

# Epidemic Diffusion of Social Updates in Dunbar-Based DOSN

Marco Conti<sup>2</sup>, Andrea De Salve<sup>1</sup>, Barbara Guidi<sup>1,2</sup>, and Laura Ricci<sup>1</sup>

<sup>1</sup> University of Pisa - Department of Computer Science,  
Largo B. Pontecorvo, 56127, Pisa, Italy

<sup>2</sup> IIT-CNR, via G. Moruzzi, 1 56124 Pisa, Italy  
`{desalve,guidi,ricci}@di.unipi.it`  
`m.conti@iit.cnr.it`

**Abstract.** Distributed Online Social Networks (DOSNs) do not rely on a central repository for storing social data so that the users can keep control of their private data and do not depend on the social network provider. The ego network, i.e. the network made up of an individual, the ego, along with all the social ties she has with other people, the alters, may be exploited to define distributed social overlays and dissemination protocols. In this paper we propose a new epidemic protocol able to spread social updates in Dunbar-based DOSN overlays where the links between nodes are defined by considering the social interactions between users. Our approach is based on the notion of Weighted Ego Betweenness Centrality (WEBC) which is an egocentric social measure approximating the Betweenness Centrality. The computation of the WEBC exploits a weighted graph where the weights correspond to the tie strengths between the users so that nodes having a higher number of interactions are characterized by a higher value of the WEBC. A set of experimental results proving the effectiveness of our approach is presented.

**Keywords:** DOSN, P2P, Information Diffusion, Dunbar.

## 1 Introduction

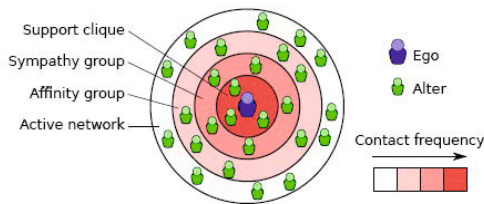
In the last few years, Online Social Networks (OSNs) have become one of the most popular Internet services and they have changed the way of how people interact with each other. Currently popular OSNs are based on centralized servers which store all the user's information. In this way, users give up the control of their own data and their data is scattered over the Internet in different OSN providers. Centralized social networking services present several problems that include both technical and social issues that emerge as a consequence of the centralized management of the services.

A current trend for developing OSN services is towards the decentralization of the OSN infrastructure. A Distributed Online Social Network (DOSN) [6] is an Online Social Network implemented over a distributed information management

platform, such as a network of trusted servers or P2P systems. The decentralization of the existing functionalities of Online Social Networks requires specific approaches for providing robustness against churn, distributing storage of data, propagating updates, defining an overlay topology and a protocol enabling searching and addressing, etc.

The inherent nature of how people connect with each other in a OSN makes P2P architectures suitable for building DOSNs. Our proposal takes into account the friendship relations and the interactions between users to model the system according to a simple social structure called ego network [16]. Ego networks are social networks made up of an individual (called *ego*) along with all the social ties she has with other people (called *alters*). The key property of this model is the *tie strength*, represented as the social distance between the ego and the alter involved in a relationship.

It has been shown that in (offline) ego networks there are a series of *circles* of alters arranged in a hierarchical inclusive sequence based on an increasing level of intimacy [18]. Dunbar et al. identify four layers [18]: "*support clique*", "*sympathy group*", "*affinity group*" and "*active network*" with the average size of 5, 12, 35, 150 respectively (Fig. 1). Each subset includes all the relationships of the previous circles along with an additional set of social links with a weaker level of intimacy. The last set contains simple acquaintances, with a relatively weak relationship with the ego while the first set contains only alters with a very strong relationship with the ego.



**Fig. 1.** Structuring ego networks: the Dunbar Approach

This paper presents an epidemic algorithm for the diffusion of social updates on DOSNs. A social update is defined as any social content that users share with their friends (profile information, wall postings, pictures, etc.). Users generate a huge amount of social update inside a social network which should be disseminated to their direct friends or to a larger extent, depending on the kind of the social update. Any social update should be disseminated with a low cost in term of time and messages. Even if the definition of the P2P social overlay is not the main focus of this paper, it is important to describe the model of the overlay which must support the spreading of the contents. In our proposal, the P2P overlay is defined by a one-to-one mapping between the social links

connecting the ego to its Dunbar alters and the P2P overlay links. In this way, each node of the P2P network needs to support only active overlay connections (on average, at most 150 active relations per user [2]). In the following, this overlay will be referred as Dunbar-based overlay. The diffusion of the social updates on the Dunbar based overlay is performed by an epidemic algorithm.

Most existing DOSNs flood the social updates on the social overlay and this may result in several duplicate notifications. We introduce a novel centrality index, the Weighted Ego Betweenness Centrality (WEBC), which extends the classical notion of Ego Betweenness Centrality (EBC) [7] by considering the weights paired with the edges of the social overlay. The knowledge of the WEBC of the nodes in the ego network, enables the definition of a heuristics for guiding the epidemic spreading which reduces the number of duplicate messages and enables the diffusion of social updates on paths corresponding to higher tie strengths.

The rest of the paper is structured as follows. We discuss the related work in Section 2. Section 3 introduces some basic concepts of the OSN and define the social overlay of the DOSN. The WEBC is defined in Section 4 and the epidemic algorithm based on it is shown in Section 5. In Section 6 we present the evaluation of our proposal. Finally, Section 7 reports the conclusions and discusses future works.

## 2 Related Work

Several DOSNs are designed to address privacy in OSNs either through cryptography, architectural modifications and decentralization. Diaspora [1] is a really used social network in which users' profiles are hosted on servers that are administrated by individual users, without support for encryption. LifeSocial.KOM [8] is a P2P platform for secure online social networks which provides the functionality of common online social networks in a totally distributed and secure manner. LifeSocial.KOM is designed to create a modular plugin-architecture to assure extensibility and the plugins are hosted on a general platform for P2P application. PeerSoN [4] is a distributed infrastructure for social networks that provides encryption and direct data exchange. PeerSoN has a two-tier architecture, where one tier serves as a look-up service and the second tier consists of the peers that contains user data, such as user profiles. This approach pays attention to security and privacy concerns. Safebook [5] is a decentralized and privacy preserving DOSN which is composed of three major different components, the TIS (Trusted Identification Service), Matryoshkas which are specialized overlays encompassing each user, and a P2P location substrate based on Kademlia. In [17], a recent approach based on a Super-Peer architecture is shown. The system uses Super-Peers, which are nodes with a higher degree of bandwidth, storage space and higher availability, to manage DOSNs.

The social update diffusion problem has been studied in Social Butterfly [12] and in SocialCDN [13], which achieve the social update dissemination utilizing social caching techniques.

Another approach similar to ours is proposed in [14]. The authors propose a gossip protocol for data dissemination in DOSN in which each node requires knowledge of its set of friends and friends of friends. They utilize a vertex anticentrality selection heuristics to assign to each neighbour of the producer a selection probability which is inversely proportional to the number of their common neighbours. In addition, they improve the diffusion process with histories and a fragmentation heuristics that exploits the number of connected components to select where to send the gossip messages. However, our approach differs from theirs for several reasons: *i)* we use a different selection heuristics that takes into account the centrality of the node and the strength of the relationship; *ii)* we leverage a distributed algorithm for community detection; *iii)* we propose a different dissemination method in which nodes leverage only on the knowledge of the nodes directly connected to them.

### 3 Defining Distributed Social Overlays

An Online Social Network (OSN) may be formally described by an undirected social graph  $G = (V, E)$ , where  $V$  represents the set of users and  $E$  the friendship relations between them. The *ego network* of a user represents a structure built around the ego which contains her direct friends, known as *alters* and may also include information about the direct connections between the alters. Formally, each vertex  $u \in V$  can be seen as an *ego* and  $EN(u) = (V_u, E_u)$  is the ego network of  $u$  where  $V_u = \{u\} \cup \{v \in V | (u, v) \in E\}$ , and  $E_u = \{(a, b) \in E | a = u \vee b = u \vee \{a, b\} \subseteq V_u\}$ .  $N(u) = V_u - \{u\}$  is the set of adjacent nodes of  $u$ . The alters in an ego network may be organized according to the *tie strength* between the ego and each alter. The tie strength defines the distance between them by exploiting factors of the OSN and may be represented by a weight paired with each edge of the ego network. As suggested by Dunbar, we include in the resulting ego network at most 150 active contacts and we will refer to it as Dunbar based ego network. We believe that this definition could be easily extended to consider the maximum number of active contacts an input parameter of our model.

The definition of the tie strength is currently an open problem and several alternatives have been presented in the literature. In [3] the tie strength is computed by considering a set of relational variables, like the number of likes, posts, comments, tags on the same picture and so on in a temporal window. Sala et al.[19] exploit an indirected weighted graph to describe the social relations, because they observe that most interactions on Facebook are reciprocated so that the notion of tie strength is symmetric. In other proposals (such as [10]) the notion of tie strength is asymmetric and a directed graph is exploited to describe social relations. In the following we will focus on undirected graph and we will suppose that the tie strength is computed as a function of the users' interactions occurred in both directions.

Let us now briefly discuss the issue of defining a distributed overlay for DOSN. In a DOSN, each user is mapped to a node of the distributed system and a

social overlay connecting the nodes is defined. A direct mapping of the social graph onto the distributed social overlay is feasible, but this solution presents several drawbacks. As of September 2013 the average number of Facebook friends reached 338 among adults aged between 18-29, while it is about 200 for older adults. This implies that each node of the distributed system should maintain a large amount of connections with nodes paired with friends even if a large subset of them may be under-used.

We exploit the Dunbar's approach to limit the number of connections of each node. A link between a node  $A$  and a node  $B$  is defined in the social overlay iff  $B$  is in the Dunbar-based ego network of  $A$ . Each node  $n$  maintains a view which contains the descriptors of all the nodes which are directly connected to it. The number of nodes contained in a view is limited by the Dunbar number (150), and each node can directly communicate only with nodes in its view. Each link is weighted and the weight is the tie strength between two nodes. Note that not all the social relations are mapped onto the social overlay. The social relations characterized by low tie strengths are not supported by direct connections on the social overlay, since a few interactions occur on them. These interactions may be supported by on-demand communications.

## 4 Weighted Ego Betweenness Centrality

Ego Betweenness Centrality (EBC) is defined as the Betweenness Centrality (BC) of a node restricted to the nodes and the links in its ego network. The computation of the EBC for an undirected graph has been proposed in [7]. [9] shows a distributed approach to compute the EBC on undirected and directed graph exploiting two alternative distributed protocols, a broadcast and a gossip-based protocol, for disseminating the ego networks updates to the neighbours of a node.

BC and EBC are topology-based metrics, so that value for a node  $n$  depends on the position and on the connections of  $n$  in the graph. Consider, for instance, Fig. 2(a), representing the ego network of node  $E$  and let us neglect, for the moment, the weights paired with the links of the graph. The value of the EBC for nodes  $E$ ,  $D$  and  $B$  is exactly the same,  $\frac{1}{3}$ , since a single shortest path out of 3 different shortest paths between  $A$  and  $C$  passes through each of them. Therefore, the value of the EBC does not permit us to decide, for instance, which node between them is the best choice for propagating a social update from  $A$  to  $C$ . This is due to the fact that the EBC considers only structural properties of the graph.

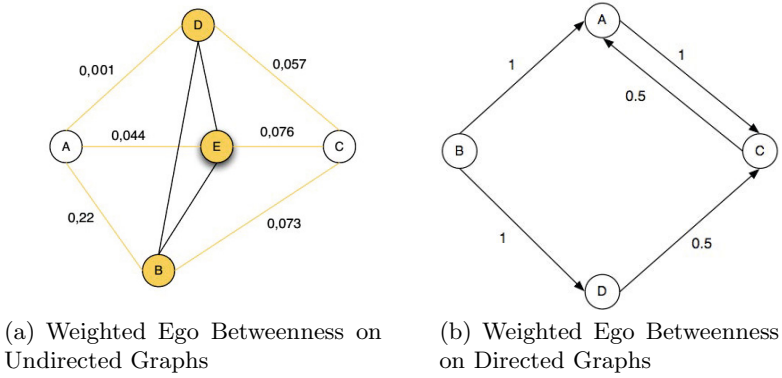
We introduce a new notion of EBC for weighted graphs, the *Weighted Ego Betweenness Centrality (WEBC)*, which discriminates the shortest paths crossing the ego according to the weights of the edges on those paths. The computation of the WEBC requires the adjacency matrix  $A_{i,j}^n$  associated with the ego network of node  $n$ ,  $ego(n)$ , and the weights matrix  $W_{i,j}^n$  which contains the weights paired with the edges of  $ego(n)$ , (note that a weight may also be equal to 0). The elements of the matrices are accessed through the nodes' identifiers  $i$  and  $j$ . Given two nodes  $i$  and  $j$  belonging to  $ego(n)$  such that  $A_{i,j}^n = 0$ , let

$$Path_{i,j}(n) = \sum_{k \in ego(n), k \neq i,j} A_{i,k}^n * A_{k,j}^n * W_{i,k}^n * W_{k,j}^n$$

be the sum of the weights of all the 2-hops paths between nodes  $i$  and  $j$  including only edges in  $ego(n)$ . The WEBC of a node  $n$  for an undirected graph is defined as follows:

$$WEBC(n) = \sum_{i,j \in ego(n), A_{i,j}^n=0, j>i} \frac{W_{i,n}^n * W_{n,j}^n}{Path_{i,j}(n)} \quad (1)$$

where all the nodes  $i \neq j$  such that there is not a directed link between them ( $A_{i,j}^n = 0$ ) in the ego network of  $n$  are considered in the summation. The path between the nodes  $i \neq j$  crossing  $n$  has weight  $W_{i,n}^n * W_{n,j}^n$ .



**Fig. 2.** Weighted Ego Betweenness on Undirected/Directed Graphs

We have previously seen that nodes  $E$ ,  $D$  and  $B$  in Fig. 2(a) are indistinguishable in term of EBC. The values of the WEBC for nodes  $E$ ,  $D$ , and  $B$  obtained by applying Eq. (1) are the following ones:  $WEBC(D)=0.0029$ ,  $WEBC(E)=0.17$  and  $WEBC(B)=0.82$ . The WEBC highlights that the path connecting  $A$  and  $C$  and crossing  $B$  has a higher weight with respect the other ones.

In a Dunbar-based overlay, where the weights paired with the links of the overlay represent the tie strength between two users, we can exploit WEBC to distinguish among different paths the most important one in term of the number of the interactions between the nodes on that path.

In some scenarios, it is useful to evaluate  $WEBC(n, a)$ , the WEBC of a node  $n$  with respect to a particular alter  $a$ . We define the  $WEBC(n, a)$  as follows:

$$WEBC(n, a) = \sum_{j \in ego(n), A_{a,j}^n=0} \frac{W_{a,n}^n * W_{n,j}^n}{Path_{a,j}(n)} \quad (2)$$

In the next section, we will see that  $WEBC(n, a)$  can be exploited to evaluate the capability of node  $n$  to connect the alter  $a$  to nodes in the ego network of  $n$  which are not directly connected to  $a$ . An epidemic algorithm can exploit this information to define a heuristics for the neighbour selection when propagating a social update.

Even if in the next section we will apply WEBC on undirected graphs, it is important to notice that the computation of the WEBC on a directed graph requires some modifications to the previous definition. As a matter of fact, when we consider a direct graph, a node  $n$  may belong to the ego network of a node  $m$ , while the other way round may be not true. When computing the WEBC of a node  $n$  on a directed graph, a possible solution is to consider all the *shortest oriented paths* passing through  $n$ . In the graph shown in Fig. 2(b), the WEBC of  $A$  considers the *weighted oriented path* between  $B$ , and  $C$  crossing  $A$  with respect to all the weighted oriented paths linking  $B$  to  $C$ , resulting in  $WEBC(A)=0.6$ .

## 5 WEBC Based Information Diffusion in DOSN

In a OSN, like Facebook, a social update produced by user  $u$  should be sent to users which are  $k$ -hops distant from  $u$  in the social graph. The extent of the diffusion depends on the kind of the social update and on the privacy settings of the users. For instance, in Facebook, each post published by a user on her own wall should be sent to her 1-hop friends, i.e. to all the users in her ego network. In other scenarios, like the comment to a post or a photo of a friend the information may be transmitted to 2-hops distant users *FriendOfFriend* (FoF). Some particular scenarios require to transmit the update even to 3-hops-away friends *FriendOfFriendOfFriend* (FoFoF).

In a DOSN, the social updates are transmitted to the  $k$ -hops neighbours through the links of the social overlay. We consider a Dunbar-based overlay, where each edge is paired with a weight representing the tie strength between the nodes it connects. The diffusion of the social update is done by an *epidemic algorithm* that exploits the WEBC selection heuristics and the community structures in order to guide spread of the social update towards more favoured regions. When a node  $v$  receives a social update it passes it to the neighbour  $n$  in its ego network maximizing  $WEBC(n, v)$ . This heuristic allows both to choose the neighbours able to connect  $v$  to nodes not belonging to its ego network and to speed up the spreading on highly weighted paths. Note that this heuristic can be implemented by exploiting only local information.

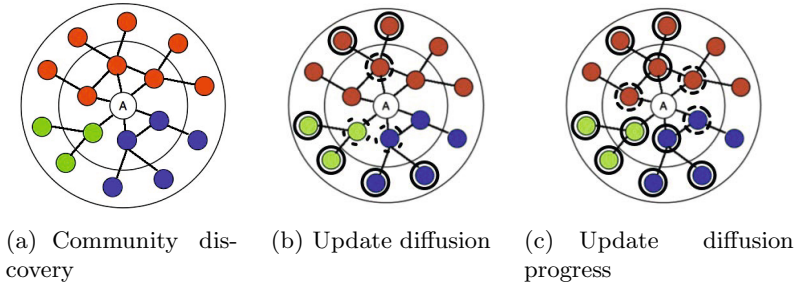
The differentiation of paths based on their weights implies that nodes belonging to highly weighted paths will receive the update earlier. This agrees with the recent design choice of current social networks, where the importance of social updates is classified according to the tie strength with the node generating the update (for instance each node in Facebook classifies the updates it receives according to this politics in order to present the updates on different news feeds, on the basis of their importance).

## 5.1 Our Algorithm

Let us suppose that a node  $X$  generates a social update. We focus on the case where the update has to be spread 2-hops away with respect to  $X$  (to its FoF), but our approach is valid also for in the general case of  $k$ -hops diffusion.

Our algorithm is organized into two phases: *i*) Communities discovery phase and *ii*) Diffusion of social content into each community phase.

The phase of community discovery allows to partition the  $i$ -hops,  $i \in 1..2$ , neighbours of  $X$  into a set of communities such that, if we exclude  $X$  from the graph, each pair of nodes within a community are connected by a path, while nodes belonging to different communities are not connected. Our community discovery algorithm is based on that proposed in [11]. Fig. 3(a) shows the node  $X$  together with its two hops social contacts. These nodes are shown into two concentric circles, according to their social distance from  $X$ . The different communities detected by the algorithm are shown with different colours.



**Fig. 3.** Phases of the epidemic algorithm

The WEBC-based epidemic algorithm is executed in the second phase. Let  $N_1(X)$ , respectively  $N_2(X)$ , be the set of nodes which are 1-hop, respectively, 2-hops from  $X$ . The information diffusion distributed algorithm works as follows:

- the producer  $X$  of the update  $o$  starts the information diffusion by choosing a node for each community (i.e. nodes surrounded by dotted line in Fig. 3(b)) and sending  $o$  to it. These nodes are selected on the basis of their WEBC computed with respect to  $X$ .
- if a node  $Y$  in  $N_1(X)$  receives the update  $o$  from another node  $K$  in  $N_1(X)$ , it checks its ego network to see if it contains neighbours in  $N_2(X)$  which are not also neighbours of  $K$  (recall that the ego network of  $Y$  includes also the links connecting neighbours of the ego). If the set of these neighbours is not empty, it sends the update to each node in this set (i.e. nodes surrounded by solid line in Fig. 3(b)). Then  $Y$  chooses one neighbour belonging to  $N_1(X)$  according to the WEBC-based heuristics (i.e. one of the nodes surrounded by dotted line in Fig. 3(c)) and propagates the update to it.
- nodes in  $N_2(X)$  do not propagate the update.



- each node  $n$  maintains the list of updated nodes *UpdatedNodes*, which is initially empty.  $n$  records in this list the nodes which are known to have received the update. Whenever  $n$  sends an update to one of its neighbours, it adds this neighbour to this list.
- if a node receives a social update, but all the neighbours in its ego network belongs to *UpdatedNodes*, it stops the diffusion of the information.

Note that the previous algorithm may be refined in several directions. For instance, in the second step, the check related to the common neighbours in the ego networks of the sender and of the receiver does not guarantee that the message will be propagated only to nodes which has not received it previously. As a matter of fact, each node has only a local and partial view of the nodes of the community, restricted to its ego network. To face this problem, it is possible to pair each update with an *history* recording the nodes which have already received the update, as proposed in [14]. This allows to reduce the number of duplicate updates at the expense of a larger usage of the network bandwidth.

## 6 Experimental Results

We have developed a set of simulations of our system using the P2P Peersim simulator [15], a highly scalable simulator written in java. We have used a Facebook Regional Network<sup>1</sup> dataset to build the Dunbar-based social overlay. The dataset is composed by:

- A *Social Graph*: an undirected graph which defines the whole network structure. An edge corresponds to a social relation between two Facebook users.
- Four *Interaction Graphs*: directed graphs which define the interactions between users within different time windows: *last month*, *last 6 months*, *last year*, *2004-2008*. The Interaction graphs contain an edge for each interaction (Post or Photo Comment) between two users happened in the considered time window. If a user  $j$  has had an interaction with a user  $i$ , a link from  $j$  to  $i$  is created in the Interaction Graph.

Table 1 shows some characteristics of the Social Graph computed in [2].

**Table 1.** Social Graph

# Nodes	3,097,165
# Edges	23,667,394
Average Degree	15.283
Average Clustering Coefficient	0.098
Assortativity	0.048

---

<sup>1</sup> Referred as Anonymous regional network A on <http://current.cs.ucsb.edu/facebook/>.

We have evaluated the EBC and WEBC centrality indexes for several networks randomly extracted from the real dataset. The contact frequencies of the social relations extracted from the Interaction Graphs are used to compute the weights associated with each relation. The nodes are ranked on the basis of the value of their EBC/WEBC. Figure 4(a) reports on the x-axis the rank of the nodes and on the y-axis the value of the centrality indexes on a logarithmic scale for one of the selected networks. We can observe that the WEBC provides a better differentiation of nodes, i.e. it allows to evaluate nodes not only from the point of view of their structural properties, but also from a qualitative point of view. We can observe that nodes with a similar EBC are redistributed by the WEBC on the basis of the contact frequency of the paths.

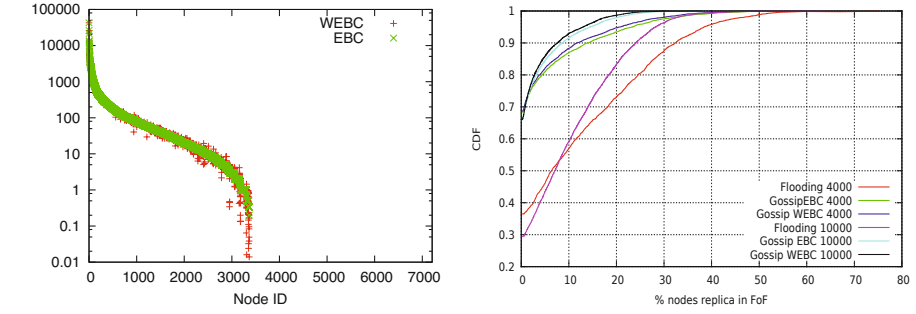
We have evaluated the epidemic diffusion algorithm on a subset of nodes extracted from the original dataset. The dimension of the extracted networks is, respectively, of 4000 and 10000 nodes. Table 2 shows the structural properties of the two extracted networks.

**Table 2.** Network properties

Network Properties	4000 Nodes	10000 Nodes
Min. Nodes Degree	1	1
Max. Nodes Degree	150	150
Mean Nodes Degree	6.692	7.674
StdDev Nodes Degree	10.788	11.84
Min. Dim. <i>FriendOfFriend</i>	1	1
Max. Dim. <i>FriendOfFriend</i>	861	2542
Mean Dim. <i>FriendOfFriend</i>	105.734	160.41
StdDev Dim. <i>FriendOfFriend</i>	74.77	139.318

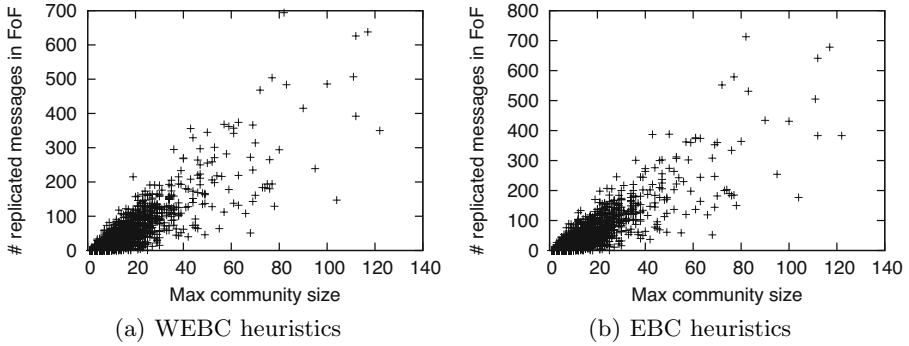
We have evaluated our protocol with the WEBC and EBC selection heuristic which selects the neighbours on the basis of the WEBC and EBC. Our protocols are compared with respect to a baseline flooding method in which source messages are transmitted to all attached network nodes. Fig. 4(b) shows the CDF of the percentage of FoF (Friend Of Friend) which have received at least a replica on each of two networks, with respect to the different solutions. The results show that our algorithm actually permits to obtain a lower number of replicas with respect to flooding, while the two heuristics are very close in term of replicated updates.

Fig. 5 show the number of replicated updates as a function of the size of the communities by using the two heuristics and by considering the network of 10000 nodes. The figures shows that the percentage of replica is similar for the two heuristics and is proportional to the size of the communities. We can conclude that the heuristics based on the WEBC outperforms the other one, because it is able both to return a similar number of replicas and to select the most important paths in terms of tie strength of the nodes.



(a) EBC and WEBC for random networks of 7208 nodes and 49170 edges obtained by the dataset (b) Replica Distribution in FoF

**Fig. 4.** Results of the simulation



**Fig. 5.** Number of replica according to the community dimension in a network of 10000 nodes

## 7 Conclusion and Future Works

This paper has presented how the Dunbar based approach can be exploited to define a Distributed Social Network. The introduction of a novel centrality index, the Weighted Betweenness Centrality, enables the definition of an efficient epidemic algorithm able to select the paths for the propagation of the social updates on the basis of the weights paired with them.

We plan to extend our work in several directions. The epidemic algorithm may be refined by pairing a history with each update in order to further reduce the number of duplicated updates. We plan to integrate with our system a strategy to guarantee the availability of the social content for peers offline. Finally, we will investigate the application of the WEBC in other contexts, for instance for the link prediction problem.

## References

1. Diaspora, <https://joindiaspora.com/>
2. Arnaboldi, V., Conti, M., Passarella, A., Pezzoni, F.: Analysis of ego network structure in online social networks. In: *SocialCom/PASSAT*, pp. 31–40. IEEE Computer Society (2012)
3. Arnaboldi, V., Guazzini, A., Passarella, A.: Egocentric Online Social Networks: Analysis of Key Features and Prediction of Tie Strength in Facebook. *Computer Communications* (2013)
4. Buchegger, S., Schioberg, D., Vu, L.H., Datta, A.: Implementing a P2P Social Network - Early Experiences and Insights from PeerSoN. In: *Second ACM Workshop on Social Network Systems (Co-located with EuroSys 2009)* (2009)
5. Cutillo, L.A., Molva, R., Strufe, T.: Safebook: A privacy-preserving online social network leveraging on real-life trust. *Comm. Mag.* 47(12) (December 2009)
6. Datta, A., Buchegger, S., Vu, L., Strufe, T., Rzađca, K.: Decentralized online social networks. In: *Furht, B. (ed.) Handbook of Social Network Technologies*, pp. 349–378. Springer (2010)
7. Everett, M.G., Borgatti, S.P.: Ego network betweenness. *Social Networks* 27, 31–38 (2005)
8. Graffi, K., Gross, C., Mukherjee, P., Kovacevic, A., Steinmetz, R.: Lifesocial.com: A p2p-based platform for secure online social networks. In: *Peer-to-Peer Computing*, pp. 1–2. IEEE (2010)
9. Guidi, B., Conti, M., Passarella, A., Ricci, L.: Distributed protocols for ego betweenness centrality computation in DOSNs. In: *The Fifth IEEE Workshop on Pervasive Collaboration and Social Networking 2014 (PerCol 2014)* (March 2014)
10. La Gala, M., Arnaboldi, V., Passarella, A., Conti, M.: Ego-net Digger: a New Way to Study Ego Networks in Online Social Networks. *Tech. rep.*, IIT-CNR (2012)
11. Leung, I.X.Y., Hui, P., Liò, P., Crowcroft, J.: Towards real-time community detection in large networks. *Physical Review E* 79(6), 66107 (2009)
12. Lu, H., Nath, B., Iftode, L., Muthukrishnan, S.: Social Butterfly: Social Caches for Distributed Social Networks. In: *IEEE Third International Conference on Social Computing (Socialcom)*, pp. 81–86 (2011)
13. Lu, H., Ponceva, M., Nath, B., Muthukrishnan, S., Iftode, L.: SocialCDN: Caching techniques for distributed social networks. In: *IEEE 12th International Conference on Peer-to-Peer Computing (P2P)*, pp. 191–202 (2012)
14. Mega, G., Montresor, A., Picco, G.P.: Efficient dissemination in decentralized social networks.. In: *Peer-to-Peer Computing*, pp. 338–347 (2011)
15. Montresor, A., Jelasity, M.: Peersim: A scalable p2p simulator. In: *Schulzrinne, H., Aberer, K., Datta, A. (eds.) Peer-to-Peer Computing*, pp. 99–100 (2009)
16. Roberts, S.G., Dunbar, R.I.M., Pollet, T.V., Kuppens, T.: Exploring variation in active network size: Constraints and ego characteristics. *Social Networks* 31 (February 2009)
17. Sharma, R., Datta, A.: Supernova: Super-peers based architecture for decentralized online social networks. In: *COMSNETS*, pp. 1–10 (2012)
18. Sutcliffe, A., Dunbar, R.I.M., Binder, J., Arrow, H.: Relationships and the social brain: integrating psychological and evolutionary perspectives. *British Journal of Psychology* 103, 149–168 (2011)
19. Wilson, C., Sala, C., Puttaswamy, K.P.N., Zhao, B.Y.: Beyond social graphs: User interactions in online social networks and their implications. *TWEB* 6(4), 17 (2012)