Ego-betweenness centrality in link streams

Marwan Ghanem, Florent Coriat, Lionel Tabourier

Ego-betweenness centrality in link streams

Marwan Ghanem, Florent Coriat, Lionel Tabourier Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6
UMR 7606, 4 place Jussieu 75005 Paris
firstname.lastname@lip6.fr

Abstract

The ability of a node to relay information in a network is often measured using betweenness centrality. In order to take into account the fact that the role of the nodes vary through time, several adaptations of this concept have been proposed to time-evolving networks. However, these definitions are demanding in terms of computational cost, as they call for the computation of time-ordered paths. We propose a definition of centrality in link streams which is node-centric, in the sense that we only take into account the direct neighbors of a node to compute its centrality. This restriction allows to carry out the computation in a shorter time compared to a case where any couple of nodes in the network should be considered. Tests on empirical data show that this measure is relatively highly correlated to the number of times a node would relay information in a flooding process. We suggest that this is a good indication that this measurement can be of use in practical contexts where a node has a limited knowledge of its environment, such as routing protocols in delay tolerant networks.

I. Introduction

Since Linton Freeman's works in the late seventies [8], *betweenness centrality* is often used to evaluate how important a node may be in order to spread an item throughout a network, whether it is a piece of information, goods or even an infection. In static graphs, the betweenness centrality of a node x is defined as the sum, over all pairs of nodes, of the fractions of shortest paths from a vertex to the other that go through x. Such a definition implies the calculation of all shortest paths in the network, which makes the exact computation of betweenness centrality expensive when the network is large. Moreover, the notion of centrality is time-dependent in most practical contexts. For example, a sick person is highly central in the infection network when contagious and in contact with many other people, but not anymore if isolated. This suggests to adapt the definition of betweenness centrality to dynamical contexts, which raises several delicate questions, not only in terms of interpretation but also in regards to computational tractability.

A usual approach to describe dynamical interactions consists in representing them as a series of snapshots of equal length, where each snapshot depicts the aggregated interactions occurring in the network during a given period of time. This approach allows to use the large toolbox designed for graphs, and in particular centrality measurements. However, it also raises several issues, which have already been brought to light in previous works (e.g. [12]). One of them is that it forces to use a specific time scale for the analysis. When the time scale is chosen too large, we loose the benefit of a time-dependent representation, and when chosen too small, the network is too disconnected and does not contain the spatio-temporal paths actually used for spreading an item from one node to another.

Therefore, it is legitimate to look for alternate definitions that would take into account the intrinsic dynamical nature of the data. For this purpose, we use a representation which focuses on the interactions and the moment when these interactions occur. Such a representation has various names and slightly different formalisms in the literature: temporal networks [11], [20], time-varying graphs [5], link streams [27] etc. In the rest of this work, we favor this last denomination as it emphasizes the possibility to analyze the data as a real-time stream of information.

In recent works, many definitions have been proposed to extend Freeman's graph-based definition to the dynamical context – [26], [15], [25], [21] among others. As we shall see, these definitions are often close to each other, although exhibiting subtle differences. In practice, the most appropriate definition certainly depends on the goal pursued, which involves practical constraints. There are a variety of situations where the definition of a temporal betweenness centrality is helpful: to detect opinion leaders, potential super-spreaders of epidemics etc. In order to set specific constraints to our study, we focus on the case of centrality evaluation in a Delay Tolerant Network (DTN), which do not guarantee end-to-end connectivity. In this context, the purpose of centrality measurements is to identify how efficient is a node for relaying messages in the communication network (its *utility*). Other kinds of centralities (closeness, eigencentrality, etc) are not discussed here, as we do not intend to make an extensive review of this very broad concept. Therefore, in the following we sometimes simply refer to *centrality* to designate betweenness centrality, either in graphs or in link streams.

Previous works have proposed centrality-based protocols in DTN. For a review, see [18]. The first approach of this kind is probably SimBet protocol [6], which mixes a graph-based ego-betweenness centrality to a similarity measurement between nodes in order to evaluate if a given node is an appropriate relay for message passing. Another good example is BubbleRap [13], in which the authors adapted the idea to temporal data by proposing a measure which is derived from the simulation of message propagation through flooding. We can also mention [28], in which the authors propose a protocol based on a definition of centrality which focuses on the duration of contacts. In these works and others, simulations often achieve good performances in terms of delivery ratio and cost when compared to benchmarks, which substantiates the validity of these approaches. However,

the large majority of these protocols use a static definition of the betweenness centrality (even if some of them take into account the recentness of the interactions) or require a training simulation period to evaluate the centrality of a node. Moreover, some of these definitions require the knowledge of the global structure of the network, which in practice does not seem to be always reachable for a node.

The achievement of a comprehensive and efficient routing protocol is out of the scope of this work. Here, we rather focus on two purposes:

- First, we propose the definition of the ego-betweenness centrality in link streams, which has interesting properties. In a few words, it takes into account the temporal information, it does not demand the knowledge of the global structure of the network, it is computationally light and parameter-free.
- Second, we investigate the comparison between various dynamical centrality measurements on several real-world datasets, to give insights on the criticality of the choice of the betweenness centrality definition implemented.

Despite the variety of definitions of temporal betweenness centralities, the underlying intuition remains similar. So we may think that the impact of choosing one definition rather than another may have moderate consequences. If this intuition is correct, we should choose a definition that is suited to our experimental constraints, which is for example the case of the ego-betweenness for DTN.

The paper is organized as follows: in Section II, we briefly review the dynamical betweenness centrality definitions that can be found in the literature, and propose our own definition adapted to ego-centered link streams. Then, Section III is devoted to experimental measurements to compare our definition to others and evaluate on real-world datasets what is the impact of choosing a given definition over another. After analyzing the observations that we have made, we conclude on the future directions, in particular the definition of a comprehensive DTN protocol based on the ego-centrality measurement.

II. Temporal betweenness centralities

A. In the literature

According to Freeman's definition [8], the betweenness centrality of a node v in a static, unweighted undirected graph is defined as

$$C(v) = \sum_{i,j \in V \times V, i<j} \frac{g_{ij}(v)}{g_{ij}},$$

where g_{ij} is the number of shortest paths between i and j and $g_{ij}(v)$ the number of these paths that go through node v. Straightforward strategies to calculate exactly the betweenness centrality usually lead to time complexities in $\Theta(N^3)$, where N is the number of nodes. Currently, a widely used approach proposed by Brandes [2] allows a computation in $O(NM)$, where N and M are the number of nodes and edges respectively. This is usually much more tractable on large sparse graphs.

However, the situations encountered in interaction networks are rarely stable. Therefore, it seems more appropriate to evaluate dynamically the importance of a node. A node may be an important relay at a given moment of the day and remain silent later, or it may gain importance progressively, etc. As a consequence, we should not only consider the overall centrality of a node during the lifetime of the network, but also whether that node is central at a certain point in time. Several works in the literature focused on adapting betweenness centrality to dynamical contexts, some of which we discuss here.

Perhaps the most natural method to represent the dynamics of interactions consists in using a sequence of static snapshots. In terms of complexity, this demands to compute centrality on each snapshot. When the network does not vary much from one snapshot to another, it may be appropriate to use a method which updates centralities. This is the principle of QUBE and its improved variants [16], which are efficient for updating centralities. However, it also has drawbacks such as a large memory consumption due to the costly pre-calculation of all shortest paths. Even if we put aside computational issues, any snapshot-based analysis misses the fact that a predefined timescale biases the analysis. For instance, a short snapshot length creates partial paths that are ignored in the betweenness computation. To illustrate this idea, an information may be immediately relayed by a node in a social network, but it may also wait for a while before being spread. In reality, the distribution of waiting times in such situations is known to be often heterogeneous (e.g. [1]).

Because of this limitation, there have been efforts to adapt the definition of betweenness to temporal representations. To do so, a natural approach consists in defining the notion of shortest path in link streams, which plays an equivalent role to a shortest path for the computation of betweenness in a graph. First, we need to define a temporal path, for this purpose we use the classic definition of a *temporal path* (or *time-respecting path* or *journey*) that can be found for example in Holme and Saramäki [11]: *"paths are usually defined as sequences of contacts with non-decreasing times that connect sets of vertices"*. Then, there are several possible options to define an equivalent of a shortest path in a temporal network, depending on the representation of the temporal data and on the choice of a time of analysis. For example, the *shortest* temporal path (minimum number of hops from the source to the destination), the *fastest* temporal path (minimum span of time) or the *foremost* temporal path (minimum arrival date to the destination) – see for example [3] for formal definitions and computational costs. Note that these denominations are not consensual and others can be found in the literature – e.g., fastest temporal path is referred to as *shortest transition* in [17].

The authors of [26] use a *trans-snapshot* representation of the dynamical data, which allows a path to go from a snapshot to the next. Then a shortest path is defined as a path that spans over the smallest number of snapshots, actually corresponding to the fastest path according to the previous typology, and the definition of the betweenness follows. They isolate important nodes in the network at a given time and show that these nodes are different from the ones that the static betweenness points out. In [15], the temporal graph representation itself defines links that connect a timestamp to the next, and a node to its future self. According to this representation, the shortest travel time from a temporal node to another is closely related to the number of hops of a path. Hence, the betweenness of a node v is computed by averaging the fraction of shortest travel time paths going through v. These two definitions are actually close to each other, since they consider the shortest travel time from a source to the destination (hence, fastest paths) to be a natural equivalent of the shortest path in a graph. However, a major drawback of these approaches is the computational cost of this path enumeration, which can be as large as $O(T^3N^3)$, T being the number of time intervals, N the number of nodes, in the case of [15] and $O(TN^3)$ in [26].

Similarly, the authors of [25] introduce the temporal coverage centrality, which can be seen as an adaptation of betweenness centrality to temporal networks, as it evaluates the importance of a node through its capacity to relay information from a vertex to another. However, it slightly differs from betweenness centrality, as coverage counts the fraction of pairs of nodes, which have a fastest path going through a given node at a given time, without normalizing it to the overall number of such paths. In [21], the authors make the choice of calling temporal betweenness of v the number of shortest time-respecting paths going through v. They also do not consider a fraction of the number of shortest paths, and refer to this quantity as *"unnormalized betweenness centrality"*. Thus, this quantity is not a direct generalization of betweenness centrality to a dynamical network, but as in the case of [25], it is cheaper in terms of computational complexity.

Note also that the various definitions of dynamical centralities do not deal with simultaneous events in the same way. In some cases (e.g. [26]), links occurring at the same moment can be involved in the same shortest path. In others, such as [15], [21], links go strictly forward in time, therefore, a shortest path cannot contain simultaneous links. In yet other cases [25], a notion of delay is integrated to the link. Depending on the application and data under study, all these choices are legitimate. In the case of a DTN, where we consider messages which are sent from a device to another, the processing is not immediate, which justifies the use of delays or of forward links. However, if the time to process the message is much shorter than the temporal resolution, it also makes sense to allow simultaneous links in a same temporal path.

The multiplicity of possible definitions leads us to a purpose-oriented point of view. In a DTN, information about the whole network is usually not available to all nodes. It is more realistic to consider that these nodes only have access to information about their direct neighbors, and we describe this point of view as ego-centered, as is the case for instance in [6]. The concept of ego-network as well as the question of how to analyze it has been widely debated in sociology (see for example [9], [4]). In the scope of this paper, an interesting aspect of this question is how the definition of centrality in ego-centered networks is correlated to the whole network centrality [19], [7]. We come back to this question later. Besides that, we aim at defining a centrality measurement that could be computed rapidly enough with limited resources, as is generally the case in DTN.

B. Ego-betweenness centrality in link streams

1) Link stream: Let us first define the notion of link stream. A link stream $\mathcal{L}$ is defined as the triplet $(V, E, \mathcal{T})$, where V is a set of nodes, $\mathcal{T} = [\mathrm{A}, \Omega]$ is a time interval, and E, a set of triplets $\{l_k\}_{k=1..|E|}$. A triplet l_k is of the form (u, v, t) with $u, v \in V \times V$, and $\mathrm{A} \leq t \leq \Omega$. Each link stands for an interaction between nodes u and v, taking place at instant t. If the interactions are directed from u to v, we will refer to the link stream as directed, and if they are not, it is undirected. Instead of t, we often use the notation t_{uv}, or $t_{u\to v}$ if the stream is directed, to indicate to which interaction t is related.

2) Most recent paths in a stream: Our definition of a (time-respecting) path in a link stream is standard: a path from u_1 to u_n is a sequence of links of the stream $\{(u_1, u_2, t_{u_1u_2}), \ \dots \ , (u_{n-1}, u_n, t_{u_{n-1}u_n})\}$ such that $\forall i$, $t_{u_iu_{i+1}} \geq t_{u_{i-1}u_i} + \epsilon$. Here, ϵ designates the delay between the sending of a message and its reception, which we consider uniform over the whole link stream, for the sake of simplicity.

Betweenness centrality of a node e in a graph is defined as the sum of the fractions of shortest paths between any pair of nodes on which e is located. Thus, defining an equivalent of this notion in link streams suggests to define an equivalent of a shortest path in an ego-centered link stream. We argue here that when looking for important information relay, an appropriate equivalent to a shortest path in a link stream is *a path that gives the most recent information to the destination about the source status*. This differs from the definitions of fastest, foremost or shortest path aforementioned.

Note that our point of view is retrospective, since at time τ we measure the paths that already exist in the link stream. The motivation for this is that we consider data as a dynamical stream, where we process each link at the time of its appearance, hence, future links are considered to be unknown. Let this specific time-respecting path be called the *most recent path* from u_1 to u_n, which is defined at a specific point in time τ, the time of analysis.

Formally, we can define a most recent path at time τ as a path $\{(u_1, u_2, t_{u_1u_2}), \ \dots \ , (u_{n-1}, u_n, t_{u_{n-1}u_n})\}$ where $t_{u_{n-1}u_n} + \epsilon \leq \tau$ and $t_{u_1u_2}$ is maximum. In the practical measurements that we implement in the following, we make the assumption – quite usual in the literature – that $0 < \epsilon = \delta t$, where δt is the time resolution of the link stream. As a consequence, if x sends a message to y at t, then y can relay this message at the next time step $t + \delta t$. On the other hand, ϵ is not null, so that if

x sends a message to y at t, while y sends a message to z, z cannot be informed of x status, which allows to eliminate the delicate simultaneity problems aforementioned.

Note also that the notions of most recent, fastest and foremost paths are closely related. If we imagine a fastest path from u to v, going through w at time t_w, this path is also a most recent path from u to v going through w at time t_v, which is the time at which it reaches v. Similarly, it is a foremost path from u to v at time t_u, when it leaves u. The difference stems essentially from the time of analysis.

3) Ego-graph and ego-link stream: There are several ways to define an ego-graph. We use the definition from [7], which is perhaps the most common. The ego-graph is composed of a node e (usually called *ego*), its links to its neighbors, and the links among its neighbors. In other words, it is the subgraph induced by e and its direct neighbors. An ego-link stream is the natural equivalent of this notion in a dynamical context. Therefore, an ego-link stream centered on e is a restriction of the link stream to the interactions between e and any of its neighbors, or between two neighbors of e. Denoting $\mathcal{N}_e$ the neighborhood of a node e, the ego-link stream is simply $\mathcal{L}_e = (\mathcal{N}_e \cup \{e\}, L_e, \mathcal{T})$, where L_e are the triplets in L which only involve nodes of $\mathcal{N}_e \cup \{e\}$.

The computation of a shortest path in an ego-graph is straightforward, as the distance between two nodes is at most 2. Consider two nodes u and v that are neighbors of e. Nodes u and v are either directly connected or they are not. In the case where they are not connected, the distance between them is 2 because of the path $u - e - v$, and possibly because of other paths $u - w - v$ passing through another neighbor w of e.

4) Most recent paths in an ego-link stream: Schematically, we represent in Figure 1 some situations that can be encountered in ego-link streams.

The time of analysis is τ. In case 1(a), direct communication from u to v at time $t_{u\to v} = 2$ allows v to know about u's status dating from time 2. Communication using e as a relay arrives at time $t_{e\to v} + \delta t = 5 > t_{u\to v}$, but it gives information to v about u's status dating from $t_{u\to e} = 1$. As a result, the most recent path at instant τ is simply $\{(u, v, 2)\}$. In case 1(b), using e as a relay allows to receive the information about u's status dating from $t_{u\to e} = 2$ while using communication from u to v through w would give u's status at instant $t_{u\to w} = 1$. As a result, the most recent path from u to v at instant τ is $\{(u, e, 2), (e, v, 4)\}$.

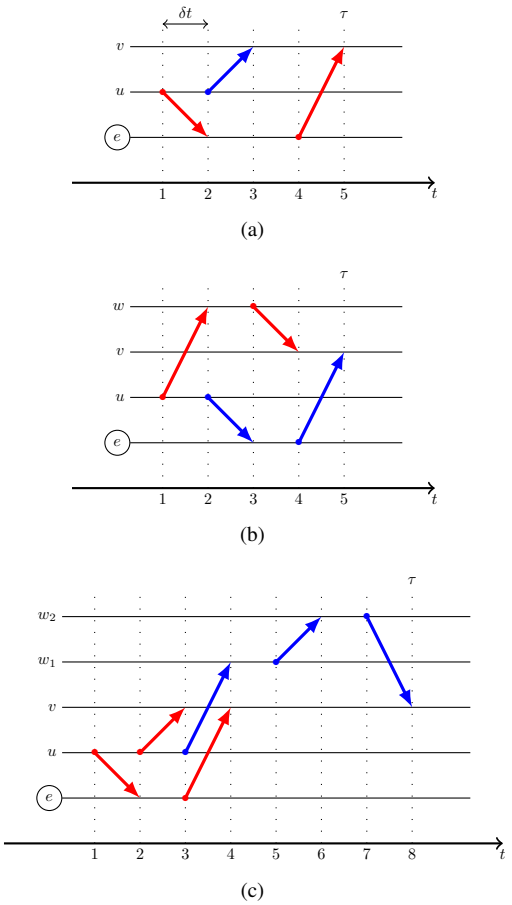

Figure 1. Various examples of most recent paths from u to v (in blue) at time τ. In red, alternative paths from u to v.

The notion of most recent path is defined for any directed pair of nodes (u, v) in a link stream, as long as there is

at least a temporal path from u to v. In other words, v is reachable from u before instant τ. More complex most recent paths can exist in a link stream, such as the case represented in Figure 1(c). Here the most recent path corresponds is $\{(u, w_1, 3), (w_1, w_2, 5), (w_2, v, 7)\}$.

5) Ego-betweenness definition and computation:

a) Definition: Ego-betweenness centrality at instant τ is defined in a similar way as centrality is in ego-centered graphs [9]. It is the sum over the directed pairs of $\mathcal{N}_e$ of the fractions of most recent paths going through e. Moreover, in the case of ego-centrality in graphs, the distance between two neighbors of e is at most 2, as previously mentioned. We would like to keep a similar idea when defining centrality in an ego-centered link stream, so that a path such as the one depicted in Figure 1(c), would not be considered when computing centrality even if is is the most recent one from u to v.

That is why we define the ego-betweenness in link streams in the following way:

$$C(e, \tau) = \sum_{i,j \in \mathcal{N}_e \times \mathcal{N}_e} \frac{p_{ij}(e, \tau)}{p_{ij}(\tau)},$$

where $p_{ij}(\tau)$ is the number of most recent paths of length at most 2 from i to j at time τ and $p_{ij}(e, \tau)$ is the number of such paths going through e. Here, *length* refers to the number of links in the path from i to j. Note that there is no restriction such as $i < j$, since the path from i to j is not equivalent to the path from j to i. This definition apply to both a directed or an undirected link stream, the difference being whether the links are directed or not.

b) Computation: We propose here an algorithm to compute the ego-betweenness of a node e. We consider the case of a directed link stream, the undirected case being simple to deduce. As we are interested in the stream around e, we only consider links involving nodes of $\mathcal{N}_e \cup \{e\}$, that is e and its neighborhood. We go through the sub-stream in chronological order. Throughout the process, we store for any directed pair of nodes (u, v) the most recent path(s) of length 1 or 2 from u to v, if it or they exist. For any directed link (u, v, t), there are three cases to consider:

1) Either $u = e$, in which case there may be new most recent paths from any neighbor w of e to v, going through e. Thus, we compare temporal paths of the form $\{(w, e, t_{w \to e}), (e, v, t)\}$ to see if they are more recent than the current most recent path from w to v, and update if necessary.
2) Or $v = e$, in which case we update the most recent path from u to e, which is now $\{(u, e, t)\}$, but it has no immediate influence on e centrality.
3) Or $u \neq e$ and $v \neq e$, in which case, there is a new most recent path from u to v, which is now $\{(u, v, t)\}$. Moreover, there may be a new most recent path from w to v of the form $\{(w, u, t_{w \to u}), (u, v, t)\}$, we thus have to compare it to the existing most recent path and update if necessary.

Then, at each step the ego-betweenness of node e is updated accordingly. The generalization to an undirected link stream is straightforward, as we simply consider that each link (u, v, t) implies a directed link from u to v and another from v to u at time t.

As already mentioned, this definition as major advantages: it is ego-centered, which corresponds to the situation where a node has limited information about its environment and reduces the computation time in comparison to dynamical betweenness on the whole stream. But, it also has several drawbacks. First, ego-centered betweenness centrality in graphs is known to have no direct theoretical relationship to the betweenness centrality *stricto sensu*, so there is certainly no simple way to relate theoretically the ego-betweenness to any existing dynamical generalization of this concept. In addition, as it takes into account the temporal evolution of the network, it is more computationally demanding than a static method would be.

c) Complexity: As the definition is ego-centered and therefore use a subpart of the stream for each node, it is difficult to make a direct comparison to the complexity of the temporal betweenness centralities aforementioned. We can express it as a function of $N_e = |\mathcal{N}_e|$, and $M_e = |L_e|$, the number of links in the ego-centered link stream. We are going through the stream once, and for each link, we have to consider at most $N_e + 1$ pairs of directed neighbors, to see if the link under consideration changes the most recent path between them. This leads to a complexity in $O(M_e N_e)$.

d) Practical example: To illustrate this definition on a practical example, we compute the evolution of the centrality of e at time t, denoted $\mathcal{C}(e, t)$, in the link stream represented in Figure 2. From time 0 to 2, there is no most recent path going through e, so that e's centrality is null. Starting from time 3, e is located on the only most recent path from u to v, so $\mathcal{C}(e, 3) = 1$. But at time 4, there is a new most recent path from u to v, which is $\{(u, v, 3)\}$, and as e is not located on it, $\mathcal{C}(e, 4) = 0$. At time 5, we identify that e is located on $\{(u, e, 1), (e, w, 4)\}$ which is the only most recent from u to w, that implies that $\mathcal{C}(e, 5) = 1$. At time 6, we observe that e is also located on a path from w to u: $\{(w, e, 4), (e, u, 5)\}$, however there is another path which is as recent: $\{(w, u, 4)\}$, $\mathcal{C}(e)$ is therefore increased by $\frac{1}{2}$ to 1.5. Finally at time 7, there are two new most recent paths which allow to relay information from w to v: $\{(w, e, 4), (e, v, 6)\}$ and $\{(w, u, 4), (u, v, 6)\}$, the first one contributes for $\frac{1}{2}$ to e's centrality, so that $\mathcal{C}(e, 7) = 2$.

III. Comparing centralities on data

In order to investigate how the dynamical centrality definition impacts the interpretation of this measurement, we are interested in comparing these measures on real-world datasets.

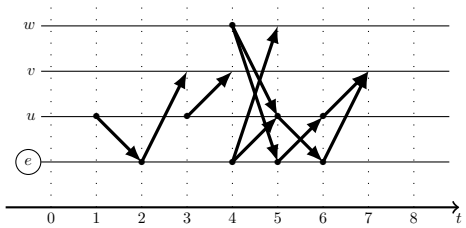

Figure 2. Example of a link stream centered on node e.

A. Dynamical betweennesses under scrutiny

First, we use a *flooding method* for the purposes of defining a reference. The idea is to simulate a flooding protocol from all the nodes of the system, and count the number of times a node relay a message according to these experiments on average. It is a means to define a common reference to all our other measurements. The goal of this choice is twofold: first, it allows to limit the number of comparisons, which is necessary because of space constraints. Second, flooding methods are actually used in practical contexts as a substitute for a centrality measurement (it is treated as such in [13] for example), and can be understood as a crude proxy to evaluate the relay-utility of a node.

Then, we use several temporal between measurements mentioned in the literature review. *Snapshot-based centrality* does not exactly belong to this category and it has already been shown that it yields quite different results from truly temporal centralities – e.g., [26] – however, it has been widely used to measure the importance of a node at a given moment in time. Note that this measurement is parametrized by the length of the snapshot. We also use the *temporal coverage centrality* [25], which aims to describe the importance of a node in a link stream with similar intentions as our definition, that is to say taking into account the dynamics without resorting to any timescale. Other dynamical centrality measurements have been considered for this comparison, in particular the one proposed by Tang *et al.* [26], however our implementation is not efficient enough to process the datasets described in the following in a reasonable time.

B. Datasets and preprocessing

1) Datasets: We compare these dynamical centrality measurements on three different undirected contact datasets:

- *Hypertext* [14] is a collection of contacts collected at the Hypertext 2009 conference in the context of *SocioPatterns* project[1]. 113 participants were equipped with radio badges recording contacts with other participants per 20 seconds windows. The total duration of the dataset is about 2 days and a half.
- *Infocom* [22] is a collection of contacts collected at the Infocom 2006 conference. Each node represents a wireless sensor recording contacts per 120 seconds windows. It contains 98 nodes consisting of 20 static devices and 78 participants. The total duration of the dataset is a little shorter than 4 days.
- *School* [24], [10] is a collection of contacts collected at a primary school, also part of *SocioPatterns* project. The dataset represents the contacts between 242 participants (pupils and teachers). A link between two nodes represents a contact within a 20 seconds window. The total duration of the dataset is about 32 hours.

Such data are particularly suited to the context of DTN. However, these centrality measurements are relevant to other cases too, such as evaluating the importance of a node in a social network. To see if we make similar observations in a different context, we also used

- *Enron* [23], this dataset contains the $47,088$ emails that 151 Enron employees exchanged during three years. It records who has sent an email to whom and when.

We summarize the main features of these four datasets in Table I, more details can be found in the references provided.

The activity of the nodes vary a lot, depending on the moment considered, which widely impacts the centrality measurements. For example, we observe large periods of low activity in the contact datasets, corresponding to the night periods. We show in Figure 3 the fraction of active nodes for each dataset through time, that is to say the nodes which have at least one contact during a given time-window (300s for the contact data, 1 week for *Enron*).

2) Computation issues and data preprocessing: Flooding and snapshot-based centralities are easy to implement and fast, but these definitions can hardly be considered as generalizations of the betweenness centrality to link streams. The computation of other dynamical centralities can be somewhat problematic. Practically, the running times on the examples that we discussed above can be quite long. We set arbitrarily the computation time limit to 100 hours on a standard working station to calculate the centralities of all the nodes of a dataset. By this standard, our implementation of coverage centrality could not achieve the computation on *Infocom* and *School*. To circumvent this problem, we modified the original time resolution of these datasets,

[1]http://www.sociopatterns.org

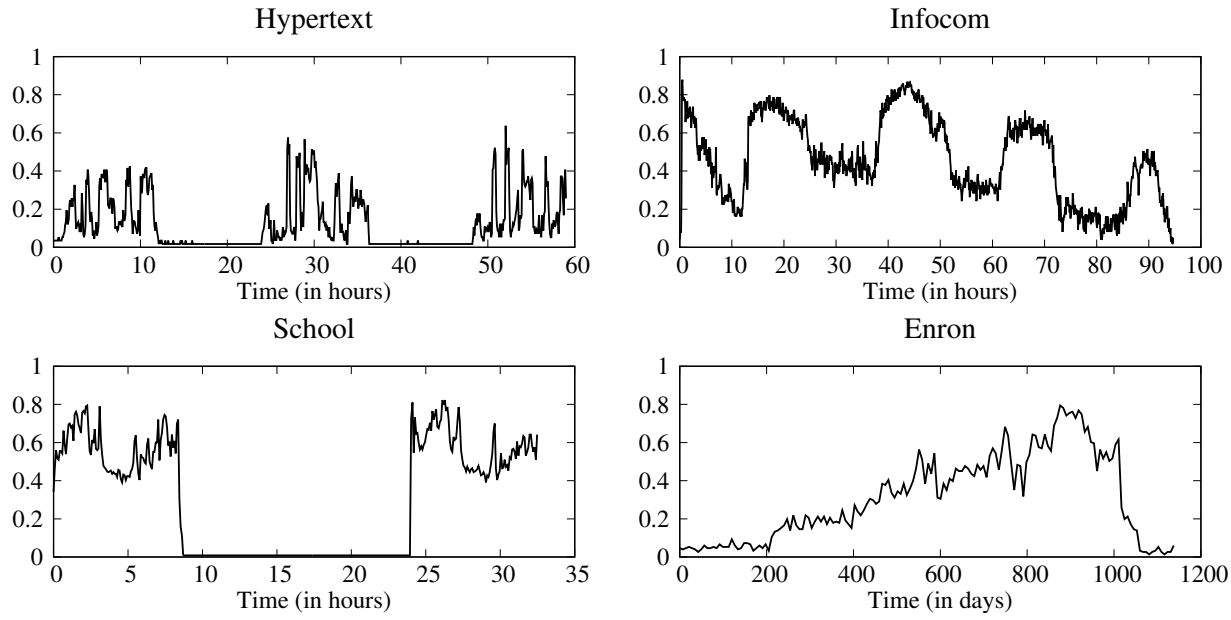

Figure 3. Fraction of active nodes in each dataset.

by gathering consecutive time steps in larger ones, as can be seen in Table I and parallelized the computation on several nodes. We then obtain the following aggregated computation times, with one significant digit:

- *Hypertext* : 90 hours (cov.), 5 minutes (ego-bet.).
- *Infocom* : 300 hours (cov.), 20 minutes (ego-bet.).
- *School* : 2000 hours (cov.), 8 hours (ego-bet.).
- *Enron* : 100 hours (cov.), 800 seconds (ego-bet.).

Concerning Tang *et al.* centrality [26], we could not compute it on any of these datasets, as mentioned earlier. Note that in both cases, we can certainly not claim that our implementations of these algorithms are optimal, or that this observation would stand with other datasets.

Dataset	$N = \|V\|$	$M = \|\mathcal{L}\|$	**Duration**	**Time resolution**
Hypertext	113	20,818	59 hours	20s
Infocom	98	98,450	95 hours	240s
School	242	46,968	32 hours	300s
Enron	151	47,088	3 years	960s

Table I
CHARACTERISTICS OF THE DATASETS.

C. Comparison tools and results

1) Comparison measures: In order to evaluate how different these centrality measurements are, we could compute the Pearson correlation coefficient through time. However, the heterogeneity of centrality values make the Pearson coefficient difficult to interpret, giving for example much weight to low centrality nodes. When considering centrality measurements, it is more relevant to focus on the ranking of nodes rather than on the value of the centrality itself. The observer is in general interested in knowing if a node is more central than another. For example, in the case of existing DTN protocols based on centrality, the condition for a message to be forwarded depends on the fact that the target is more central than the source.

Therefore, we focus on measurements that indicate if a ranking is correlated to another. For this purpose, we use the Spearman footrule correlation, which is defined as:

$$\mathcal{F}(r_1, r_2) = 1 - \frac{\sum_i |r_1(i) - r_2(i)|}{M}$$

where $r_x(i)$ designates the position of node i in ranking x, M is the maximum footrule distance, that is to say $2\lceil N/2 \rceil \lfloor N/2 \rfloor$, where N is the length of the ranking (which is also the number of nodes in the network). We also use the Kendall-tau correlation:

$$\mathcal{K}(r_1, r_2) = 1 - \frac{|\{i, j\} : r_1(i) > r_1(j) \text{ and } r_2(i) < r_2(j)|}{N(N-1)/2}$$

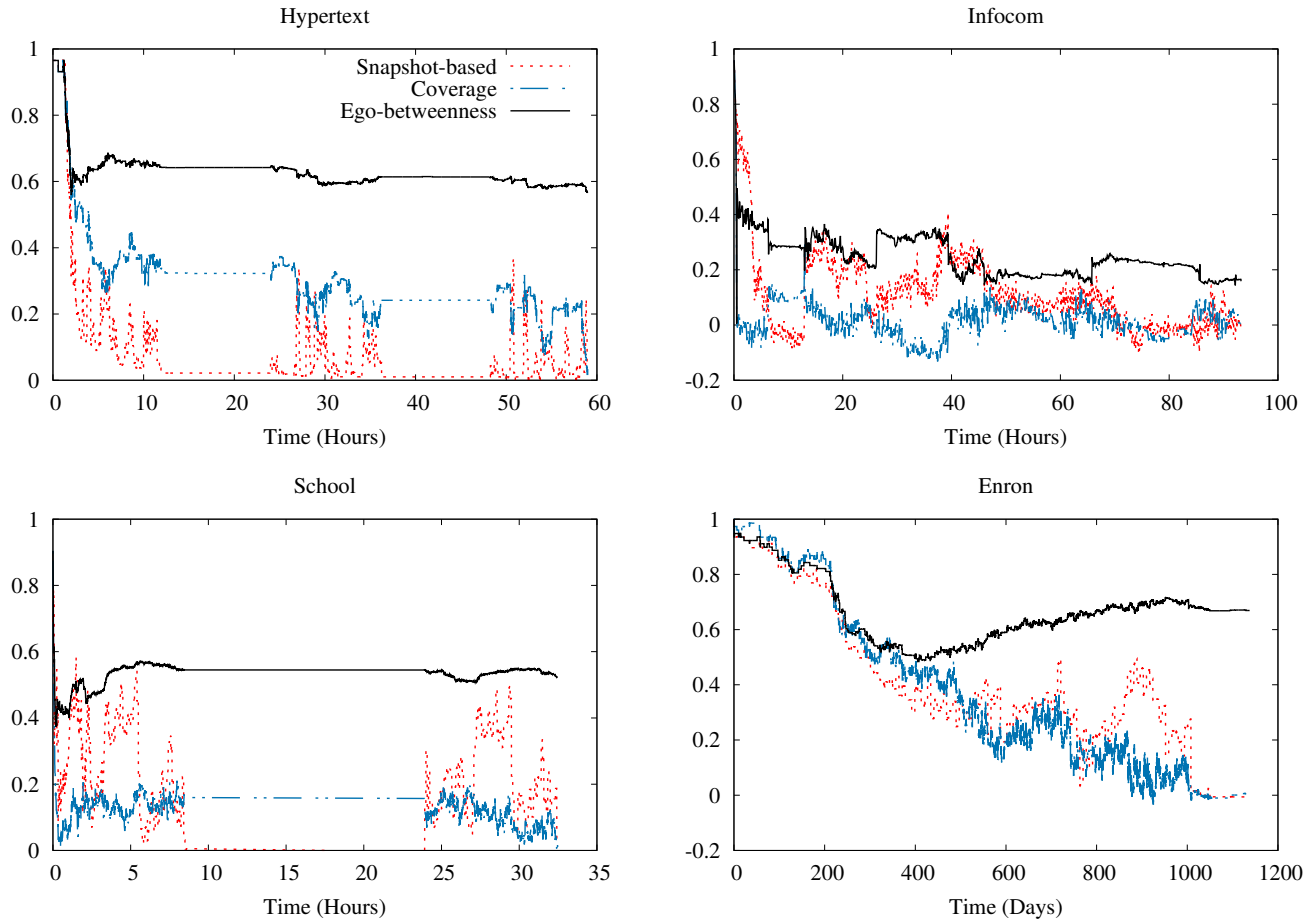

Figure 4. Spearman footrule correlations between the different centrality measurements and the ranking according to the flooding method.

These two measurements are complementary, as Spearman footrule uses the sum over all nodes of the difference of rankings, while Kendall-tau focuses on the relative rankings between two nodes, without considering by how many ranks the ranking of each node differs from one measure to the other. Note that it is frequent that two nodes share the same centrality value. In such case, we argue that the most adequate way to rank the nodes is the standard competition ranking (two nodes sharing the same score should have the same ranking).

2) Experimental results: The results of these measurements are reported in Figures 4 and 5. We can first notice that both correlation measurements behave qualitatively in a similar way, except for the fact that the Kendall-tau correlation fluctuates more. When the Kendall-tau correlation drops or increases, it is also true for the Spearman footrule. Note also that during inactivity periods, for example from hour 9 to hour 24 in *School*, as the centralities do not evolve, all measurements remain strictly constant.

We now analyze the results obtained according to each centrality measurement. Concerning the ego-betweenness centrality, there is first a transition period during which the correlation with the flooding ranking can be unstable (for example in the case of *Hypertext*), then the correlation stabilizes. This is related to the fact that the value of the ego-betweenness in the past affects the value of the ego-betweenness in the future. For example, if a node and its neighborhood are not active any longer starting from time t, then the ego-betweenness of the node remains constant. This observation is also true for the flooding score. In the cases of *Enron*, *Hypertext* and *School*, this level is quite high, as the Spearman footrule are around 0.7, 0.6 and 0.5, respectively and the Kendall tau: 0.6, 0.5 and 0.4. In the case of *Infocom* however, the level of correlation is very low ($\mathcal{F} \simeq 0.2$, $\mathcal{K}$ is close to 0). Except in this last case, it indicates that the ego-betweenness centrality yields rankings quite similar to the flooding method. For example, a Kendall-tau larger than 0.5 means that more than three quarter of the pairs of nodes are ranked in the same order according to both rankings.

Concerning the snapshot-based centrality, it is difficult to identify a clear pattern of correlation with the flooding case. During periods of low activity, the snapshots are very sparse, causing the correlation to drop to very low levels. During higher activity periods, the correlation increases according to both measurements but fluctuates around levels which are lower than what we observed with the ego-betweenness. This was partly expected, as it has been noticed in previous works that the snapshot-based centrality can yield results very different from dynamical centrality measurements. Note that we tested several snapshot sizes

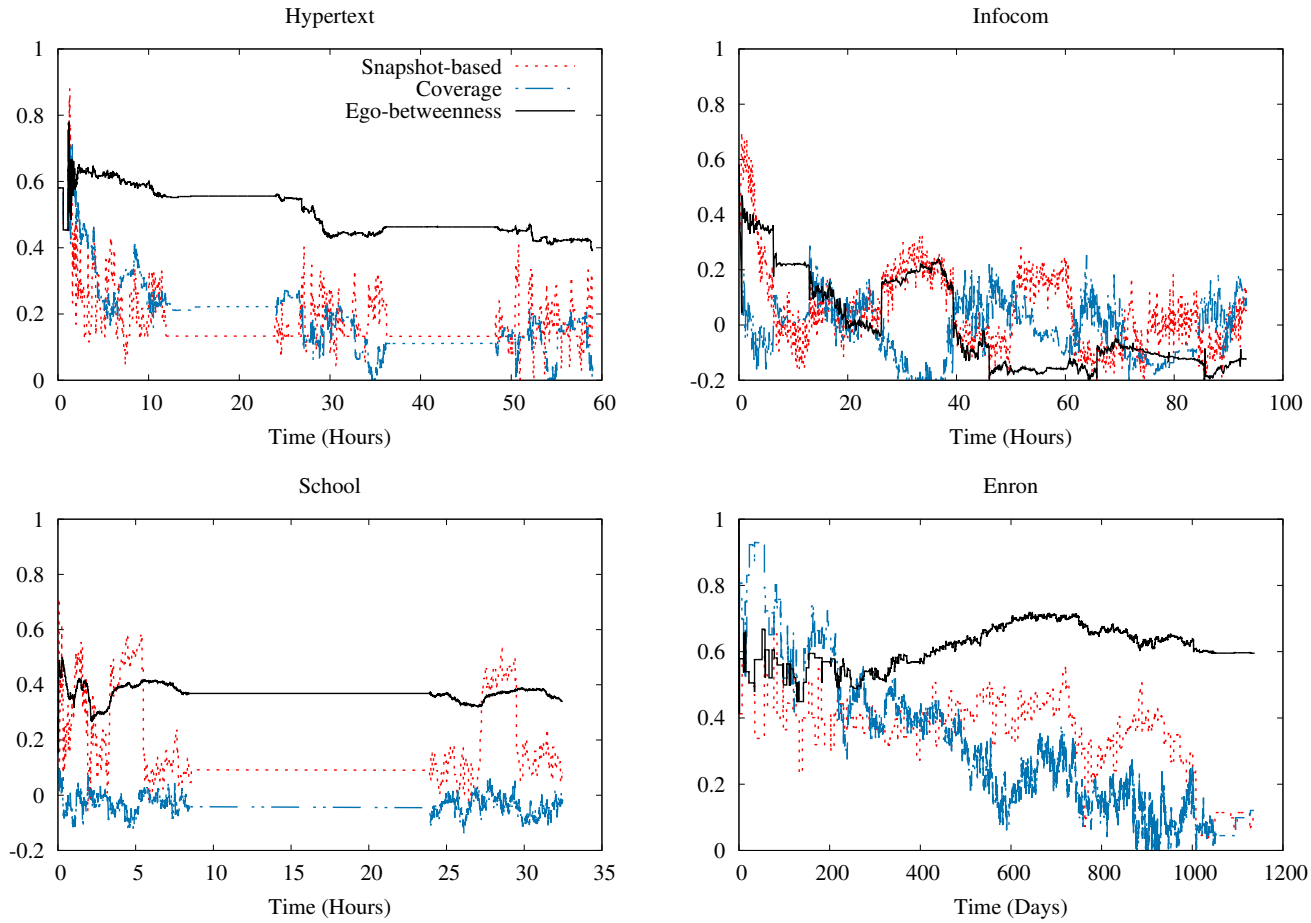

Figure 5. Kendall tau correlations between the different centrality measurements and the ranking according to the flooding method.

(5, 10, 20 and 30 minutes for contact data), and observed similar qualitative behaviors. We plotted the value corresponding to the highest level of correlation between the flooding method and the snapshot-based centrality, which corresponds to 5 minutes snapshots in all contact network cases, and 1 week for *Enron*.

More surprisingly, the coverage centrality also yields very different results from the ego-betweenness case. The level of correlation to the flooding is lower than for the ego-betweenness. Compared to the snapshot-based centrality, it varies from a dataset to another, but the correlation measurements seem to be of the same order of magnitude, positive but at quite low levels. This was unexpected as coverage centrality seems to rely on a similar intuition as the ego-betweenness. Several reasons can be evoked to explain this observation: the fact that coverage centrality is not normalized by the number of paths going from one node to another, or the difference between a fastest and a most recent path. However, we think that the most plausible cause is the fact that coverage centrality considers the whole stream, while ego-betweenness is restricted to the sub-stream around e. In a flooding experiment, a node which receives a message send it to all its future neighbors, so this measurement also relies mostly on the local structure around e. Thus, the ego-betweenness as we defined it seems to be closer to what a flooding process would do. Depending on the context, a user would have to choose which measurement is the most appropriate to his problem.

We think that the results reported in this section give useful insights about the characteristics of the link stream that a centrality measure captures. However, they also depend on the features of the datasets, and notably their size. So in the short term, we would like to compare systematically and on a larger scale the ego-betweenness to these dynamical centrality measurements and others that could not be implemented here (e.g. [26], [15], [21]).

IV. Conclusion

In this work, we defined the ego-betweenness centrality in link streams as an extension of the concept ego-betweenness centrality in graphs to a dynamical context. We also proposed a computation algorithm, which proved to be tractable on several real-world datasets. Its node-centered design allows to compute it with the mere knowledge of the neighborhood of a node. Such a property is desirable in many contexts, notably networks where there is no guarantee of an end-to-end connectivity. We compared the ego-betweenness to other centrality measurements in the literature, which also aim at assessing the utility of a

node as a relay of information in a dynamical network. We observed that most of the times, it is relatively highly correlated to a flooding-based centrality measure. Therefore, we have good hopes that the ego-betweenness could be useful to opportunistic routing in DTN. In order to develop this application, the next step is to implement this measure in a comprehensive protocol. Existing protocols based on centrality often use it jointly with a similarity measurement, thus we contemplate the idea of defining an ego-centered similarity measure achievable in this context.

ACKNOWLEDGEMENTS

The authors would like to thank Taro Takaguchi for providing the source code of the *coverage centrality* reachability computation program. We would also like to thank Louisa Harutyunyan for her useful suggestions. This work is funded in part by the European Commission H2020 FETPROACT 2016-2017 program under grant 732942 (ODYCCEUS), by the ANR (French National Agency of Research) under grants ANR-15-CE38-0001 (AlgoDiv) and ANR-13-CORD-0017-01 (CODDDE), by the French program "PIA - Usages, services et contenus innovants" under grant O18062-44430 (REQUEST), and by the Ile-de-France program FUI21 under grant 16010629 (iTRAC).

REFERENCES

[1] A.L. Barabasi. The origin of bursts and heavy tails in human dynamics. *Nature*, 435(7039):207–211, 2005.
[2] U. Brandes. A faster algorithm for betweenness centrality*. *Journal of mathematical sociology*, 25(2):163–177, 2001.
[3] B.M. Bui Xuan, A. Ferreira, and A. Jarry. Computing shortest, fastest, and foremost journeys in dynamic networks. *International Journal of Foundations of Computer Science*, 14(02):267–285, 2003.
[4] R.S. Burt. *Structural holes: The social structure of competition*. Harvard university press, 2009.
[5] A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408, 2012.
[6] E.M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pages 32–40. ACM, 2007.
[7] M. Everett and S.P. Borgatti. Ego network betweenness. *Social networks*, 27(1):31–38, 2005.
[8] L.C. Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.
[9] L.C. Freeman. Centered graphs and the structure of ego networks. *Mathematical Social Sciences*, 3(3):291–304, 1982.
[10] V. Gemmetto, A. Barrat, and C. Cattuto. Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC infectious diseases*, 14(1):695, December 2014.
[11] P. Holme and J. Saramäki. Temporal networks. *Physics reports*, 519(3):97–125, 2012.
[12] T. Hossmann, F. Legendre, and T. Spyropoulos. From contacts to graphs: Pitfalls in using complex network analysis for dtn routing. In *INFOCOM Workshops 2009, IEEE*, pages 1–6. IEEE, 2009.
[13] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *Transactions on Mobile Computing*, 10(11):1576–1589, 2011.
[14] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.F. Pinton, and W. Van den Broeck. What's in a crowd? analysis of face-to-face behavioral networks. *Journal of Theoretical Biology*, 271(1):166–180, 2011.
[15] H. Kim and R. Anderson. Temporal node centrality in complex networks. *Physical Review E*, 85(2):026107, 2012.
[16] M.J. Lee, S. Choi, and C.W. Chung. Efficient algorithms for updating betweenness centrality in fully dynamic graphs. *Information Sciences*, 326:278–296, 2016.
[17] Y. Léo, C. Crespelle, and E. Fleury. Non-altering time scales for aggregation of dynamic networks into series of graphs. In *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*, page 29. ACM, 2015.
[18] N. Magaia, A.P. Francisco, P. Pereira, and M. Correia. Betweenness centrality in delay tolerant networks: A survey. *Ad Hoc Networks*, 33:284–305, 2015.
[19] P.V. Marsden. Egocentric and sociocentric measures of network centrality. *Social networks*, 24(4):407–422, 2002.
[20] V. Nicosia, J. Tang, C. Mascolo, M. Musolesi, G. Russo, and V. Latora. Graph metrics for temporal networks. In *Temporal networks*, pages 15–40. Springer, 2013.
[21] I. Scholtes, N. Wider, and A. Garas. Higher-order aggregate networks in the analysis of temporal networks: path structures and centralities. *The European Physical Journal B*, 89(3):1–15, 2016.
[22] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. CRAWDAD dataset cambridge/haggle (v. 2009-05-29), May 2009.
[23] J. Shetty and J. Adibi. Discovering important nodes through graph entropy the case of Enron email database. In *Proceedings of LinkKDD '05*, pages 74–81. ACM Press, August 2005.
[24] J. Stehlé, N. Voirin, A. Barrat, C. Cattuto, L. Isella, J.F. Pinton, M. Quaggiotto, W. Van den Broeck, C. Régis, B. Lina, and P. Vanhems. High-resolution measurements of face-to-face contact patterns in a primary school. *PLOS ONE*, 6(8):e23176, 08 2011.
[25] T. Takaguchi, Y. Yano, and Y. Yoshida. Coverage centralities for temporal networks. *The European Physical Journal B*, 89(2):1–11, 2016.
[26] J. Tang, M. Musolesi, C. Mascolo, V. Latora, and V. Nicosia. Analysing information flows and key mediators through temporal centrality metrics. In *Proceedings of the 3rd Workshop on Social Network Systems*, SNS'10, pages 3:1–3:6, 2010.
[27] T. Viard, M. Latapy, and C. Magnien. Computing maximal cliques in link streams. *Theoretical Computer Science*, 609:245–252, 2016.
[28] K. Xu, V.O.K. Li, and J. Chung. Exploring centrality for message forwarding in opportunistic networks. In *Wireless Communications and Networking Conference (WCNC)*, pages 1–6. IEEE, 2010.