A Measurement of Mixing Time in Social Networks

Matteo Dell'Amico[1] Yves Roudier[2]

Eurecom
2229, route des Crêtes, BP 193
F-06560 Sophia-Antipolis cedex, France

Abstract

Transitive trust propagation on webs of trust (social networks representing trust relationships between individuals) is a well-known pattern used to compute reputation, to defend against Sybil attacks and to introduce incentives to cooperation in peer-to-peer applications. Based on the fact that it is difficult for an attacker to get trusted by many honest users, it is possible to prove that some methods based on transitive trust propagation are indeed resilient to attacks mounted by malicious users.
The other side of the coin is that honest users can be mistaken for malicious ones: such an event is likely to happen when the web of trust has a high mixing time. Performing a random walk over the web of trust, mixing time is the number of steps needed to ensure that the landing point of the random walk does not depend on the starting node.
To help assessing the effectiveness of methods based on transitive trust propagation, we measure the mixing time of four large and publicly available webs of trust. Our experimental results show that mixing time is not directly related with well-known characteristics such as degree distribution or clustering coefficient. Rather, they suggest that mixing time is an interesting metrics on its own, explained by the presence or absence of long-range links in the social network.

Keywords: Mixing time, social networks, trust, transitive trust propagation

1 Introduction

Trust value computation is an important application of social networks: to decide whether a user deserves to be trusted, endorsements received by peers are taken in consideration. This approach can be used to secure collaborative applications, to introduce incentives to cooperation in P2P networks, and to defend against Sybil attacks.

The idea of transitive trust stems from a circular definition: given a network of endorsements between users (a "web of trust"), a trusted user is one which is endorsed by other trusted users. This idea is reflected into a pattern where trust gets propagated "transitively" (e.g., "I trust people that my friends trust") from some pre-trusted nodes along paths in the social network. Underlying this is the

[1] Email: matteo.dell-amico@eurecom.fr
[2] Email: yves.roudier@eurecom.fr

assumption that honest users are connected to each other via a tight network of connections, while malicious and/or fake ones won't be able to trick more than a few honest users into trusting them. Even if malicious users collude or create fake identities, this won't create problems since the endorsements received from untrusted users won't be taken in consideration.

A well-designed system can ensure that it is difficult for a malicious attacker to get trusted. However, another issue must be taken into account: honest users should be trusted. This happens more often if the former assumption – that is, having tight trust relationships between honest nodes – is verified. When this happens, the social network is *fast mixing*: after a short random walk over the network, the probability distribution describing the endpoint of this random walk rapidly converges to the stationary distribution of this random walk. By measuring mixing speed in a network, we can therefore help assess the effectiveness of trust computation algorithms.

The small-world property discovered by Watts and Strogatz [21] is related to mixing speed: small-world networks, while possibly highly clustered, have short paths that connect arbitrary pairs of nodes. A fast mixing graph is necessarily a small world, but the opposite is not true: even if short paths that connect arbitrary pairs of nodes exist, random walks over the network can often remain confined in a local cluster. For a small world network to be fast mixing, the "long range" edges that connect different clusters do not only need to exist: they have to be a substantial percentage, since otherwise most of the random walks would remain confined within the originating cluster.

The small world property is widely observed in many kind of networks and it is virtually ubiquitous in social network [18]. On the other hand, while the relevance of mixing time in trust propagation algorithms is acknowledged [23,12], not much is known about mixing time of real social networks. Systems that depend on mixing speed are typically validated by running the trust propagation algorithm on a single dataset and evaluating the performances of the algorithm under scrutiny [22], without explicitly taking mixing speed into account. This work strives to fill this gap by measuring mixing time on four different large-scale publicly available webs of trust, by discussing the differences observed between them and by explaining these differences with a model based on the long-range connections between nodes in different large clusters.

The rest of this paper is structured as follows: Section 2 discusses the relevance of mixing speed with respect to applications in social networks; Section 3 introduces the datasets and Section 4 shows measurements performed on them; in Section 5 we interpret the data obtained and propose a model describing mixing speed based on the synthetic model for navigable small worlds created by Jon Kleinberg [11]. Section 6 concludes.

2 Background

This section introduces the concept of transitive trust on webs of trust and discusses the importance of mixing speed in measures that rely on transitive trust propagation.

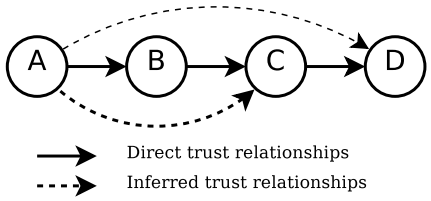

Fig. 1. Transitive trust propagation. User A trusts B, who in turn trusts C. Since B is considered a good recommender, C gets also trusted by A. This mechanism can happen recursively, so a (possibly lower) degree of trust is propagated also to D.

2.1 Transitive Trust

A *web of trust* is a social network where connections express trust between individuals. It may be modeled as either a directed or an undirected graph; in directed graphs, an $A \to B$ edge expresses the information that A trusts B, while an undirected edge connecting the two nodes represents mutual trust between the two principals. An undirected web of trust can easily be converted to a directed network where an undirected edge between A and B maps to both the $A \to B$ and the $B \to A$ edges. Without loss of generality, we will thus refer to directed networks in the following.

The core assumption of transitive trust is exemplified in Figure 1: trusted users are considered faithful recommenders. Therefore, users that are recommended by trusted users get trusted themselves, albeit possibly to a lower degree. This assumption is not, in general, verified – in plenty of cases, people do not get along with the friends of their friends. It is however a valuable heuristic, yielding both good results and resilience to attack in many practical cases, as the rest of this section will attest.

2.2 Mixing Time

As definition of mixing speed, we adopt the one described in the book of Mitzenmacher and Upfal [16]. A random walk on a web of trust can be described by a Markov chain whose states represent users and transitions represent web of trust links. A unique equilibrium probability distribution $\overline{\pi}$ for this random walk is guaranteed to exist for all connected undirected graphs, and for all aperiodic strongly connected directed graphs. A sufficient condition for a graph to be aperiodic is to have cycles of lengths with greatest common divisor of 1. Since all of our graphs are strongly connected and have cycles of length 2 and of length 3, the uniqueness of the stationary distribution is guaranteed.

The *variation distance* between two probability distributions D_1 and D_2 on a countable state space S is defined as

$$\|D_1 - D_2\| = \frac{1}{2} \sum_{x \in S} |D_1(x) - D_2(x)| .$$

The variation distance between the probability distribution p_x^t of a random walk of length t starting from node x and the stationary distribution $\overline{\pi}$ is

$$\Delta_x(t) = \left\| p_x^t - \overline{\pi} \right\| . \tag{1}$$

Given a threshold ε, we can now define the mixing time for node x as

$$\tau_x(\varepsilon) = \min\{t : \Delta_x(t) \leq \varepsilon\}$$

and the mixing time for the whole network as

$$\tau(\varepsilon) = \max_{x \in S} \tau_x(\varepsilon).$$

$\tau(\varepsilon)$ is therefore the length of time needed to ensure that the deviation between the stationary probability distribution and p_x^t falls below ε for all values of x.

A network is regarded as fast mixing if $\tau(\varepsilon)$ is $O(\log n)$.

2.3 Trust Propagation and Mixing Speed

Many families of algorithms for computing reputation have been proposed based on the transitive pattern; from them, we can highlight two recurrent strategies:

(i) Based on random walks: these approaches compute the probability that a random walk that follows trust links starting from trusted nodes would land onto a given node. The seminal work introducing this approach is PageRank [19], the algorithm used by Google to rank web page relevance; algorithms using variations of this idea have been extensively used as foundation for trust metrics. For a review, see [10].

(ii) Based on flow: the trust value given to a node is proportional to the maximal flow (or, equivalently, the minimal cut) in the web of trust having pre-trusted nodes as sources and the target node as sink [13,8].

Mixing time is correlated with the quotient cut problem [9]: finding a small number of edges which disconnect a large number of nodes from the rest of the graph. If a small quotient cut exists, then the network is slow mixing (and the maximal flow between pairs of nodes will often be low); on the contrary, a fast mixing network has no small quotient cut. Given the relationship between network cut and mixing speed, in both cases the trust metrics distribute trust effectively between well-intentioned nodes only if the network is fast-mixing. This means that, in a slow-mixing network, the case where well-intentioned nodes are incorrectly recognized as malicious will be frequent.

2.4 Sybil Attacks

Social networks and transitive trust propagation can be used to defend against Sybil attacks, i.e. the cases where an attacker creates many malicious identities with the goal of subverting a P2P system. Among them, two approaches can be isolated:

(i) Sybil attack-resistant reputation metrics [13,4,14]: the number of Sybil nodes that will be able to cooperate in the network can be unlimited, but those nodes will have low reputation. The application needs to be carefully designed so that low-reputation nodes cannot damage the system.

(ii) Systems that aim to actually limit the number of Sybil identities [24,22,5] participating in the system. This approach is less flexible in the sense that nodes that are tagged as Sybils cannot interact with the rest of the network. Such

an approach can also be implemented by creating a reputation system of the former type, and only accepting in the P2P network nodes whose reputation score exceeds a given threshold.

For these systems to be efficient, it is essential that the social network is fast-mixing: when clusters of honest nodes are separated, some of them may be mistaken for Sybils. Again, a slow mixing network would result in bad performances for the algorithm taken in consideration.

2.5 Mix Networks

Mix networks are designed to anonymize communications by relaying them via chains of proxy using nested encryption. Nagaraja [17] investigated the usage of social networks as underlying topologies, letting data be exchanged only between friends, leveraging on the trust between them to ensure cooperation. The fast mixing property is essential to ensure that the exit point of the proxy chain is not correlated with the originator of the message without needing to resort to extremely long proxy chains. Nagaraja measured the entropy of the probability distribution for the exit point on various synthetic networks and one real social network from LiveJournal. With respect to his work, we focus on measuring and comparing various real-world social networks, showing that the mixing time can be very different from network to network.

2.6 Other Applications

Fast mixing time is also essential to provide fast convergence for gossip algorithms [2,6]: knowing the mixing time of social networks is essential to discuss the feasibility of these algorithms in darknets (or "friend-to-friend networks") [20]: peer-to-peer networks where each node establishes connections only to trusted acquaintances. If the social network has an adequately fast mixing time, then the algorithms can be efficient enough to be deployed unmodified on the social network. Conversely, a slow-mixing social network would not be adequate for this kind of applications.

Transitive trust propagation has also been used in recommender systems [15,25,7], based on the *homophily* property stating that "birds of a feather flock together": friends in the social network will have similar preferences. In this case, a slow mixing social network will result in stronger personalization of the recommendation, while a fast mixing social network will result in recommendations which are more homogeneous between users.

3 Datasets

In this section, we describe the datasets on which we performed our mixing speed measurement. The Advogato, DBLP and Epinions datasets were obtained from the Trustlet.org website [3]; the OpenPGP web of trust has been obtained via the "Web of trust statistics and pathfinder" website [4] developed by Jörgen Cederlöf. All the

[3] http://www.trustlet.org/wiki/Datasets
[4] http://www.lysator.liu.se/~jc/wotsap/

graphs have been managed using the Python NetworkX library [5].

- *Advogato* [6] is an online community devoted to free software. Its participants certify each other as belonging to different levels of trust (Observer, Apprentice, Journeyer and Master). A custom trust metric [13] assigns one of those labels to each user of the website, assigning them different rights (for example, untrusted users cannot post stories on the front page). In our analysis, we didn't take into account the certification level, considering the network as unweighted.
- *DBLP* [7] is a computer science bibliography listing more than 1.2 millions publications. The web of trust, in this case, is the graph connecting authors who co-authored papers.
- *Epinions* [8] is a website of consumer-generated reviews. Users register for free and write reviews about any kind of product; they can actually be paid if their reviews are found useful. Each user can add others to their "web of trust", i.e. the set of reviewers "whose reviews and ratings have been consistently found to be valuable" [9].
- *OpenPGP* is a standard for privacy and authentication based on public-key cryptography [3]. The system relies on mutual authentication of public keys: users certify each other that a public key belongs to a given individual they met in person, and make those signed attestations public by uploading them to a network of "keyservers". No trusted central certification authority is required: a user's identity can be validated if some other trusted user has signed their key. It is sometimes customary to organize "key-signing parties" where attendants verify each other's identities and mutually certify their keys.

Since the notion of equilibrium distribution for random walks has meaning only on connected components, we performed our measurements only on the biggest ("giant") strongly connected component (SCC) of each dataset. The giant components of each network have sizes which is orders of magnitude larger than all other SCCs in the network; moreover, even if many nodes do not belong to the giant components, most of the links of the networks belong to them. This is explained by the fact that nodes outside the giant SCC are generally "less active", with none or few links between them. Information about the strongly connected components in our datasets is synthesized in Table 1 on the following page.

4 Measurements

In Table 2 on the following page, we show various key features of our datasets. For the undirected case, the degree is the number of incident edges per node; in a directed network, the out-degree and in-degree refer respectively to outgoing and incoming edges. Obviously, the average outdegree and average indegree have the same value.

[5] http://networkx.lanl.gov/
[6] http://advogato.org/
[7] http://www.informatik.uni-trier.de/~ley/db/
[8] http://www.epinions.com/
[9] From the Epinions Web of Trust FAQ (http://www.epinions.com/help/faq/?show=faq_wot).

Network	Giant SCC nodes	Giant SCC edges	Second biggest SCC size
Advogato	23.59%	82.27%	3
DBLP	84.78%	94.82%	46
Epinions	31.88%	83.98%	15

Table 1
Percentage of nodes and edges belonging to the biggest ("giant") strongly connected components. All other connected components have size orders of magnitude smaller. Most of the edges belong to the giant strongly connected component, suggesting that nodes outside of it are peripheric and less active in the social network. No data is available about OpenPGP, since our source already excluded nodes not belonging to the giant strongly connected component.

Network	Type	Nodes	Edges	Avg. degree	C
Advogato	Directed	3,254	47,227	14.51	0.2527
DBLP	Undirected	565,612	2,087,803	7.38	0.6409
Epinions	Directed	36,490	602,722	16.52	0.1763
OpenPGP	Directed	41,292	414,424	10.04	0.3641

Table 2
Statistics about the giant strongly connected components of our datasets. The C column refers to the clustering coefficient defined in Equation 2.

4.1 Clustering Coefficient

The clustering coefficient C is a measure introduced by Watts and Strogatz [21] to describe "small world" networks. For a node i with degree k_i in an undirected network $G = (V, E)$, the clustering coefficient is defined as

$$C_i = \frac{2\left|\{e_{jk} \in E \,\text{such that}\, e_{ij} \in E \wedge e_{ik} \in E\}\right|}{k_i\left(k_i - 1\right)},$$

that is the number of neighbors of i which are connected by an edge divided by all possible $\frac{k_i(k_i-1)}{2}$ pairs. In directed graphs, since there may be two edges between j and k ($j \to k$ and $k \to j$), this value becomes

$$C_i = \frac{\left|\{e_{jk} \in E \,\text{such that}\, e_{ij} \in E \wedge e_{ik} \in E\}\right|}{k_i\left(k_i - 1\right)},$$

The value C representative of the whole network is obtained by simply averaging all the C_i values:

$$C = \frac{\sum_{i \in V} C_i}{|V|}. \tag{2}$$

The clustering coefficient C measures the "cliquishness" of a graph by taking into account the local neighborhoods of each node. Networks where most links are part of tightly-connected communities (i.e., subgraphs where each node is connected to many of the others) have high clustering coefficient. Intuition may suggest that networks with higher C have slower mixing times, but, as we will show, our evidence shows that these measures are not obviously correlated.

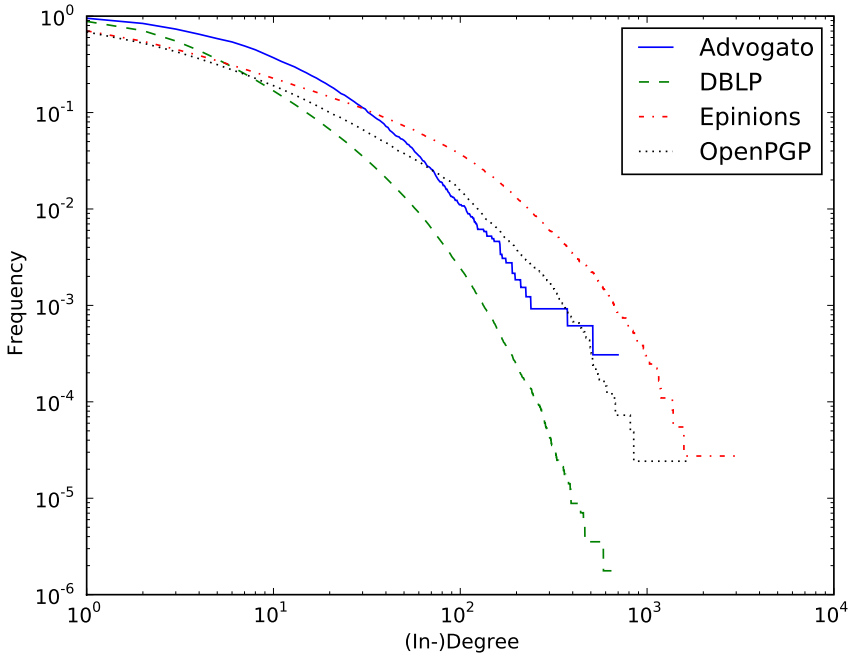

Fig. 2. Degree distribution: frequency of nodes with degree *higher* than x. It can be described as 1 minus the CDF (cumulative distribution function) of node degrees, and would result in a straight line in this log-log plot for a power-law degree distribution.

4.2 Degree Distribution

Figure 2 shows the degree distribution of these datasets. In all cases, there are "hubs" with a number of links which is much higher than the average, but the degree distribution does not correspond to the power laws which are often observed in complex networks [1]. It could be imagined that the presence of large hubs would lead to faster mixing (after all, many different paths would rapidly converge to the same hub) but, again, we will not observe a clear correlation between these features in our datasets.

4.3 Mixing Time

Since the mixing time defined in Section 2.2 is an asymptotic behavior, it is not possible to directly measure it in our datasets. We can however measure the variation distance $\Delta_x(t)$ from the equilibrium of Equation 1 on page 3 and show how quick is its convergence to 0.

In our experiments, we computed the values of $\Delta_x(t)$ for a random sample of 1,000 nodes on each network. The values of p_x^t can be computed in a straightforward way by representing the Markov chain as a matrix and multiplying the starting probability distribution by that matrix for t times; the equilibrium distribution is the dominant eigenvector of that matrix. To perform these calculations, we adopted the sparse matrix library included in the Python SciPy package [10].

In Figure 3 on the following page we plot the variation distance $\Delta_x(t)$ defined in Equation 1 on page 3 against t for each of our datasets. It can be easily noticed that mixing speeds are very different between the four datasets: OpenPGP has by far

[10] http://www.scipy.org/

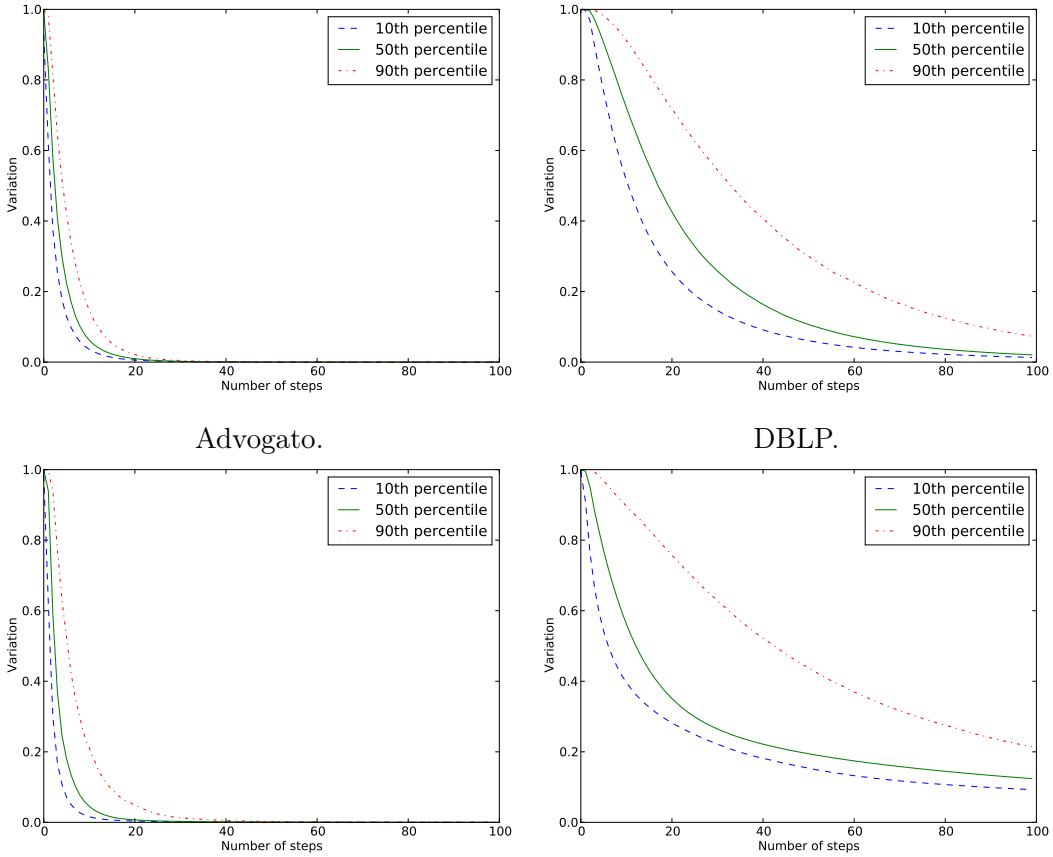

Fig. 3. Mixing time: variation distances $\Delta_x(t)$ from Equation 1 on page 3.

the slowest mixing speed; the mixing speed of DBLP is slow as well, while a random walk in Epinions and Advogato converges close to the stationary distribution in as little as 20 steps.

5 Discussion

Network sizes and degree distributions does not seem closely related with mixing time: the Epinions and the OpenPGP networks are similar in size, but they have completely different characteristics in term of mixing time. Moreover, a "fast mixing" network is defined as having mixing time logarithmic with respect to size; even the two orders of magnitude of size difference between the smaller Advogato network (3 thousands nodes, 47 thousands edges) and the larger DBLP network (566 thousands nodes, 2 millions edges) would not justify such a difference in terms of mixing time. The degree distribution also does not appear closely related with mixing time: for example, the Advogato network has smaller "hubs" than OpenPGP and Epinions but has faster mixing than both.

The clustering coefficient (reported in Table 2 on page 7) may appear as a more correlated measure: after all, a network where nodes are divided into tightly-knit subcommunities should have a high clustering coefficient (because two members of

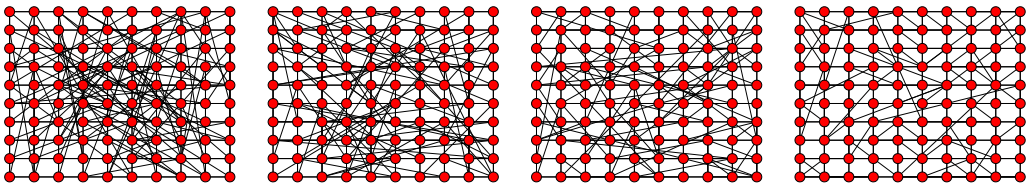

Fig. 4. Kleinberg graphs with the same number of nodes and edges. As the α parameter grows, the graph appears less visually cluttered because long-range links become shorter.

the same community are likely to be connected to a common third neighbor) and slow mixing speed (because random walks are going to remain often confined in the same cluster). Results do not confirm this intuition either: while it is true that the two "slow mixing" networks (OpenPGP and DBLP) have higher clustering coefficients than Advogato and Epinions, OpenPGP has a definitely slower mixing speed than DBLP while having a lower clustering coefficient; the same can be said about respectively Advogato and Epinions.

We believe that, while both clustering coefficient and mixing speed denote the presence or absence or clusters, the clusters they identify belong to different scales: while a high clustering coefficient is indicative of the presence of many small communities such as cliques of friends, a low mixing speed is instead indicative of the presence of larger communities (for example, nations) that have a much lower connection density.

It is very indicative that the two "slow-mixing" networks are those where acquaintances are usually a consequence of physical meetings: people usually meet in the same place to write a paper together (and establish a connection in the DBLP network), or when they exchange the fingerprints of their OpenPGP public keys. This makes it quite unlikely that two people residing in different nations establish a connection. For a random walk to escape the large-scale cluster of a given nation or continent, the random walk has to encounter a node with connections outside this cluster and actually choose one of those connections as exit point for the random walk. On the other hand, a purely Internet-based community such as Epinions or Advogato encourages the creation of links between different parts of the globe. We attribute the difference in mixing speed between DBLP and OpenPGP to the fact that computer science researchers are more likely to travel and establish connections across the world than ordinary users of cryptographic applications.

In our view, the main feature differentiating fast-mixing networks and slow-mixing ones is the presence of many long-range links connecting users in completely unrelated places or communities. A network with a high clustering coefficient can be fast mixing if the links escaping the tightly-connected cliques lead to "far away", unrelated, communities. Conversely, a network with a low clustering coefficient can be slow mixing if nodes are divided into large, sparse communities, and very few edges connect nodes in these communities.

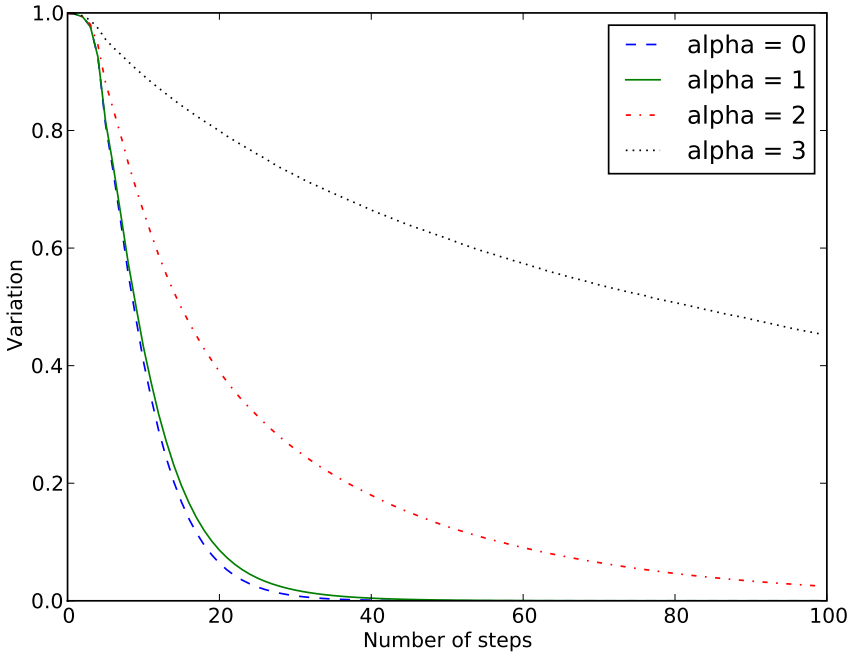

Fig. 5. Median variation distances $\Delta_x(t)$ for Kleinberg networks generated with different values of α. This experiment has been performed on 100×100 grids containing thus 10,000 nodes.

5.1 The Kleinberg Model

We claim that network mixing speed depends eminently on long-range links. To further support this explanation, we adopted Jon Kleinberg's model for navigable small world networks [11]. In this model, nodes belong to a 2D grid network mimicking a clustered social structure: each node is connected to its neighbors in the grid; each u node also adds a directed long-range link to another random node. The probability of choosing a node v is proportional to $r^{-\alpha}$, where r is the "Manhattan" distance between u and v (number of grid links that must be traversed in order to reach v from u). The α parameter influences the length of these random links, with higher α resulting in shorter connections: social networks where far-away clusters are unlikely to be connected are represented by a higher value of α. Figure 4 on the previous page shows some small graphs generated with Kleinberg's model.

We measured the mixing speed in synthetic Kleinberg networks in the same way we did with the real-world social networks described earlier in this paper. In Figure 5, we plot the variation distances with respect to the equilibrium distribution. Confirming our hypothesis, the α parameter controlling the length of random edges has a dramatic impact on the network mixing time. It is interesting to note that the mixing time does not change much when passing from $\alpha = 1$ (the probability of choosing a node is inversely proportional to its distance) to $\alpha = 0$ (endpoints for long-range edges are chosen uniformly at random). We interpret this by the fact that, when $\alpha = 1$, most of the random edges lead to distant nodes anyway. In fact, the cumulative probability of selecting far away endpoints is higher than for close ones, simply because the far nodes outnumber the neighbors.

6 Conclusions

Mixing time is an interesting metric, succinctly describing the presence or absence of large-scale isolated communities in a network. When characterizing a complex network, it is a useful addition to known metrics such as clustering coefficient, average diameter and degree distribution.

Our study shows that mixing time is very different across networks, and we proposed an explanation based on the "length" of links connecting nodes in different large-scale clusters. This hypothesis is substantiated by the fact that the slowest-mixing datasets that we analyzed are those requiring physical interaction for the creation of connections. On the other hand, the fast mixing speed of Internet-based networks can be attributed to the ease of creating social links between users belonging to very different communities. To describe network mixing speed, we reused the model developed by Kleinberg to describe navigable small worlds network: confirming our hypothesis, the parameter tuning the length of random links in the network has a dramatic influence over the mixing speed.

Knowing the mixing time of networks is important to design effective algorithms. Reputation metrics rely on fast mixing time in order to discriminate effectively between honest and lazy, malicious or fake nodes. Mixing speed is also essential when considering gossiping algorithms, "friend-to-friend" darknets, and networks representing homophily for recommender systems.

Our results also yield an interesting lesson that can be applied to webs of trust and reputation systems: insisting that users add to their webs of trust only well-known peers results in better quality of trust links. This makes it harder for malicious users to obtain high reputation values; on the other hand, our results show that this can have the unwelcome effect of discouraging the long-range links that are essential to make reputation systems work smoothly, and honest users may suffer from being not trusted. This trade-off should be carefully taken into account when designing reputation systems.

References

[1] Barabasi, A. L. and R. Albert, *Emergence of scaling in random networks*, Science **286** (1999), pp. 509–512.

[2] Boyd, S., A. Ghosh, B. Prabhakar and D. Shah, *Randomized gossip algorithms*, IEEE/ACM Trans. Netw. **14** (2006), pp. 2508–2530.

[3] Callas, J., L. Donnerhacke, H. Finney, D. Shaw and F. Thayer, *RFC 4880 - OpenPGP message format*, Technical report, Internet Engineering Task Force (2007).
URL http://tools.ietf.org/html/rfc4880

[4] Cheng, A. and E. Friedman, *Sybilproof reputation mechanisms*, in: *P2PECON '05: Proceeding of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems* (2005), pp. 128–132.

[5] Danezis, G. and P. Mittal, *SybilInfer: Detecting Sybil nodes using social networks*, in: *Proc. of NDSS 2009*, 2009.

[6] Dell'Amico, M., *Mapping small worlds*, in: *Peer-to-Peer Computing, 2007. P2P 2007. Seventh IEEE International Conference on*, 2007, pp. 219–228.

[7] Dell'Amico, M. and L. Capra, *SOFIA: Social filtering for robust recommendations*, in: *Proc. of IFIPTM 2008*, 2008, pp. 135–150.

[8] Feldman, M., K. Lai, I. Stoica and J. Chuang, *Robust incentive techniques for peer-to-peer networks*, in: *ACM Conference on Electronic Commerce*, 2004, pp. 102–111.

[9] Guruswami, V., *Rapidly mixing Markov chains: A comparison of techniques* (2000).
URL http://www.cs.washington.edu/homes/venkat/pubs/papers/markov-survey.ps

[10] Hussain, F. K., E. Chang and O. K. Hussain, *State of the art review of the existing PageRank-based algorithms for trust computation*, in: *Second International Conference on Systems and Networks Communications (ICSNC 2007)*, 2007, p. 75.

[11] Kleinberg, J. M., *Navigation in a small world*, Nature **406** (2000).

[12] Laas, C. L., *A Sybil-proof one-hop DHT*, in: *SocialNets '08: Proceedings of the 1st workshop on Social network systems* (2008), pp. 19–24.

[13] Levien, R., *Advogato's trust metric* (2000).
URL http://www.advogato.org/trust-metric.html

[14] Massa, P., *Reputation is in the eye of the beholder: on subjectivity and objectivity of trust statements*, in: *Security Issues in Reputation Systems*, 2007.
URL http://www.gnuband.org/2007/06/09/reputation_is_in_the_eye_of_the_beholder_on_subjectivity_and_objectivity_of_trust_statements/

[15] Massa, P. and P. Avesani, *Trust-aware recommender systems*, in: *Proceedings of ACM Recommender Systems Conference, Minneapolis, Minnesota, USA*, 2007.

[16] Mitzenmacher, M. and E. Upfal, "Probability and Computing: Randomized Algorithms and Probabilistic Analysis," Cambridge University Press, New York, NY, USA, 2005.

[17] Nagaraja, S., *Anonymity in the wild: Mixes on unstructured networks*, in: *Privacy Enhancing Technologies*, 2007, pp. 254–271.

[18] Newman, M. E. J., *The structure and function of complex networks*, SIAM Review **45** (2003), pp. 167–256.

[19] Page, L., S. Brin, R. Motwani and T. Winograd, *The PageRank citation ranking: Bringing order to the web*, Technical report, Stanford Digital Library Technologies Project (1998).

[20] Sandberg, O., *Distributed routing in small-world networks*, in: *ALENEX 2006*, 2006.

[21] Watts, D. J. and S. H. Strogatz, *Collective dynamics of 'small-world' networks.*, Nature **393** (1998), pp. 440–442.

[22] Yu, H., P. B. Gibbons, M. Kaminsky and F. Xiao, *Sybillimit: A near-optimal social network defense against Sybil attacks*, in: *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, 2008, pp. 3–17.

[23] Yu, H., M. Kaminsky, P. B. Gibbons and A. Flaxman, *Discussion on the SIGCOMM 2006 paper "SybilGuard: Defending against sybil attacks via social networks"* (2006).
URL http://conferences.sigcomm.org/sigcomm/2006/discussion/showpaper.php?paper_id=26

[24] Yu, H., M. Kaminsky, P. B. Gibbons and A. Flaxman, *Sybilguard: defending against Sybil attacks via social networks*, in: *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications* (2006), pp. 267–278.

[25] Zheng, R., F. Provost and A. Ghose, *Social network collaborative filtering*, Technical report, New York University (2007).
URL http://hdl.handle.net/2451/23407