrest R. Pitts. A graph theoretic approach to historical geography. *The Professional Geographer*, 17(5):15–20, 1965.

[46] John R. Seeley. The net of reciprocal influence: A problem in treating sociometric data. *Canadian Journal of Psychology*, 3:234–240, 1949.

[47] Claude E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:379–423, 623–656, 1948.

[48] Karen Stephenson and Marvin Zelen. Rethinking centrality: Methods and examples. *Social Networks*, 11(1):1 – 37, 1989.

[49] Trystan Upstill, Nick Craswell, and David Hawking. Query-independent evidence in home page finding. *ACM Trans. Inf. Syst.*, 21(3):286–313, 2003.

[50] Sebastiano Vigna. Spectral ranking, 2009.

[51] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*. Cambridge Univ Press, 1994.

[52] T.H. Wei. The algebraic foundations of ranking theory, 1952.