

Towards Robust and Scalable Peer-to-Peer Social Networks

Alexandra Olteanu

EPFL and Vrije Universiteit Amsterdam
alexandra.olteanu@epfl.ch

Guillaume Pierre

Vrije Universiteit Amsterdam
guillaume.pierre@vu.nl

Abstract

Peer-to-peer Online Social Networks (OSNs) promise to combine the functionalities of centralized OSNs with the good properties of peer-to-peer systems. However, in time, the number of connections between users of OSNs grows super-linearly in the number of users: the average node degree increases with the overall system size. In large-scale settings, mapping each friendship relation into an overlay link will thus overwhelm popular nodes. We propose simple techniques to build and maintain the peer-to-peer OSN overlay while significantly shifting node degrees towards lower ranges. We evaluate our algorithms using real large-scale datasets, and show that they can disseminate information efficiently while controlling node degrees, even in the presence of high churn.

Categories and Subject Descriptors H.3.4 [Systems and Software]: Distributed Systems; C.2.1 [Network Architecture and Design]: Distributed networks

Keywords Peer-to-peer, Online Social Networks, Scalability, Accesibility, Availability

1. Introduction

Online Social Networks (OSNs) are woven into a main part of online operations, and users are increasingly prone to disclosing their privacy on such media. In essence, the commercial transaction between an OSN and its users works as follows: the OSN provides and operates an expensive platform for users, without charging them for usage. In return, users donate their private information, which constitutes the only asset an OSN can exploit to make profits. To allow end users to enjoy the functionalities of an OSN without giving up on their privacy, an alternative is to structure the OSN as a peer-to-peer (P2P) network [1, 3–6, 8, 9, 11, 12, 15–17].

Scalability is an often-overlooked issue in P2P OSNs. A common belief is that the average node degree is constant, independent of the evolution of system size. However, it has been shown that, in time, the average node degree grows polynomially in the number of nodes [13, 14], while the effective diameter shrinks. As a result, currently more than 50% of Facebook users have hundreds or more friends, a fraction that is bound to increase in time [2, 19]. In this context, the cost of mirroring the social network links onto P2P overlay links can be prohibitively high. Hence, the P2P OSN needs to trim the active connections per node, thereby enabling the system to scale and remain practical.

This paper shows how a P2P OSN overlay can exploit the high clustering coefficient of the social graph to ensure its own robustness and scalability. The clustering coefficient is defined as the probability that two nodes which are linked to the same third node, are themselves linked together. For example, Facebook’s clustering coefficient remains large regardless of the node degree [19]. Additionally, a node’s friends of friends have a significant number of friends in common with the node. For instance, for a node with degree 100, more than 50% of the friends of its friends are non-unique. This means that a social network is extremely robust to edge removals and highlights the possibility of aggregating multiple edges into one compound link that serves multiple social relations.

We start by organizing the P2P overlay such that overlay links between nodes are mapped onto social links between users. A one-to-one mapping would obviously incur many problems, such as the necessity for the system to maintain large number of connections per node. Instead, we largely reduce the number of overlay links such that the resulting system implements publish-subscribe (pub-sub) functionality. Each user is associated with one topic in the pub-sub system. Establishing a social relation between two users translates into subscribing to each other’s topic. Publishing information in the social network translates into publishing information in the pub-sub overlay.

In this context, a P2P OSN should satisfy the following requirements:

Network scalability. A P2P OSN system must link millions of users without any noticeable loss of performance.

Tailored topology. OSN users may have hundreds or thousands of friends. Maintaining active connections with every one of them is expensive, since each reliable connection incurs the cost of periodic messages, or of maintaining connection state (e.g., in TCP). Yet, users with only few friends risk remaining unconnected in a dynamic environment. A P2P OSN system must therefore tailor the number of connections for each node.

Connectivity. Updates from one user must reach all her friends. We assume that a user A 's updates reach user B if there is a path in the overlay network between A and B . A P2P OSN system must ensure with high probability that all user friends remain connected.

Cost-effective update propagation. Users profiles are often updated, and most of these updates represent small messages. A P2P OSN must therefore allow (i) the propagation of updates through a small number of edges; and (ii) the aggregation of different small messages into bigger composite messages, amortizing encapsulation costs.

Data Availability. Users may join or leave the network at any time, but their profiles must remain available from other online nodes. A P2P OSN must thus provide mechanisms to ensure high availability and topic-connectivity, even for unpopular users and under high churn rates. Topic-connectivity measures the fraction of users that can be reached by all their friends in the P2P overlay.

This paper makes the following main contributions:

- *A system model.* We map the OSN onto a pub-sub system where each user, represented by one P2P node, has one topic to publish on. We must distribute updates for this topic to all friends of this user, who express their friendship by subscribing to this topic. This allows one to build all required OSN functionality.
- *A system design.* Implementing this pub-sub system in a naive way, where pub-sub links are mapped onto the friendship links, is inefficient: popular topics will have too many links, while unpopular ones will not have enough. For avoiding this, we adapt the SpiderCast protocol [7], which exploits the fact that OSNs have a high clustering coefficient [19].
- *An overlay-friendly replication scheme.* The overlay built in this way keeps, with high probability, a user's online friends in the same connected component. Hence, ensuring accessibility becomes equivalent with ensuring that, at any moment in time, we have at least one replica of each user profile within the connected component to which her friends belong.
- *Detailed scalability evaluations.* We make a detailed characterization of the behavior of such an overlay under large scale settings involving up to ~ 1.3 million nodes, ~ 20 million edges, and under high churn rates, which in turn confirms our scalability claims.

OSN	Pub-Sub	P2P
user	publisher of her profile & subscriber to her friends' profiles	node
user profile	topic	locally stored data
user friends	user interests	node neighbors
social link between two users	two (logic) pub-sub links	P2P link between two nodes

Table 1. Mapping OSN elements to our system model elements

2. Design Overview

2.1 System Model

In our design, each user runs a dedicated P2P application which stores all her social data locally. To advertise her profile updates to her friends, we organize the system as a decentralized topic-based pub-sub system. Each user has one topic to publish on, represented by her profile. This allows other users who have expressed their friendship to subscribe to this topic and receive all profile updates. A friendship link between two users also means that they have an interest in each other. Table 1 highlights how OSN concepts are mapped onto the pub-sub system, and onto the underlying P2P system.

Modeling an OSN as a pub-sub system allows one to build all key functionalities of OSNs:

Social profiles, friend lists, social links. Users can enrich their social network by subscribing to other users profiles. Establishing a relationship between two users creates two symmetric pub-sub subscriptions. The corresponding links in the P2P system form a *social overlay* along which profile updates are propagated.

Update propagation. In a topic-based pub-sub system, messages are disseminated through logical channels (i.e., topics). The pub-sub infrastructure takes care of message distribution to all nodes interested in these topics. Hence, the pub-sub infrastructure routes a user's updates to all her friends who subscribed to her topic.

Search. A pub-sub system does not provide immediate support for searching for new friends. However, the P2P overlay it creates can be used to implement this functionality, for example using an A* search algorithm through the overlay. The high clustering coefficient and low diameter of OSNs implies that most new friends were already known as friends of friends.

2.2 Architecting the Social Overlay

A naive implementation could map the P2P overlay links onto the friendship links. However, such an organization is ineffective: high-degree nodes would have too many links (incurring a high maintenance cost), while low-degree ones would have not enough links (and risk to remain isolated in a dynamic environment). We address this issue using the SpiderCast protocol [7]. SpiderCast exploits the high clustering coefficient and short path lengths of a small world

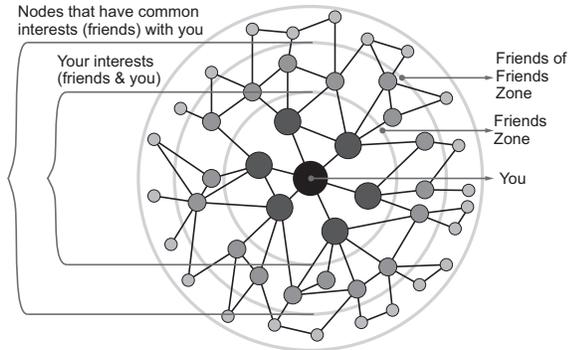


Figure 1. Social graph – user local view

network [19] to reduce the number of connections of the overlay while allowing data to propagate efficiently from each node to its subscribers (even those to which it does not have an active connection).

We define a node’s *local view* as the information it stores locally about the social network. This local view is composed of the IDs of all nodes with whom it shares interests: its own direct friends, the friends of its friends, and the social relationships between them (see Figure 1).

The P2P overlay is built as follows. First, each node gathers information about its 1-hop social neighborhood: its own friends as well as the friends of its friends, and the friendship links between them. In a second phase it iteratively selects nodes within this neighborhood and links to them, until each of its own OSN friends can be reached through a path in the overlay. This allows to drastically reduce the number of links per node: thanks to the high clustering coefficient, most friends are also known as the friend of a friend.

We say that a node’s interest is covered if it is represented in the subscription of at least one of its neighbors in the overlay [7]. In practice this means that it can receive all updates from its friends in roughly two hops (Section 3.1.2.). The rationale is to build the overlay such that all interests of all nodes are covered. To cover its interests, each node keeps track of the interests covered by its current P2P links, and iteratively selects from its local view nodes that cover yet-uncovered interests and links to them. The selection is done by employing one of the following selection strategies: *greedy strategy* - selects the links that maximize the number of covered interests, or *random strategy* - randomly selects a node that increases the number of covered interests. Unlike SpiderCast, we do not combine the overlays obtained with each selection strategy.

SpiderCast exhibits great potential to building highly scalable P2P OSNs and represents a good starting point for their design. However, there remain key issues that still need to be addressed. In particular, we need to guarantee high availability of a user’s data while this user is disconnected. We address this important question in the next section.

Network	No. nodes	No. edges	Avg. Deg.	Max. Deg.	Clust. Coef.
Collab. Network	18,722	198,080	21.1	504	0.63
Filtered Facebook	1,288,842	19,311,084	29.96	3593	0.21

Table 2. Datasets characteristics

2.3 Towards High Availability and Accessibility

An important aspect of P2P systems is churn: at any moment of time, a significant fraction of users must be assumed to be offline. In the case of a P2P OSN where each user stores its own profile, this means that two friends must be online simultaneously for updates in one user’s profile to become visible from the other user. Obviously, such a limitation would be unacceptable to many users.

One key feature of our P2P overlay is high topic connectivity. If we assume that all topics are connected, then ensuring profile data availability and accessibility is equivalent to ensuring that, at any moment in time, at least one replica of each user profile is present within the social overlays to which its friends belong. An effective measure to ensure data availability is to proactively replicate a user’s profile at a number of nodes in the overlay.

A simple approach would be to replicate a user’s profile in nodes owned by its friends, or the friends of its friends. However, unpopular users may have very few friends to whom they can replicate their profile. We instead propose to fix the number of online replicas that all user profiles should have. Nodes holding a replica of a given profile ping each other periodically to detect nodes’ departures. Whenever one node has disconnected from the system, the oldest remaining node is in charge of creating a new replica and advertising its location to the other nodes holding the same replica. This new replica is preferably chosen at a friend of the concerned user. When this is not possible, the node holding this new replica is chosen randomly in the system. As long as a node holds a replica, it also subscribes to the replica’s topic. This will enable the overlay construction protocol to connect nodes holding replicas to nodes interested in this information. We expect that simple cryptographic techniques will allow to avoid disclosing personal data to the nodes chosen to hold replicas.

3. Evaluation

We evaluate our P2P OSN design using trace-driven simulations. Our simulator allows us to simulate large number of nodes, and replay the behavior of the entire P2P system. We evaluate our design using two real-world datasets presented in Table 2: (i) a dataset from Facebook (F.Fb.) obtained by filtering out nodes with degree 1 from one of the datasets released by [20]¹; and (ii) a collaboration network (C.N.) dataset [13].

¹We filtered out all nodes with degree 1 because they were clearly over-represented in the dataset. We speculate that this is an artifact of the sampling method used to collect this dataset.

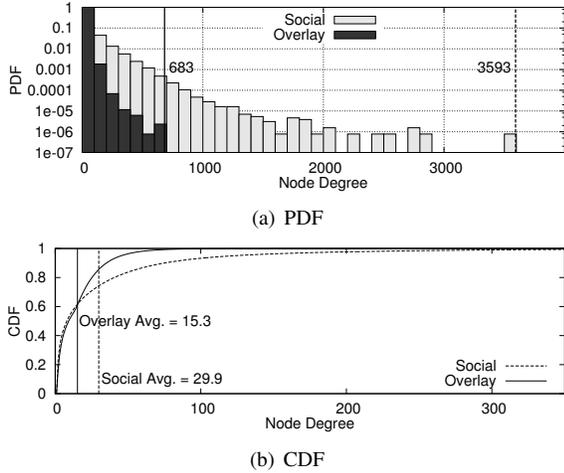


Figure 2. Node degree distribution in the F.Fb/random overlay

Network	Cov. Strategy	Avg. Deg.	Connect.
C.N.	greedy	6.31	0.9867
C.N.	random	9.45	0.9652
F.Fb.	greedy	16.0	0.9842
F.Fb.	random	15.37	0.9997

Table 3. Node degrees and topic connectivity

We evaluate our system in static settings (where all nodes remain connected throughout the entire simulation) and dynamic settings (with churn). In dynamic settings each live node disconnects after a time-to-live (TTL) which is randomly generated. When the TTL expires, another node, selected uniformly at random, is woken up. The simulation therefore maintains a constant fraction of online nodes (set to 30%). In all simulations, we assume that a communication event takes hundreds of milliseconds.

3.1 Scalability in Static Settings

Figure 2 shows the distribution of node degrees in the P2P overlay and the original social graph, in the case of a system with no churn. We can see that the node degrees are significantly reduced, with the average node degree dropping from 29.96 to 15.37 and the maximum node degree dropping from 3593 to 683. Similar results are obtained for the Collaborative network overlay (we omit the figures due to lack of space).

Despite this drastic reduction of the number of links per node, the overlay achieves very good topic connectivity. Table 3 shows that connectivity remains greater than 96% in all cases, up to 99.97% for the Facebook overlay using a random link selection strategy. This means that the reduction of number of links in the overlay does not introduce significant disruption for the good propagation of profile updates. Note that we do not use replication in this section.

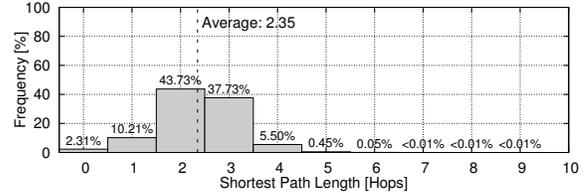
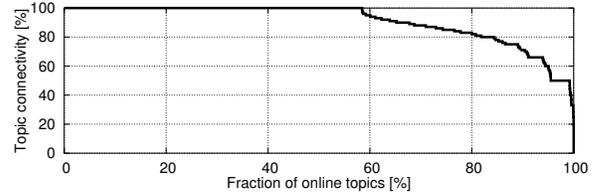


Figure 3. Distribution of shortest paths in the F.Fb/random overlay



(a) Online topics connectivity (F.Fb/random overlay)

Network	2.5 min	5 min	7.5 min	10 min	Final
C.N.	0.972728	0.977735	0.979386	0.98157	0.986721
F.Fb.	0.989449	0.991767	0.992761	0.993272	0.999706

(b) Convergence of topic-connectivity

Figure 4. Connectivity in dynamic settings

The path length, i.e., number of hops between two nodes, is a good metric for the cost of distributing profile updates. In a social network, most communications happen between friends. Therefore, we measure the shortest paths between pairs of friends. We sampled 2.5 million random friendship links out of 20 million. For each pair of friends we measured the shortest path length between them in the P2P overlay. Figure 3 shows the distribution of these shortest path lengths in the Facebook/random case. The average shortest path length is 2.35, with a median value of 2. Only 0.06% pairs of friend nodes are more than 5 hops away from each other. The overlay therefore keeps friend nodes close to each other, which allows for efficient communication between friends.

3.2 Connectivity in Dynamic Settings

We now move to a system where nodes continuously join and leave the system. In this dynamic context where only 30% of nodes are online at any moment in time, keeping topics connected is much harder than in a static system which has ample opportunity to converge. Unless otherwise specified, the TTL values are randomly generated within an interval of $[0, 300]$ seconds.

We first measure the connectivity level of online topics. A topic is considered to be online if there is at least one online node interested in it (regardless of the current availability of the user whose profile is propagated using this topic). In this evaluation, we exploit the natural replication created by the fact that the friends of an offline node may be online and propagate the information. As such, we assume that

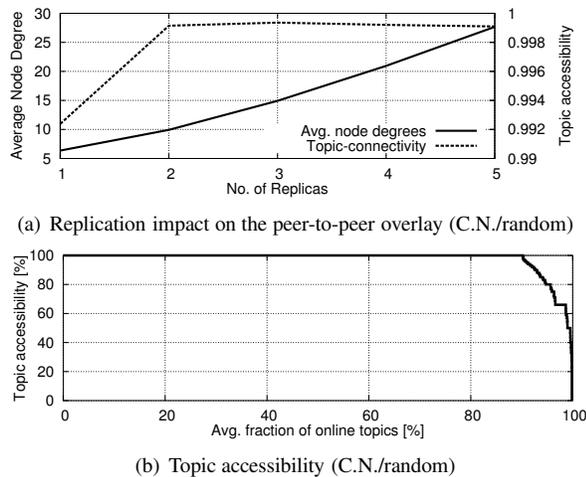


Figure 5. Replication Performances

nodes locally cache their friends most recent updates (this allows efficient implementation of OSNs key features, such as newsfeeds). However, we do not add any artificial replication yet.

Figure 4(a) shows that in these difficult settings, about 60% of the online topics are fully connected. The other 40% of online topics are not fully connected, but half of them still link more than 80% of their respective recipients. The average node degree is only 4.18. We obtain slightly better results for the C.N. overlay thanks to its higher clustering coefficient: 80% of online topics are fully connected, with an average node degree of 4.69 (we again omit the figure due to lack of space).

Note that these results are obtained at snapshots in time, and may capture transient states, when some online topics’ connectivity has not yet converged. We therefore analyze the evolution of topic connectivity during the formation of the overlay in static settings. Table 4(b) shows that the topic-connectivity is converging towards a state of connectivity for almost all topics.

3.3 Replication

Placing replicas at random locations in the overlay produces two simultaneous effects: it increases data accessibility, at the cost of forcing the replica node to establish a new P2P link in case it didn’t have a suitable one already; while placing them at friends has no such effect. Figure 5(a) shows these two metrics when we vary the number of replicas per user profile in static settings and with replicas placed at random nodes. The average node degree grows roughly linearly with the number of replicas. Note though that this is the upper bound for the average node degree, since our replication scheme first attempts to place replicas at friend nodes. The first two replicas significantly improve topic connectivity, but additional replicas do not bring any noticeable benefit.

We now evaluate data availability when using three replicas per user profile in dynamic settings. Data is available if at a given time the user or any of her replicas is online, regardless of pub-sub connectivity. We do not consider replica failures or restrictions on nodes storage capacity. For all simulations and all snapshots, we observed 100% data availability for all topics with 3 replicas in place.

Having data available in the network does not necessarily mean it is reachable, as the node holding data may be disconnected. Figure 5(b) plots the topic accessibility, namely the ability of interested nodes to locate and fetch the data they are looking for. Replication clearly provides a significant benefit: 90% of topics are fully accessible, and the rest is generally more than 50% accessible.

4. Related Work

P2P social networks. The need and opportunities to implement P2P social networks is well documented [5, 21], and has triggered a number of research efforts in this direction [6, 8, 9, 11, 16]. From the open source community, Diaspora² is the most notable example. Not all systems have the same goal: some systems aim to replicate the same functionalities as those found in centralized systems while others exploit decentralization to implement new functionalities.

Gossple meaningfully associates and exchanges information between users, without requiring any previous relations [4]. The focus is on finding like-minded people while preserving users’ anonymity. However, while Gossple connects users that exhibit a certain level of similarity, our system connects them based on pre-existing relations.

PeerSoN [6], LifeSocial.KOM [11] and Pokurt [17] aim to overcome the OSN limitations by focusing on user privacy consolidation by means of encryption, access control and decentralization, while preserving OSN features. To provide reliable and replicated data storage, LifeSocial.KOM relies on PAST, while Pokurt exploits users’ geographical location and online time period. However, these proposals have not been evaluated in large-scale settings so it is difficult to anticipate their performance if they were deployed at a scale of millions of users.

Recent proposals rely on gossiping protocols to build and maintain an overlay network whose links mirror the real social relations [9, 16]. These systems are similar to ours, but do not trim the links to reduce the average and maximum node degrees. As we have seen, mapping the overlay links one-to-one on the social relationships is likely to create load imbalance and overwhelm the most popular nodes.

Few proposals explicitly address scalability issues in P2P OSNs. eXO focuses on query processing, assuming that a P2P overlay has already been built [15]. It therefore does not address the scalability issues at the overlay level. Additionally, issues such as profiles updates dissemination and content availability are left as future work. Google’s PubSub-

² <http://joindiaspora.com/>

Hubbub decentralized pub-sub protocol has been proposed as a solution to building scalable P2P OSN [12]. The protocol assumes a large number of subscribers, many publishers, and few hubs that store the social graph of the publishers and filter appropriate subscribers to push the content. However, scalability is not evaluated, relying only on scalability claims about the Google’s public hub.

Pub-Sub Systems. A number of scalable P2P pub-sub protocols have been proposed [7, 10, 18]. SpiderCast relies on the assumption that if, for each topic of interest, nodes connect to n random nodes interested in the same topic, the topic-specific overlay will be globally connected. This works well in practice in the presence of churn provided that $n \geq 3$. In our case, even for $n = 1$, topic connectivity remain largely ensured thanks to the high clustering coefficient in OSNs and the use of replication techniques.

Systems such as Magnet and Vitis bound the number of connections per node [10, 18]. Vitis relies on a gossip-based peer sampling service, while Magnet uses the Oscar overlay which exhibits small world properties and arranges nodes in a ring [10]. Although these systems work well for general-purpose pub-sub application, their behavior in the specific context of OSNs remains unclear.

5. Conclusions

Scalability and robustness are two often-overlooked properties in P2P online social networks. We demonstrated that the use of relatively simple techniques allows to maintain the P2P node degrees under control while ensuring good data availability and accessibility properties, even in the presence of heavy churn.

This paper deliberately ignores the issue of user privacy. Many other research efforts have focused specifically on this topic. We believe security techniques are largely orthogonal to our own work but we leave the integration of both concerns for future work.

Acknowledgments

We would like to thank the anonymous reviewers for their feedback and suggestions. We thank Stefan Bucur and Cristian Zamfir for their help and advice. Alexandra Olteanu was partially supported by the grant “Reconcile: Robust Online Credibility Evaluation of Web Content” from Switzerland through the Swiss Contribution to the enlarged European Union.

References

- [1] M. Backes, M. Maffei, and K. Pecina. A security API for distributed social networks. In *Proc. NDSS*, 2011.
- [2] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, and S. Vigna. Four degrees of separation. *CoRR*, abs/1111.4570, 2011.
- [3] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin. Persona: an online social network with user-defined privacy. In *Proc. SIGCOMM*, 2009.
- [4] M. Bertier, D. Frey, R. Guerraoui, A.-M. Kermarrec, and V. Leroy. The Gossple anonymous social network. In *Proc. Middleware*, 2010.
- [5] S. Buchegger and A. Datta. A case for P2P infrastructure for social networks - opportunities and challenges. In *Proc. WONS*, 2009.
- [6] S. Buchegger, D. Schiöberg, L. H. Vu, and A. Datta. PeerSoN: P2P social networking - early experiences and insights. In *Proc. SNS*, 2009.
- [7] G. Chockler, R. Melamed, Y. Tock, and R. Vitenberg. SpiderCast: a scalable interest-aware overlay for topic-based pub/sub communication. In *Proc. DEBS*, 2007.
- [8] L. Cuttillo, R. Molva, and T. Strufe. Safebook: A privacy-preserving online social network leveraging on real-life trust. *Communications Magazine, IEEE*, 47(12):94–101, Dec. 2009.
- [9] A. Datta and R. Sharma. GoDisco: selective gossip based dissemination of information in social community based overlays. In *Proc. ICDCN*, 2011.
- [10] S. Girdzijauskas, G. Chockler, Y. Vigfusson, Y. Tock, and R. Melamed. Magnet: practical subscription clustering for internet-scale publish/subscribe. In *Proc. DEBS*, 2010.
- [11] K. Graffi, C. Gross, P. Mukherjee, A. Kovacevic, and R. Steinmetz. LifeSocial.KOM: A P2P-based platform for secure online social networks. In *Proc. P2P*, 2010.
- [12] P. Kapanipathi, J. Anaya, A. Sheth, B. Slatkin, and A. Pasasant. Privacy-aware and scalable content dissemination in distributed social networks. In *Proc. ISWC*, 2011.
- [13] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proc. KDD*, 2005.
- [14] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1), Mar. 2007.
- [15] A. Loupasakis, N. Ntarmos, and P. Triantafillou. eXO: Decentralized autonomous scalable social networking. In *Proc. CIDR*, 2011.
- [16] G. Mega, A. Montresor, and G. P. Picco. Efficient dissemination in decentralized social networks. In *Proc. P2P*, 2011.
- [17] R. Narendula, T. G. Papaioannou, and K. Aberer. Privacy-aware and highly-available OSN profiles. In *Proc. WETICE*, 2010.
- [18] F. Rahimian, S. Girdzijauskas, A. H. Payberah, and S. Haridi. Vitis: A gossip-based hybrid overlay for internet-scale publish/subscribe enabling rendezvous routing in unstructured overlay networks. In *Proc. IPDPS*, 2011.
- [19] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow. The anatomy of the Facebook social graph. *CoRR*, abs/1111.4503, 2011.
- [20] C. Wilson, B. Boe, A. Sala, K. P. Puttaswamy, and B. Y. Zhao. User interactions in social networks and their implications. In *Proc. EuroSys*, 2009.
- [21] C.-M. A. Yeung, I. Llicardi, K. Lu, O. Seneviratne, and T. Berners-Lee. Decentralization: The future of online social networking. In *Proc. MSNWS*, 2009.