Betweenness Centrality in Delay Tolerant Networks: A Survey

Naércio Magaia[a,*], Alexandre Francisco[a], Paulo Pereira[a], Miguel Correia[a]

[a]*INESC-ID/Instituto Superior Técnico, Universidade de Lisboa,*
Rua Alves Redol, no 9, 1000-029 Lisboa, Portugal

Abstract—**Dynamic networks, in particular Delay Tolerant Networks (DTNs), are characterized by a lack of end-to-end paths at any given instant. Because of that, DTN routing protocols employ a store-carry-and-forward approach, holding messages until a suitable node to forward them is found. But, the selection of the best forwarding node poses a considerable challenge. Additional network information (static or dynamic) can be leveraged to aid routing protocols in this troublesome task. One could use centrality metrics, therefore providing means to differentiate the importance of nodes in the network. Among these metrics, betweenness centrality is one of the most prominent, as it measures the degree to which a vertex is in a position of brokerage by summing up the fraction of shortest paths between other pairs of vertices passing through it. So, in this paper, betweenness centrality is surveyed, that is, its definitions and variants in static and dynamic networks are presented. Also, a survey of standard algorithms used to compute the metric (exact and approximate) is presented. Finally, a survey and a discussion on how DTN routing protocols make use of the betweenness centrality metric and algorithms to aid message forwarding is also presented.**

Index Terms—**Delay Tolerant Networks, Betweenness Centrality, Centrality Algorithms, Routing**

1 Introduction

Delay Tolerant Networks (DTNs) [1] are networks in which end-to-end paths might not exist at all time between a source-destination pair. This contrasts with traditional networks (e.g., Mobile Ad hoc Networks - MANETs) where a continuous end-to-end path is assumed to exist before messages are exchanged. However, end-to-end connectivity allowing messages to be forwarded between any pair of nodes may never exist in real MANETs due to node heterogeneity (different radios, resources), volatile links (due to node mobility, devices being turned off or running out of battery), or even energy efficient node operation (duty cycling). With the DTN *store*, *carry* and *forward* approach, mobility issues are no longer seen as obstacles, since nodes can carry messages with them while moving until an appropriate next node is found. In this approach, messages are relayed from one node into another until they reach their destination, or they are discarded.

Despite its inherent appealing interest, DTN routing presents the challenging task of finding the most suitable node to forward messages to. A variety of network information is used to address this problem, namely: (1) dynamic network information (DNI), e.g., location information, traffic information and encounter information; (2) static network information (SNI), e.g., social relations among nodes. Through social network analysis, static network information that is more stable over time can be leveraged and used by DTN routing protocols to facilitate forwarding messages [2].

Centrality [3,4], widely used in graph theory and network analysis, can be seen as a quantitative measure of the structural importance of a given vertex (or node) in relation to others within the graph. Typically, a vertex can be considered as *central* if it plays an important role in the connectivity of the graph, e.g., if it is much required within a graph for the transportation of information, or if it is more apt to connect to other nodes in the graph. In DTNs, central nodes can be seen as good candidates to be relay nodes. The three most common centrality metrics are: degree centrality, betweenness centrality and closeness centrality [3–5]. *Degree centrality*, the simplest one, is defined as the number of links, i.e., direct neighbors, incident upon a given node. It is a local metric, as it is only determined by the number of neighbors of the node, thus not taking into consideration the global structure of the network. The other two are based on measuring shortest paths to quantify the relevance of a node. On the one hand, there is *closeness centrality*, which can be defined as the total geodesic (i.e., shortest path) distance from a given node to all other nodes. Closeness can be perceived as a measure of how long it will take to spread information from a given node to all other nodes. It lacks applicability in networks with disconnected components, that is, nodes belonging to different components do not have a

* Corresponding author. Tel.: +351 213 100 286; Fax: +351 213 145 843.
Email address: naercio.magaia@tecnico.ulisboa.pt (Naércio Magaia), aplf@tecnico.ulisboa.pt (Alexandre Francisco), Paulo Pereira (prbp@inesc.pt), Miguel Correia (miguel.p.correia@tecnico.ulisboa.pt)

finite distance among them. On the other hand, there is *betweenness centrality*, which was introduced independently in [5,6] and can be defined as the number of shortest paths passing through a given node. Betweenness takes into account the global structure of the network and can be applied to networks with disconnected components. It can be perceived as a measure of the load placed on a given node since it measures how well a node can facilitate communication among others. Betweenness is also classified as a measure of mediation [4]. However, in most networks, information does not flow only along shortest-paths [7,8]. Consider, for example, the famous small-world[1] experiment of Milgram[2] in 1967, or a modern-day equivalent [9], where despite the explicit instructions given to participants to deliver the message to the target using the most direct route possible, there is no evidence of effectiveness in this task. Consequently, besides the shortest-path, it can be assumed that more realistic betweenness metrics include also paths other than the shortest ones. Also, the determination of betweenness centrality has always been a challenging task, since in most cases it requires complete topology knowledge. So, it cannot be directly applied to dynamic networks, such as DTNs.

Some taxonomies have been proposed to classify routing protocols in DTNs. According to [10], routing protocols in DTNs are classified as *deterministic* (or *scheduled*), *enforced* or *opportunistic*. Deterministic routing happens if contact information is known a priori. Enforced routing is used to deliver messages to disconnected parts of a network (i.e., islands) by means of ferries [11] or data mules [12]. In opportunistic routing, no additional information (connectivity or mobility) is known a priori, nor special propose nodes, such as data mules or ferries, are used. Opportunistic routing can be sub classified into three basic routing primitives, namely *replication*, *forwarding* and *coding*. In the message replication scheme, a relay node carrying a message may decide to spawn a new copy of the message and forward it to a newly encountered node. This scheme can be further sub classified in *greedy*, if a new copy of a message is spawn and forwarded to any node encountered that does not contain it, *controlled*, if there is a context (e.g., time-based, probability-based or copy-based) associated with each given message keeping track of the number of copies created, and *utility-based*, if a set of parameters related to the nodes in question (i.e., the current custodian and the candidate relay) are evaluated in order to assess the candidate node suitableness as a relay for a given message destined to a certain target node. In the message forwarding scheme, a relay node carrying a message may decide to pass that message over to another node it encounters, and by doing so it relinquishes its copy of the message and ceases to be one of its custodians. In the message coding scheme, a message may be coded and processed at the source (i.e., source coding) or as it traverses the network (i.e., network coding). Another approach can be used to analyze DTN routing, despite the common goal of finding a path to a destination taking into account the available information. The rationale behind it is to treat DTN routing as a resource allocation problem, thus having an intentional effect on DTN routing instead of an incidental one. The idea behind resource allocation is to forward or replicate a message to a relay based on the available resources in order to maximize the likelihood of message delivery, whenever two nodes meet. Resource allocation routing can use any of the three basic routing primitives. Taking into account the delivery semantics [13], routing protocols are divided in *unicast*, *multicast* and *anycast*. Unicast schemes deliver messages from a single source to its single destination. Multicast schemes deliver messages from a single source to a group of destinations. Anycast schemes deliver messages from a single source to any node within the ones composing a group.

Other taxonomies were also proposed taking into account the amount of social information used. According to [14], routing protocols are divided in *social-oblivious* and *social-aware* schemes. This classification is based on the amount of social information employed while making routing decisions. In social-oblivious schemes, message replicas are randomly diffused hoping that one will reach the destination. In social-aware schemes, probabilistic delivery estimation is made by nodes buffering messages to a certain destination. The idea is to relay messages to the most promising next hop based on its successful delivery probability. The authors of [2] surveyed applications, taxonomy and design-related issues in social-aware routing protocols for DTNs. Social-aware routing protocols are classified in *self-reported* or *detected*. Self-reported routing protocols are those where routing decisions are made taking into account prior completely known social information. If nodes' social behavior is detected by means of an online method, and forwarding decisions are made based on that, these protocols are called detected routing protocols. The authors of [15] surveyed and classified social-based routing protocols for DTNs according to the *positive* or *negative effects* of their social characteristics. Positive social characteristics are those that improve the DTN routing performance. Meanwhile, if nodes attempt to maximize their own utility or conserve their resources (that may be limited during operation) they tend to behave selfishly, thus presenting a negative social characteristic. Social-(aware or based) routing protocols can be further classified as single-property or multi-property (also called hybrid) depending on the number of social properties used.

A considerable number of social-aware or social-based routing protocols for DTNs, hereinafter called *social routing protocols*, that use centrality[3] metrics have been proposed. If the most common centrality metrics are considered, these protocols can be grouped as follows: degree centrality [16–19], closeness centrality [20,21] and betweenness centrality [18,22–35]. Among these metrics, betweenness centrality has shown its relevance to problems such as identifying important nodes that control flows of information between separate parts of a network, and identifying casual nodes to influence other entities behavior (e.g., genes in

[1] The small world phenomenon is exhibited by social networks since it was observed that individuals are often linked by a short chain of acquaintances.

[2] The Milgram's 1967 experiment, is a classic example of the small world problem [67], in which 60 participants in Nebraska were asked to forward a letter to be delivered to a stockbroker in Boston. In average the chain length of intermediate holders was approximately 6, which led to the notion of 'six degree of separation'.

[3] An emphasis is given, in this paper, to the social routing protocols using betweenness centrality.

genomics or customers in marketing studies) [36]. It has been also used to analyze social networks [37–39] and protein networks [40], to identify and analyze behavior of key bloggers in dynamic networks of blog posts [41], to identify significant nodes in wireless ad hoc networks [42], to study online expertise sharing communities [43], to study the importance and activity of nodes in mobile phone call networks [44] and interaction patterns of players on massively multiplayer online games [45] and to measure network traffic in communication networks [46]. In relation to DTNs, betweenness centrality has been used by social routing protocols to enhance routing in vehicular networks [47,48], mobile social networks [49–54], pocket switched networks [55] and bus switched networks [56]. Additionally, betweenness centrality in DTNs has shown its relevance to problems such as the construction of mobile backbone [57], the offloading of data in wireless social mobile networks [58], and to the dissemination of information [59–61] and content placement [62] in opportunistic networks.

Although surveys about routing in DTNs [10,13] and social routing protocols in DTNs [2,15] have been published, to the best of our knowledge no survey about how betweenness centrality is used to enhance routing in DTNs has been published. Therefore, the contribution of this paper is threefold:

- First, a survey of betweenness centrality metrics and its variants in the literature is presented.
- Second, a survey of standard algorithms used to compute betweenness centrality (exact and approximate ones) within literature is presented.
- Third, a survey of DTN routing protocols that use betweenness centrality to enhance message forwarding is presented. In addition, a discussion on how the metric and its standard algorithms are used by DTN routing protocols is also presented.

The rest of this paper is organized as follows. Section 2 presents assumptions and notations used by static and dynamic networks. Section 3 discusses betweenness centrality metrics for static and dynamic networks. Variants of betweenness centrality are also presented. Section 4 presents exact and approximate algorithms, with their computation complexity comparison, that are used to compute betweenness centrality. In Section 5, a survey of DTN routing protocols that use betweenness centrality is presented. And finally, section 6 presents a discussion and concluding remarks.

2 Assumptions and Notations

2.1 Static Networks

For static networks, a notation similar to [63,64] is used. It is assumed that $G = (V, E)$ is a weighted, directed or undirected graph. Each vertex (or node) $v \in V$ can be identified by an integer value $i = 1,2, \dots, |V|$. Each edge (or link) $e \in E \subseteq V \times V$ is identified by a pair (i, j) representing a connection between vertex i and vertex j to which a weight $\omega(i, j)$ might be associated by $\omega : E \to \mathbb{R}^+$.

Definition 1. Walk. In a graph, a walk is a sequence of vertices $v_1, v_2, \dots, v_k$, such that $(v_i, v_{i+1}) \in E$ for $i = 1,2, \dots, k-1$, and v_1 and v_k are the walk's end vertices. The length of a walk is its number of edges. Two non-adjacent vertices are connected if there is at least one walk connecting them. Given a pair of distinct vertices $(s, t) \in V \times V, s \neq t$, a walk where all vertices and edges are distinct is considered a *path* p_{st}. The vertices s and t are called *endpoints* of p_{st} and the vertices in $\mathrm{Int}(p_{st}) = p_{st} \backslash \{s, t\}$ are the *internal vertices* of p_{st}. The shorthand $s \to t$ indicates that s is connected to t, and its transitive closure $s \xrightarrow{+} t$ indicates that s is connected to t through a path of one or more edges in G.

Definition 2. Subgraph. $H = (V_H, E_H)$ is a subgraph of $G = (V, E)$, denoted $H \subset G$, if and only if (iff) $V_H \subset V$ and $E_H \subset E$.

Definition 3. Local subgraph. H is a local subgraph with respect to a vertex $v \in V$, iff all vertices in the subgraph can be reached from v.

Definition 4. Geodesic path (or *shortest path*). A geodesic path between a pair of vertices s and t, is one with the minimum length[4] d_{st}. A path length is the number of edges connecting vertices s and t. If no paths exist between vertices s and t, then $d_{st} = \infty$. Let $S_{d_{st}}$ denote the set of shortest paths between vertices s and t, and $S_{d_{st}(v)}$ denote the set of shortest paths between vertices s and t that passes through vertex v. A vertex $v \in \mathrm{V}$ lies on a shortest path between vertices $s, t \in V$, iff $d_{st} = d_{sv} + d_{vt}$.

Definition 5. Vertex-diameter (VD). Let $\mathbb{S}_G$ be the union of all the $S_{d_{st}}$'s, for all pair $(s, t) \in V \times V$ of distinct nodes $s \neq t$. The vertex-diameter $VD(G)$ of G is the size of the shortest path in G with the maximum size, that is, it is the maximum number of vertices among all shortest paths in G, and is given by $VD(G) = \max\{|p| : p \in \mathbb{S}_G\}$.

Definition 6. Predecessors. The *predecessors* of a vertex v on the shortest path from s is $P_s(v) = \{u \in V : (u, v) \in E, d_{sv} = d_{su} + \omega(u, v)\}$.

[4] Many *geodesic paths* may exist between two vertices.

Table 1
Betweenness relevant notations.

Notation	Description	Notation	Description
δ_{st}	Pair-dependency of s, t	$c_{PB(k)}$	k-path betweenness centrality
σ_{st}	Number of shortest paths between s and t	δ_s	Dependency of a vertex s
c_B	Betweenness centrality	$\tilde{c}_B$	Unbiased betweenness estimator
c_{FB}	Flow betweenness centrality	$\tilde{c}_B^H$	Local approximation of betweenness over the local subgraph H
ς_{st}	Throughput for a fixed *s*-*t* pair	c_{EFB}	Egocentric flow betweenness
c_{CB}	Current-flow betweenness centrality	c_{DB}	Directed betweenness
c_{RWB}	Random-walk betweenness centrality	c_{B_t}	Importance of delivering messages to the destination node t
$c_{\mathcal{T}B,T}$	Temporal betweenness centrality at time T	BetwU_{vt}	Betweenness utility of node v to the destination t
$c_{\mathcal{T}B}$	Temporal betweenness centrality over the entire temporal graph	c_{GB}	Betweenness centrality of a gateway in connecting two communities
$c_{B(\varepsilon)}$	ε-betweenness centrality	$c_{B_{l,l'}}$	Inter-community betweenness
$c_{B(k)}$	k-betweenness centrality	$c_{\tilde{S}_{l,l'}}$	Optimal betweenness set

2.2 *Dynamic Networks*

For dynamic networks, a notation similar to [65,66] is used. Consider a set of entities (or vertices, nodes) V, a set of relations (or edges, links) E among these entities, and an alphabet L incorporating possible properties that such relation might have (e.g., *terrestrial link, bandwidth of 8MHz)*; specifically, $E \subseteq V \times V \times L$. It is assumed that entities' relations happen along a time span $\mathcal{T} \subseteq \mathbb{T}$ called the system's *lifetime*. The temporal domain $\mathbb{T}$ is commonly assumed to be: (1) $\mathbb{N}$ for discrete-time systems, (2) $\mathbb{R}^+$ for continuous-time systems. Thus, the dynamics of the system can be described by a temporal graph (or time-varying graph) $\mathcal{G} = (V, E, \mathcal{T}, \rho, \zeta)$, where

- $\rho : E \times \mathcal{T} \rightarrow \{0,1\}$, named *presence* function, meaning that an edge is available at a particular time.
- $\zeta : E \times \mathcal{T} \rightarrow \mathbb{T}$, named *latency* function, meaning the amount of time necessary to cross a particular edge at a particular time (edge's latency can vary in time).

Definition 7. Journeys. A journey is composed by a sequence of pairs $\mathcal{J} = \{(e_1, T_1), (e_2, T_2), \dots, (e_k, T_k)\}$, so that $\{e_1, e_2, \dots e_k\}$ that is a walk in G is also a *journey* in $\mathcal{G}$ iff $\rho(e_i, T_i) = 1$ and $T_{i+1} \geq T_i + \zeta(e_i, T_i), \forall i < k$. *departure*$(\mathcal{J})$ and *arrival*$(\mathcal{J})$ denote a journey's $\mathcal{J}$ starting date T_1 and last date $T_k + \zeta(e_k, T_k)$, respectively. Thus, journeys can be assumed as *paths over time* from a source to a destination having:

- *Topological length*, which is the number $|\mathcal{J}| = k$ of pairs, that is, the number of hops,
- *Temporal length*, which is the end-to-end duration: *arrival*$(\mathcal{J})$ − *departure*$(\mathcal{J})$.

Definition 8. Distance. Similarly to journey's length, distance in temporal graphs is also measured in terms of hops and time:

- The *topological distance* $(d_{st,T})$ from a node s to a node t at time T is given by $Min\{|\mathcal{J}| : \mathcal{J} \in \mathcal{J}^*_{st}, departure(\mathcal{J}) \geq T\}$. For a given time T, a *shortest* journey is one whose departure is $T' \geq T$ and topological length = $d_{st,T}$. $\mathcal{J}^*_{st}$ denotes the set of all possible journeys starting at node s and ending at node t.
- The *temporal length* $(\hat{d}_{st,T})$ from a node s to a node t at time T is given by $Min\{arrival(\mathcal{J}) : \mathcal{J} \in \mathcal{J}^*_{st}, departure(\mathcal{J}) \geq T\} - T$.
    - For a given date T, a *foremost* journey is one whose departure $T' \geq T$ and arrival is $T + \hat{d}_{st,T}$.
    - For any given date T, a *fastest* journey is one whose departure is $T' \geq T$ and temporal length is $Min\{\hat{d}_{st,T}: T' \in \mathcal{T} \cap [T, +\infty)\}$.

Definition 9. Temporal graphs as a sequence of footprints. Given a temporal graph $\mathcal{G} = (V, E, \mathcal{T}, \rho, \zeta)$, a *footprint* of this graph from T_1 to T_2 is the static graph $G^{[T_1,T_2)} = \left(V, E^{[T_1,T_2)}\right) \mid \forall e \in E, e \in E^{[T_1,T_2)}$ iff $\exists T \in [T_1, T_2), \rho(e, T) = 1$. Specifically, the footprint aggregates all interactions of a given time windows ω into static graphs. Considering that $\mathcal{T}$ is partitioned in consecutive sub-intervals $\tau = [T_0, T_1), [T_1, T_2), \dots, [T_i, T_{i+1}), \dots$; so that, $[T_k, T_{k+1}) \Longleftrightarrow \tau_k$ denotes a *sequence of footprints* of $\mathcal{G}$ according to τ is $SF(\tau) = G^{\tau_0}, G^{\tau_1}, \dots$.

Table 1 presents betweenness relevant notations used in the following sections of this paper.

3 Betweenness Centrality

In this section, betweenness centrality metrics and variants for static and dynamic networks are presented.

3.1 Concepts

In networks, the importance of a vertex or edge can be determined by the number of paths in which it participates. Centrality denotes the order of importance that vertices or edges have in a network by assigning real values to them. Since shortest paths are defined for both vertices and edges, centrality can be computed for a vertex v or an edge e (i.e., an element x) as presented below.

3.1.1 Shortest-Path

The *shortest-path betweenness centrality* [63] of an element x, which can be a vertex v or an edge e, is based on the number of shortest paths that contain x. Let δ_{st} denote the fraction of shortest paths between the pair of vertices s and t containing vertex v, i.e.,

$$\delta_{st}(v) = \frac{\sigma_{st}(v)}{\sigma_{st}}$$

where $\sigma_{st} = \left|S_{d_{st}}\right|$ and $\sigma_{st}(v) = \left|S_{d_{st}(v)}\right|$. The ratio $\delta_{st}(v)$, also called pair-dependency of s, t on v, can be considered as the probability of any communication between vertices s and t involving vertex v. The shortest-path betweenness centrality $c_B(v)$ is defined as

$$c_B(v) = \sum_{s \neq v \in V} \sum_{t \neq v \in V} \delta_{st}(v) \qquad (1)$$

From equation (1), one can conclude that the shortest-path betweenness centrality of a vertex measures the control over communications between others, since the shortest paths ending and starting in v were excluded. In disconnected networks, any pairs of vertices s and t without any shortest paths between them must add zero to the shortest-path betweenness centrality of every other vertex in the network.

For an edge e, the pair-dependency of s, t on e is given by

$$\delta_{st}(e) = \frac{\sigma_{st}(e)}{\sigma_{st}}$$

and, the shortest-path betweenness centrality $c_B(e)$ of edge e is given by

$$c_B(e) = \sum_{s \in V} \sum_{t \in V} \delta_{st}(e)$$

3.1.2 Flow

It was previously mentioned that shortest path based centrality metrics assume that the flow of information happens along the shortest paths. By considering small-world experiments [9,67], one could assume that despite the shortest paths, a more realistic betweenness metric also included paths other than the shortest ones. In [7], a more sophisticated betweenness metric, called *flow betweenness centrality*[5], was proposed also including contributions from non-shortest paths.

According to [68], flow betweenness centrality of a vertex v is defined as the amount of flow through v when the maximum flow [69] is transmitted from s to t, averaged over all s and t. Since there might not be a unique solution to the flow problem, flow betweenness centrality can be more adequately defined as the maximum possible flow through a vertex v over all possible solutions to the st maximum flow problem[6], averaged over all s and t [7]. It can be seen as a measure of betweenness of vertices in a network in which a maximum amount of information is uninterruptedly pumped between all sources and targets. Flow betweenness centrality cannot be computed directly by counting paths as the set of edge-independent paths among pairs of nodes are not unique.

Let W denote the matrix of maximum flows among nodes, that is, the number of edge-independent paths among them, and $W[v]$ be the principal submatrix of W, that is, the matrix resulting from W by removing column and row v. Additionally, let $W[v]^*$ be the matrix obtained by deleting node v from the original network, and recalculating the flow matrix. The flow betweenness centrality is given by

[5] It is called flow betweenness centrality because of the association between the number of edge-independent paths among pairs of nodes and the quantity of material that could flow from one node to another through all possible edges [118].

[6] The maximum flow problem can be solved using standard algorithms [69].

$$c_{FB}(v) = \sum_{st} \frac{w[v]_{st} - w[v]^{*}_{st}}{w[v]_{st}} \qquad (2)$$

Other betweenness metrics can be obtained from equation (2) by changing matrix W. Hence, to compute the shortest-path betweenness centrality, W becomes the shortest path count matrix in which w_{st} gives the number of shortest paths from s to t. Note that the value returned by (2) is twice the one returned by (1).

3.1.3 Current-Flow

In the *current-flow betweenness centrality* [63] metric, which is another alternative to the shortest-path betweenness centrality metric, the flow of information follows the behavior of an electrical current flowing through an electrical network. An electrical network is defined by a connected undirected graph $G = (V, E)$, together with a conductance function $c : E \to \mathbb{R}$. A supply function $b : E \to \mathbb{R}$, specifies an external electrical current entering and leaving the circuit.

Similar to shortest-path betweenness centrality, that counts the fraction of shortest *s*-*t*-paths through a vertex, current-flow betweenness of a vertex characterizes the portion of unit *s*-*t*-supplies through that vertex. The throughput of a vertex v, for a fixed *s*-*t* pair, forms the current-flow equivalent of $\sigma_{st}(v)$ through v, i.e., with respect to a unit *s*-*t*-supply b_{st}, the throughput of vertex $v \in V$ is

$$\varsigma_{st}(v) = \frac{1}{2}\left(-|b_{st}(v)| + \sum_{e \ni v} |x\vec{e}|\right)$$

where $-|b_{st}(v)|$ sets to zero the throughput of a vertex with non-zero supply. This guarantees that a given unit *s*-*t*-supply is not considered for the throughput of its source node s and sink node t, for the current-flow betweenness. Thus, the current-flow betweenness centrality $C_{CB} : V \to \mathbb{R}$ for an electrical network $N = (G = (V, E)c)$ is

$$c_{CB}(v) = \frac{1}{(n-1)(n-2)} \sum_{s,t \in V} \varsigma_{st}(v)\,, \forall v \in V$$

where $1/(n-1)(n-2)$ is a normalizing constant. Current-flow betweenness centrality measures the portion of throughput through vertex v taken over all possible source-destination pairs.

3.1.4 Random-Walk

Due to the lack of global knowledge, it may not be possible sometimes for a vertex to compute shortest paths. For these cases, an alternative way of traversing the network can be used by means of a random-walk model. The random walk model consists of walking from vertex to vertex, through the network's edges, i.e. an edge is randomly selected from a vertex v to be followed, and the process is repeated from the new vertex.

It is assumed here that the graph is unweighted, connected and undirected. If, for example, a vertex s wants to send a message to a vertex t but neither s nor its adjacent vertices knows how to reach t through the shortest path, each vertex that gets the message for t, then selects at random one of its adjacent vertices to send the message.

It was demonstrated in [63] that the random-walk betweenness centrality $c_{RWB} : V \to \mathbb{R}$ is equivalent to the current-flow betweenness centrality, that is $c_{RWB}(v) = c_{CB}(v), \forall v \in V$. For a more detailed discussion, please refer to [63].

3.1.5 Ego

An ego network, also known as the neighborhood network (or first order neighborhood) of the ego, can be defined as a network consisting of a single actor (ego) along with the actors it is connected to (alters) and all links among the latter. Ego networks allow an easier collection of data if compared to collecting data from the entire network, because the ego usually provides complete information of the alters (including how they are connected). By sampling such information, statistically significant conclusions about the entire population can be attained [70].

As previously stated, centrality measures allow finding the most important actors within a network and betweenness centrality studies the degree to which an actor is among all other actors within the network. If an actor is between two other actors, it follows that no alters on the path connecting the actors share a connection, or else it would form a shortest path. Hence, there is a connection between the betweenness centrality of the actor in the whole network and the one in the ego network (even though it may be difficult to quantify this association) [70]. Previous works have provided evidence of the usefulness of betweenness centrality in ego networks [71].

To compute ego betweenness centrality (EBC), first it is necessary to compute betweenness of a single actor. Due to the ego networks' structure, the shortest paths in the network are either of length 1 or 2. Every single pair of non-adjacent alters must have a shortest path of length 2 which passes through the ego. Note that shortest paths of length 1 do not contribute to the betweenness computation.

Table 2
A summary and comparison of the betweenness centrality metric.

Betweenness metric	Network Type	Main idea	Network knowledge	Drawbacks	Comments
Shortest path	Static	The flow of information happens along the shortest paths.	Global	The flow of information may not take the shortest-path (e.g. the small-world experiments).	It measures the control over communications between others.
Flow	Static, Dynamic	Although preferring shortest paths, the flow of information tries to exploit all possible paths.	Global	The flow of information may not be maximum and not follow optimal flow paths from source to target nodes.	It is based on the idea of maximum flow. It is a measure of betweenness of vertices in a network in which a maximal amount of information is continuously pumped between all sources and targets.
Current-flow	Static	The flow of information follows the behavior of an electrical current flowing through an electrical network.	Global	It can only be applied to electrical networks.	It is equivalent to random-walk betweenness. The current flows along all paths from source to target, but more on along the shortest ones (i.e., the ones in which the resistance is smaller).
Random-walk	Static, Dynamic	Uses the random-walk model to traverse the network.	Partial	It includes contributions from many paths that are not optimal in any sense.	It is suitable to a network in which information wanders around at random until it finds its target.
Ego	Static	It consists in summing the reciprocals of entries given by number of shortest paths of length 2 between a pair of non-adjacent vertices.	Partial	It is difficult to normalize the metric scores with respect to the ego network size.	There is no direct connection between the betweenness centrality computed for the entire network and the EBC.
Temporal	Dynamic	It is the fraction of fastest journeys among the shortest ones that pass through a given vertex.	Global	Similar to the shortest-path version.	It is based on the concept of shortest journeys. It measures the control over communications between others over time.

Let A be the adjacency matrix of G, then A^2_{ij} contains the number of walks of length 2 connecting vertex i and vertex j. The shortest paths can be obtained by counting the number of paths of length 2 of non-adjacent pairs of actors. So,

$$A^2[1-A]_{ij} \qquad (3)$$

where *1* is a matrix of all 1's, and (3) gives the number of shortest paths of length 2 between i and j. The ego betweenness centrality is given by the sum of the reciprocal of entire entries.

Computing ego betweenness centrality of the entire network is one order of magnitude faster than computing, for example, the shortest-path betweenness centrality.

3.1.6 Temporal

Similar to the shortest-path betweenness centrality metric used in static networks, the *temporal betweenness centrality* [66] of a vertex v could be defined as the fraction of shortest journeys that pass through v. However, besides the shortest journeys that pass through a vertex, it is also important to consider for how long a vertex along the shortest path holds a message before forwarding it, i.e., the fastest journeys among the shortest ones. Therefore, the temporal betweenness centrality of a vertex v at time T is:

$$c_{\mathcal{T}B,T}(v) = \frac{1}{(n-1)(n-2)} \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{\psi_{st,T}(v)}{\psi_{st,T}}$$

where $\psi_{st,T}(v)$ returns the number of fastest journeys among the shortest ones from s to t passing through vertex v.

The temporal betweenness for vertex v over the entire temporal graph $\mathcal{G} = G^{[T_{min}, T_{max})}$ is:

$$c_{\mathcal{T}B}(v) = \frac{1}{|SF(\tau)|} \sum_{\tau=0}^{|SF(\tau)|} C_{\mathcal{T}B,T}(v)((T \times w) + T_{min})$$

where $|SF(\tau)|$ is the number of graphs in the sequence.

Table 2 presents a summary and comparison of betweenness centrality based on the metrics, the type of network, the main idea, the type of network knowledge (global or partial), drawbacks and comments.

3.2 Variants

In this section, variants of betweenness centrality proposed in the literature are presented.

3.2.1 Canonical-path betweenness

The authors of [72] proposed a simple variant of betweenness centrality, called canonical path betweenness centrality (or simple canonical centrality), in which only a single canonical shortest path between any source-target pair is considered. The reasoning behind this variant are road networks, where multiple routes do exist in practice, but they usually share most edges. As a result, in general, $\delta_{st}(v)$ is one or zero. Also, unique shortest paths are enforced by perturbing the edge weights in some route planning methods [72].

3.2.2 ε-betweenness

In [73], the authors considered terrorist networks models [74–76], which may be large, dynamic and characterized by uncertainty[7]. Terrorist networks are considered: large, as the networks are unknown, i.e., the set of actors being monitored is likely a superset of those actually engaged in illicit activities, and dynamic, as the knowledge we have of them changes over time. Network's dynamics, i.e., mobility and nodes joining or leaving the network, may reflect inaccuracies in the shortest path calculations (besides the uncertainty in the shortest path length between a pair of nodes, the path itself may also change) causing perturbations in the betweenness centrality values. So, it may be necessary to recalculate betweenness values as the network evolves through time. Some algorithms [77,78] have been proposed that support network's dynamics, thus avoiding, for example, the recalculation of shortest paths between all pairs of nodes.

A path p_{st} is called an ε-shortest path if $|p_{st}| \leq (1+\varepsilon)d_{st}$. The ε-betweenness centrality is defined as

$$c_{B(\varepsilon)}(v) = \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{\sigma_{st}^{\varepsilon}(v)}{\sigma_{st}^{\varepsilon}}$$

where $\sigma_{st}^{\varepsilon}(v)$ is the number of ε-shortest paths that include vertex $v \in V$, and $\sigma_{st}^{\varepsilon}$ is the number of ε-shortest paths between s and t in G. No analytical or empirical results on the stability of the metric was provided.

3.2.3 Bounded-distance betweenness

In [4,79], the authors limited the length of paths based on the idea that very long paths were only occasionally used, consequently not contributing to the betweenness centrality of a node. This metric was called bounded-distance betweenness centrality (also known as k-betweenness), where k gives the maximum length of paths counted. The bounded-distance betweenness centrality of a vertex v is defined as the sum of dependencies of pairs at most k hops apart, that is,

$$c_{B(k)}(v) = \sum_{\substack{s \neq v \in V \\ d_{st} \leq k}} \sum_{\substack{t \neq v \in V \\ d_{st} \leq k}} \delta_{st}(v).$$

The bounded-distance betweenness centrality ($c_{B(k)}$) only considers contributions from shortest paths whose lengths are bounded by a constant k. For $k = n - 1$, it is equal to equation (1), and, for $k = 2$, it is similar to EBC (Section 3.1.5) differing in that shortest paths of length two with a non-neighbor as intermediate are also taken into account.

3.2.4 Distance-scaled betweenness

Another variant of betweenness mentioned in [4,79] counts paths of all lengths, but weights all shortest paths inversely in proportion to their length as in

$$c_{B(k)}(v) = \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{\delta_{st}(v)}{d_{st}}$$

This metric is called *length-scaled betweenness* since the dependencies are scaled by a factor depending only on the length of the shortest path, being the same for all its inner vertices.

3.2.5 α-weight betweenness

The authors of [80] proposed a variant of the betweenness centrality metric for weighted networks that incorporates both the number of ties between nodes (i.e., communication, cooperation, friendship, or trade) and their weights. The weight of a tie can have different meanings depending on the context. For example, in social networks, it can be seen as a function of duration, emotional intensity, intimacy, or exchange of services, whereas in non-social networks, it quantifies the capacity or capability of the tie, such as the number of seats among airports, or the number of synapses and gap functions in a neural network.

[7] In covert networks, there may be a deliberate effort to hide illicit activity, thus dynamicity and uncertainty also apply.

Table 3

A summary and comparison of the variants of betweenness centrality.

Variant	Betweenness metric	Main idea	Comments
Canonical-path betweenness	Shortest-path	Only a single canonical shortest path between any source-target pair is considered.	Used in road networks.
ϵ-betweenness	Shortest-path	Consists in dynamically updating betweenness centrality in face of network's changes.	Used in terrorist networks analysis.
Bounded-distance betweenness	Shortest-path	Considers only contributions from shortest paths whose lengths are bounded by a constant k.	NA
Distance-scaled betweenness	Shortest-path	The longer a path, the less valuable it may be to control it.	NA
α-weight betweenness	Shortest-path	It incorporates both the number of ties and their weights in weighted networks.	There are also variants for degree and closeness centrality that incorporate the tuning parameter.
k-path betweenness	Random-walk	It is based on a similar assumption about the random traversal of a message from a source *s*. It is assumed that the message's traversals are only along random simple paths of at most k edges.	Nodes with high k-path centrality have high node betweenness centrality.

It is commonly assumed when analyzing shortest paths that intermediate nodes may increase the cost of interaction. If a high number of intermediate nodes is considered, the necessary interaction time between nodes increases. Intermediate nodes are also in the position of powerful third-parties, being able to distort or delay information between nodes.

Let α be a tuning parameter which determines the relative importance of the number of ties compared to tie weights. So, the length of the shortest path between two nodes is given by

$$d_{st}(\omega\alpha) = \min\left(\frac{1}{(\omega_{si})^{\alpha}} + \cdots + \frac{1}{(\omega_{it})^{\alpha}}\right) \qquad (4)$$

Equation (4) is an extension of the implementation in [81][82] of the Dijkstra's algorithm [83] by taking into account the number of intermediate nodes. Both the tie weight and the number of intermediate nodes affect the identification of shortest paths. If $\alpha = 0$, the definition falls back to d_{st}, whereas if $\alpha = 1$, the definition falls back to Dijkstra's algorithm. If $0 < \alpha < 1$, a shortest path composed of weak ties is preferred over a longer one with short ties. On the other hand, if $\alpha > 1$, the influence of extra intermediate nodes is insignificant in comparison to the strength of the ties and paths with additional intermediaries are favored.

3.2.6 k-path betweenness

k-path betweenness [36] is based on the random traversal of a message from a source s similarly to the random-walk betweenness. The following assumptions were made: (1) messages' traversals are only along single-paths, and (2) messages' traversals are only along paths of at most k edges, where k is network dependent.

The k-path betweenness centrality of a vertex v is defined as the sum over all possible source nodes s of the probability that a message originating from s goes through v, assuming that the message's traversals are only along random simple paths of at most k edges.

Let p_{sl} be an arbitrary simple path with start vertex s and having $l \leq k$ edges, i.e., $p_{sl} = \{s, u_1, u_2, \dots, u_{l-1}, u_l\}$, and let $N(u_i)$ denote the set of outgoing neighbors of $u_i, \forall i : 0 \leq i \leq l$. For every vertex v of G, $c_{PB(k)}(v)$ is given by

$$c_{PB(k)}(v) = \sum_{\substack{s \neq v \in p_{sl} \\ d_{sl} \leq l}} \sum_{1 \leq l \leq k} \frac{\chi[v]}{\prod_{i=1}^{l} |N(u_{i-1}) - \{s, u_1, u_2, \dots, u_{i-2}\}|},$$

where $\chi[v : v \in p_{sl}]$ is 1 if v lies on p_s, and 0 otherwise.

Table 3 presents a summary and comparison of betweenness centrality variants based on the type of variant, the betweenness metric, the main idea and relevant comments.

4 Algorithms

Betweenness centrality is one of the most widely used centrality metrics in social and complex networks analysis and it is based on shortest paths enumeration. Since it requires the computation of all shortest paths between a given pair of nodes, its exact determination is computationally-expensive. Betweenness computation requires $O(n^3)$ time and $O(n^2)$ space, where n is the number of vertices in the network [81].

In this section, algorithms used to compute standard betweenness centrality are presented. These algorithms can be either exact [81] or approximate, and the latter can be subdivided according to the type of techniques used, namely random sampling [72,84,85], adaptive sampling [86] and local techniques [87].

4.1 Exact Computation

Brandes (2001). In [81], the author proposed an algorithm to evaluate simultaneously all centrality metrics based on shortest paths, thus reducing the algorithm's time and space requirements. The proposed approach integrates well with traversal algorithms that solve the single-source shortest-paths (SSSP) problem.

It can be seen from equation (1), that in order to determine betweenness centrality two steps are necessary: in the first step, it is necessary to compute the length and number of shortest paths between all pairs; in the second step, it is necessary to sum all pair-dependencies. The author observed that the second step of the betweenness centrality computation was responsible for its complexity.

For $s \neq v \in V$, the combinatorial shortest path counting is given by

$$\sigma_{sv} = \sum_{u \in P_s(v)} \sigma_{su} \tag{5}$$

By applying (5) in traversal algorithms like BFS and Dijkstra's, and if the priority queue is implemented with a Fibonacci heap [88], the algorithms run times become $\mathcal{O}(m)$ and $\mathcal{O}(m + n \log n)$, respectively (where m is the number of edges in the network).

To reduce the complexity of the second step of the algorithm, i.e., the need for explicit summation of all pair-dependencies, the concept of *dependency* of a vertex $s \in V$ on a single vertex $v \in V$, was introduced in [81] as

$$\delta_s(v) = \sum_{t \in V} \delta_{st}(v)$$

and it was also observed that these dependencies obey a recursive relation.

Theorem 1 ([81]**)**. *The dependency of $s \in V$ on any $v \in V$ obeys*

$$\delta_s(v) = \sum_{w:\, v \in P_s(w)} \frac{\sigma_{sv}}{\sigma_{sw}} (1 + \delta_s(w)) \tag{6}$$

Algorithm 1. *First, for each vertex $s \in V$ do a SSSP computation, maintain during the process the lists of predecessors $P_s(v)$. Then, for every $s \in V$ compute the dependencies $\delta_s(v)$ for all other $v \in V$ using the list of predecessors and the information along the directed acyclic graph of shortest paths. Finally, in order to obtain the centrality index of a vertex v, compute the sum of all dependencies values.*

Thus, for weighted and unweighted graphs, betweenness centrality can be computed in $\mathcal{O}(nm + n^2 \log n)$ and $\mathcal{O}(nm)$ times, respectively and $\mathcal{O}(n + m)$ space. For additional details, please refer to [81].

4.2 Approximation Techniques

Taking into account that, for large-scale graphs, the exact centrality computation is computationally-expensive, some approximation techniques have been proposed, which are introduced in the following sections.

4.2.1 Random Sampling Techniques

Brandes and Pich (2007). The authors of [84] presented an experimental study of estimators for centrality metrics based on a restricted number of SSSP computations from selected source vertices (a generalized approach of [89]). Source vertices, also known as *pivots*, are those from which the shortest path computations are initiated.

Let $X_1, X_2, \dots, X_k$ be independent random variables, so that

$$\bar{X} = \frac{X_1 + X_2 + \cdots + X_k}{k}$$

and $\mu = E[\bar{X}]$ is the expected mean.

Theorem 2 ([90]**)**. *If $X_1, X_2, \dots, X_k$ are independent, $a_i \leq X_i \leq b_i, i = 1,2, \dots, k$, then for $\xi > 0$*

$$Pr\{|\bar{X} - \mu| \geq \xi\} \leq e^{-2k^2\xi^2 / \sum_{i=1}^{k}(b_i - a_i)^2} \tag{7}$$

By (6), the contribution of the source vertex $s_i \in V$ to the centrality of a vertex $v \in V$ is given by $\delta_s(v)$. In order to extrapolate, for a single estimate, from the average contributions of k source vertices, let

$$X_i(v) = \frac{n}{n-1}\delta_s(v)$$

be the random variable. To apply the bounds given by (7), let $a_i = 0$, $b_i = \frac{n}{n-1}(n-2)$, and $\xi = \varepsilon(n-2)$. Since the expectation of estimate $\frac{1}{k}\left(X_1(v) + X_2(v) + \cdots + X_k(v)\right)$ is the sum of all dependencies values on v, (7) guarantees that the error is bounded from above by $\varepsilon(n-2)$ with probability at least $e^{-2k\left(\frac{\varepsilon(n-1)}{n}\right)^2}$.

The authors of [84] concluded that in order for the contributions of $X_i(v)$ to be independent, the pivots needed to be selected at random. A drawback of this approach happens to unimportant nodes near a pivot, since it produces large overestimates of the betweenness centrality values. For example, if a one-degree node is selected as a pivot, the betweenness centrality of a two-degree node connecting the former to the rest of the network is overestimated by a factor of n/k.

Geisberger et al. (2008). In [72], the authors proposed a generalized framework, which uses canonical centrality, for unbiased approximation of betweenness centrality to address the overestimates' problem of unimportant nodes near the pivots.

Let the proposed estimator be parameterized by

- $\ell : E \rightarrow \mathbb{R}$ on the edges, named *length function*
- $f : [0,1] \rightarrow [0,1]$, named *scaling function*.

Let $P = (e_1, e_2, \ldots, e_k), k = |E|$ be a path, such that $\ell(P) = \sum_{i=1}^{k} \ell(e_i)$. The algorithm performs, in each interaction, one of $2n$ possible (forward or backward) shortest path searches with uniform probability $1/2n$. A *scaled contribution* can be defined as

$$\delta_{st}(v) = \begin{cases} \dfrac{f\left(\ell(S_{d_{sv}})/\ell(S_{d_{st}})\right)}{\sigma_{st}} & \text{for forward search} \\ \dfrac{1 - f\left(\ell(S_{d_{sv}})/\ell(S_{d_{st}})\right)}{\sigma_{st}} & \text{for backward search} \end{cases}$$

So, v gets the following contributions

$$\delta(v) = \begin{cases} \sum_{t \in V} \left\{\delta_{st}(v) : S_{d_{st}} \in S_{d_{st}}(v)\right\} := \delta_s(v) & \text{for forward search} \\ \sum_{s \in V} \left\{\delta_{st}(v) : S_{d_{st}} \in S_{d_{st}}(v)\right\} := \delta_t(v) & \text{for backward search} \end{cases}$$

Theorem 3 ([72]**).** *If $X = 2n\delta(v)$ is an unbiased betweenness centrality estimator, then $E(X) = c_B(v)$.*

Hence, by averaging k independent runs of $X_i(v)$, an approximation $\bar{X}$ of $c_B(v)$ can be obtained. The authors of [72] proposed two implementations of their framework, namely a linear and bisection scaling. In the linear scaling, the contribution of the samples depends linearly on the distance to the sample, whereas in the bisection scaling, a sample only contributes on the second half of the path. According to the authors, both approaches perform better than the one proposed in [84], and the bisection approach even produced a good approximation for less important nodes with a small number of pivots.

Riondato et al. (2014). The authors of [85] proposed two efficient algorithms for betweenness centrality estimation, based on the random sampling of the shortest paths, which offer probabilistic guarantees on the quality of the approximation.

With a probability at least of $1 - \varphi$, both algorithms work as follows. The first algorithm estimates the betweenness of all vertices, and ensures that the approximate betweenness values are within an additive factor ε from the real values. The second algorithm focuses on the top-*K* vertices with the highest betweenness. It returns a superset of the top-*K* vertices, while ensuring that the approximate betweenness value is within a multiplicative factor ε from the real value. According to the authors, it is the first algorithm that can compute such approximation for the top-*K* vertices.

In order to derive the appropriate sample size necessary to achieve the desired approximation, Vapnik-Chernovenkis (VC) dimension theory [85] notions and results are used. A range set associated with the problem at hand is defined, and the upper and lower bounds to its VC-dimension are proven. So, the resulting sample size is independent from the number of vertices in the network, depending only on the vertex-diameter.

Let $\mathcal{I}_v(s,t) \subseteq \mathbb{S}_G$ be the set of all shortest paths, from s to t, that v is internal to

$$\mathcal{I}_v(s,t) = \{p \in S_{d_{st}} : v \in \mathrm{Int}(p)\}$$

The betweenness centrality of a vertex $v \in V$ in the normalized form is defined as

$$c_B(v) = \frac{1}{n(n-1)} \sum_{p_{st} \in \mathbb{S}_G} \frac{|\mathcal{I}_v(s,t)|}{\sigma_{st}}$$

Theorem 4 ([91,92]**).** *Let $\mathcal{R}$ be a range set on a domain D with $VC(\mathcal{R}) \le d$, and let ϕ be a distribution on D. Given $\varepsilon, \varphi \in (0,1)$ let S be a collection of $|S|$ points from D sampled according to ϕ, with*

$$|S| = \frac{c}{\varepsilon^2}\left(d + \ln\frac{1}{\delta}\right) \tag{8}$$

where c is an universal positive constant. Then, S is an ε-approximation to $(\mathcal{R}, \phi)$ with probability at least $1-\varphi$.

In order for the algorithm to compute a set of approximations for the betweenness centrality of the (top-K) vertices in a graph through sampling, with probabilistic guarantee on the quality of the approximations, let $d = \lfloor \log_2 VD(G) - 2 \rfloor + 1$. With (8), the resulting sample size r is

$$r = \frac{c}{\varepsilon^2}\left(\lfloor \log_2 VD(G) - 2 \rfloor + 1 + \ln\frac{1}{\varphi}\right) \tag{9}$$

Algorithm 2. *Repeat r times the following steps: First, sample a pair s, t of distinct vertices uniformly at random. Second, compute the set $S_{d_{st}}$ of all shortest paths between s and t. Third, select a path p from $S_{d_{st}}$ at random. Fourth, increase by $1/r$ the betweenness estimation of each vertex in* $\mathrm{Int}(p)$. *If the sampled vertices s and t are not connected, the third and fourth steps can be skipped.*

The unbiased estimator $\tilde{c}_B(w)$ for the betweenness $c_B(w)$ of a vertex w is the sample average

$$\tilde{c}_B(w) = \frac{1}{r} \sum_{p_{st} \in S} |\mathcal{I}_w(s,t)|$$

where S is the set of paths sampled in the algorithm. The desired accuracy and confidence are achieved with (9). For additional proof, please refer to [85].

4.2.2 Adaptive Sampling Techniques

The adaptive sampling technique was proposed in [86] for estimating the size of the transitive closure of a directed graph. The proposed algorithm presented adaptive sampling of source vertices. To be precise, the information acquired from each sample depends on the number of samples.

Bader et al. (2007). In [86], the authors proposed an approximate algorithm for computing betweenness centrality of a single vertex, for both weighted and unweighted graphs, which is based on an adaptive sampling technique. The proposed approximation is a sampling algorithm, since centrality is estimated by means of sampling and SSSP computations of a subset of vertices, and, it is an adaptive algorithm, since the information acquired from each sample depends on the number of samples. So, this approach significantly reduces the number of SSSP computations for high centrality vertices.

The authors noted that through scores' extrapolation from a fewer number of path computations, centrality can be estimated in contrast to [81] which estimates centrality scores of all vertices in the graph. Nonetheless, betweenness centrality scores are difficult to estimate, and the quality of the approximation was found to be dependent on the source vertices.

Let $a_i = \delta_{v_i*}(v)$ denote the dependency of the vertex v_i on v, and let $A = \sum a_i = c_B(v)$ denote the quantity to estimate.

Algorithm 3. *Repeatedly sample a vertex $v_i \in V$; using a graph traversal algorithm, do a SSSP from v_i and maintain a running sum χ of the dependency scores $\delta_{s*}(v)$. Sample until $\chi > cn$ (c is a constant and ≥ 2). If k is total number of samples, then the estimated betweenness centrality score of v is*

Table 4
A summary and comparison of the algorithms used to compute betweenness centrality.

Publication	Type of Algorithm (Section)	Main idea	Complexity
Brandes (2001) [81]	Exact computation (4.1)	Compute centrality of all graph's vertices in the same asymptotic time bounds as n SSSP computations.	$\mathcal{O}(nm)$ – unweighted graphs $\mathcal{O}(nm + n^2 \log n)$ – weighted graphs
Brandes and Pich (2007) [84]	Random Sampling (4.2.1)	The selection at random of source vertices is superior to deterministic strategies.	$\mathcal{O}(km)$ – unweighted graphs $\mathcal{O}(k(m + n \log n))$ – weighted graphs $k = \frac{\log n}{\epsilon^2}$
Geisberger et al. (2008) [72]	Random Sampling (4.2.1)	Obtain good betweenness centrality estimates of unimportant nodes.	
Bader et al. (2007) [86]	Adaptive Sampling (4.2.2)	Reduce the number of SSSP computations of vertices with high centrality.	
Riondato et al. (2014) [85]	Random Sampling (4.2.1)	Compute betweenness estimation, based on the random sampling of the shortest paths, which offer a probabilistic guarantees on the quality of the approximation.	$\mathcal{O}(\mathrm{r}(\mathrm{n} + \mathrm{m}))$ – unweighted graphs $\mathcal{O}(\mathrm{r}(\mathrm{m} + \mathrm{n} \log n))$ – weighted graphs
Hinne (2011) [87]	Local Techniques (4.2.3)	Gives a local centrality estimate based on a subgraph of vertices around a specific vertex.	$\mathcal{O}(nk^{2d}), k^{2d} < m$ where k is the average degree, and d denotes distance

$$c_B(v) = \frac{n\chi}{k}$$

Theorem 5 ([86]). *For $0 < \epsilon < 0.5$, if the centrality of a vertex v is n^2/γ for some constant $\gamma \geq 1$, then with probability greater or equal to $1 - 2\epsilon$ its centrality can be estimated to within a factor of $1/\epsilon$ with $\epsilon\gamma$ samples of source vertices.*

The authors of [86] demonstrated through experimental evaluation that their algorithm performed similarly for other vertices, besides those with high centrality (showed from theoretical results). For detailed discussion of the algorithm, please refer to [86].

4.2.3 Local Techniques

Hinne (2011). The authors of [87] proposed a local strategy to derive an approximation and the corresponding error bound's analysis of the true centrality metric using only the vertices directly adjacent to a target vertex. For example, the estimation of the vertex's centrality could be obtained by examining the vertex, its neighbors and its neighbor's neighbors.

Let the normalized version of (1) be defined as

$$c_B(v) = \frac{1}{(n-1)(n-2)} \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{10}$$

To obtain a local approximation of (10), the term $\sigma_{st}(v)$ can be decomposed as $\sigma_{st}(v) = \sigma_{sv}\, \sigma_{vt}$. By applying this decomposition to the summation terms in (10),

$$\frac{\sigma_{st}(v)}{\sigma_{st}} \propto \sum_{\substack{v_0 \to v \to v_1 \\ v_0 \neq v_1}} \frac{\sigma_{v_0 v_1}(v)}{\sigma_{v_0 v_1}}$$

where v_0 and v_1 are the predecessors and successors of v, respectively. So,

$$c_B(v) \propto \tilde{c}_B^H(v) = \sum_{\substack{v_0 \to v \to v_1 \\ v_0 \neq v_1}} \frac{\sigma_{v_0 v_1}(v)}{\sigma_{v_0 v_1}}$$

where $\tilde{c}_B^H(v)$ is the local approximation of betweenness centrality, and H is the local subgraph of v. The approximation error is given by

$$\left| c_B(v) - \tilde{c}_B^H(v) \right| = \sum_{s \nrightarrow v \nrightarrow t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Since local subgraphs are smaller than the graph itself, there is a trade-off between computation and accuracy in local approximations. Specifically, smaller subgraphs lead to faster computations, decreasing as a result the approximation accuracy.

Table 4 presents a summary and comparison of the algorithms used to compute betweenness centrality based on the type of algorithm, the main idea and their complexity.

5 DTN Routing Protocols

DTNs routing protocols face a troublesome task of finding a suitable next node to forward messages, due to the network's dynamics. This problem is augmented when additional requirements, such as good delivery probability or low end-to-end delay, are foreseen. Many routing protocols have been proposed up until now, and in some social network analysis is leveraged to enhance the delivery of messages. Some of the routing protocols hereby presented make use of many social metrics, where at least one is betweenness centrality. Let it be noted that betweenness centrality is computed over some inherent social network (see column 3 of Table 5), and not directly over the DTN.

5.1 SimBet

The authors of [22] proposed a DTN routing protocol called *SimBet* that exploits two social metrics for data forwarding, namely betweenness centrality and social similarity[8]. If neither the sender nor its contacts know how to reach the destination node, the message is forwarded to a node structurally more central as its odds of discovering a suitable carrier are higher. Unlike previous works, no assumptions of control of node movements, or knowledge of future movements is made. It is assumed that only a single copy of each message exists in the network, which reduces resource consumption if compared to multi-copy strategies.

The authors argued that metrics based on direct or indirect encounters were not appropriate for discovering suitable carriers for routing messages, since some networks contained cliques (i.e., groups of nodes – clusters – that interacted more among them than with members of other clusters). So, node's centrality was estimated in the network in order to identify bridge nodes, i.e., message carriers among disconnected groups. Based on concepts from graph theory and network analysis, centrality is used to quantify a vertex's importance within graphs.

As previously stated, betweenness centrality measures the extent to which a node has control over information flowing between others. Thus, high betweenness centrality nodes are regarded as having a capacity to facilitate interactions between nodes they link, i.e., having a capacity of facilitating communication to other nodes in the network. It was also previously referred that a well-known disadvantage of centrality metrics is their computational complexity for large networks. Due to this, the authors used *ego networks* which do not require complete knowledge of the network as ego network analysis is performed locally by each individual node.

It was shown in [93] that betweenness centrality based on egocentric measures is not equivalent to its sociocentric counterpart, despite the node's ranking based on both metrics being identical. Consequently, a comparison of locally calculated betweenness values between two nodes can be made, and the one with the higher value may be found. The betweenness values show 'how much a node connects nodes that are themselves not directly connected'. SimBet calculates betweenness centrality using an egocentric network representation of nodes with which the ego node has come into contact. Therefore, egocentric betweenness centrality is given by equation (3).

5.2 SimBetAge

In [23], the authors proposed a DTN routing protocol for highly dynamic socially structured networks called *SimBetAge*. The routing scheme proposed in [22] was exploited while simultaneously taking into account social relations' strengths and gradual aging, i.e., the progression of the social network over time. Similarity and betweenness centrality were modeled over weighted graphs instead of unweighted ones as in [22].

A more realistic view of a social network is modeled by a weighted time dependent graph $G(T) = (V, E, \omega(e, T))$, where $G(T)$ is a fully connected graph, the weight ω is called *freshness* of an edge. If $e = (u, v)$ and $T \in \mathbb{T}$, $\omega(e, T) = 0$ means that the nodes in e have never been connected; $\omega(e, T) = 1$ means a permanent connection between them. The freshness of a single edge can be perceived as an indicator of the likelihood of two nodes u and v being connected at a given time T, due to its representation as a logistic growth function (i.e., the contacts become fresher at each new encounter), and exponential decay function (i.e., the contacts become older with time). The freshness of a path $\omega(P, T)$ is defined by the product of all freshness values in it.

The authors proposed a new metric called *egocentric flow betweenness* defined as

$$c_{EFB}(v) = \sum_{u,w \in V^*(v)} \frac{(\omega_{vu}\ \omega_{vw})^2}{\omega_{vu} + \sum_{u \neq v \in V^*(v)} \sum_{w \neq v \in V^*(v)} \omega_{vu}\ \omega_{vw}},$$

[8] Similarity expresses the amount of common features of a group in social networks. In sociology, the probability of two individuals being acquainted increases with the number of common acquaintances between them.

i.e., $c_{EFB}(v)$ is the sum over the age of all paths between pairs of nodes $u, w \in V^*(v)$ passing through u divided by the age of all possible paths between them, and weighted with the age of the edges between u and v, and u and w. If two nodes u and w want to compare their utilities to a destination t in an ego-centric manner:

- *Betweenness* should be used, if both nodes are far away from t;
- *Similarity* should be used, if t is at most two hops away of any of them;
- *Directed betweenness* should be used otherwise, as it considers only the paths containing t instead of all possible paths in the neighborhood of v. Directed betweenness is defined as

$$c_{DB}(v,t) = \sum_{\substack{u,w \in V^*(v) \\ t \in V^*(v) \vee u=t}} \frac{(\omega_{vu}\,\omega_{vw})^2}{\omega_{vu} + \sum_{u \neq v \in V^*(v)} \sum_{w \neq v \in V^*(v)} \omega_{vu}\,\omega_{vw}}$$

5.3 *Bubble Rap*

In [24], a DTN routing protocol for pocket switched networks (PSN) called *Bubble Rap* was proposed. A PSN is a network without infrastructure composed of a multitude of devices carried by persons. Therefore, two social metrics, namely community and centrality, are exploited for data forwarding, instead of mobility due to the network's unpredictability and highly dynamic topological structure. Node mobility is used by MANET and DTN routing algorithms to build and update their routing tables. According to sociology, a community can be defined by a group of people living in the same location. So, people from the same community tend to interact more often between themselves, than with a randomly chosen member of the population [94]. Another important aspect to consider within a community is the degree of interaction among its members. Usually, some members tend to interact more than others. For example, the postman meets customers more often, in comparison to a network engineer. As a consequence, there are more popular members (hubs) which have higher centrality values, and these popular hubs are a better choice for relays than unpopular ones.

In [24], the authors assumed that each node belonged to at least one community, and that it had a global (for the whole system) and local (for its local community) centrality values. Also, a node could belong to a single node community or to multiple communities, thus having multiple local centrality values.

A PSN is modeled as a temporal network (or a time evolving network) due to its characteristics. Thus, betweenness centrality of a node in a temporal graph is obtained by counting the number of times a node acted as a relay for other nodes on all the shortest delay deliveries[9], over a large number of emulations of unlimited flooding with different uniformly distributed traffic patterns.

In order to approximate centrality, the authors found out that the degree per unit-time (e.g., the number of unique nodes seen per t hours) and the node centrality had a high correlation value (for t = 6, the correlation coefficient is 0.9511). This correlation led them to conclude that what mattered was the frequency of interaction, not the number of known persons. They compared the average unit-time degree with a greedy ranking algorithm called RANK, and found out that they performed similarly. RANK, which is similar to the greedy strategy in [95], assumes that each node only knows its ranking and the ranking of those it encounters. But, it does not know the ranking of the other nodes it does not encounter, neither does it know which node has the highest rank in the system. However, since in a distributed manner, it is difficult to compute the average unit-time degree individually throughout the whole experiment, two approaches were proposed, namely: (1) the single window (S-Window) approach, in which upon an encounter, nodes compare how many unique nodes they met in the previous unit-time slot; (2) the cumulative window (C-Window) approach, which consists in calculating the average value on all previous windows (e.g., from yesterday until now), and then calculating the average degree for every t hours. C-Window is similar to a statistical technique called exponential smoothing [96].

The authors used two centralized community detection algorithms, namely *K*-CLIQUE [97] and weighted network analysis (WNA) [98], to identify local community structures as it would be helpful in designing good strategies for information dissemination. They use the two since each has useful features and they complement each other.

Bubble Rap Forwarding works as follows: if a node wants to send a message to a destination, this node first bubbles (forwards) this message up based on the global centrality until it finds a node which is in the same local community as the destination of the message. Then the local centrality is used instead of the global one, and the node continues to bubble up the message based on the local centrality through the local community until the destination is found or the message expires.

5.4 *PQBCF*

The authors of [25] proposed a peer-to-peer (P2P) query algorithm based on betweenness centrality forwarding (PQBCF) for Social Opportunistic Networks (SONs). In SONs, mobile devices are carried by people and consequently the mobility model exhibits social characteristics. Since betweenness centrality quantifies the importance of nodes in message delivery throughout the network, nodes with higher betweenness values can be seen as more active and having more opportunities of encountering more nodes. So, they are naturally good candidates to act as relay nodes.

[9] The delivery with shortest delay when the same message is delivered to the destination over different paths.

To share or publish contents in SONs, a message dissemination scheme commonly used is the P2P inquiry/response[10] that works as follows: first, a query node sends an inquiry message throughout the network searching for a node with the response message; then, the response message is forwarded to the query node.

In PQBCF, a node looking for some data, e.g., an audio file, generates an inquiry message containing the description of the required data. To reduce the expected query latency, assuming that many nodes contain the requested data, multiple copies of the inquiry message are created. With more copies, the query delay reduces, but the network overhead also increases as a result of the additional number of message copies occupying nodes' buffers, as well as more message transmissions. Therefore, a tradeoff between the query delay and overhead should be obtained by the inquiry/response scheme. PQBCF achieved this by calculating the number of copies of the inquiry message in the network based on the expected query delay, the mobility and the nodes density. During the inquiry message dissemination, betweenness centrality is used as the metric for relays selection. If the inquiry message is passed to a node with information matching the inquiry message, it creates a response message, calculates the number of copies of the response message, and sends it back to the inquiry node.

The betweenness centrality of a node v is defined as

$$c_B(v) = \frac{2}{(n-1)(n-2)} \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{g_{st}(v)}{g_{st}}$$

where g_{st} is the number of all the messages successfully delivered between a pair of nodes s and t, and $g_{st}(v)$ represents the number of messages that passed through node v during the forwarding process. The ratio $g_{st}(v)/g_{st}$ indicates the importance of node v in delivering messages between s and t. If the destination node is fixed, the importance of node v in delivering messages to the destination node t is given by

$$c_{B_t}(v) = \frac{2}{(n-1)} \sum_{s \neq v \in V} \frac{g_{st}(v)}{g_{st}}$$

Each node maintains a table with the necessary information to calculate $c_B(v)$ and $c_{B_t}(v)$ which can be obtained in a distributed manner.

5.5 GrAnt

Multi-agent systems in which the behavior of a single agent (also known as artificial ant) is inspired by the behavior of real ants, are called ant systems. The Ant Colony Optimization (ACO) metaheuristic [99], a particular class of ant algorithms which use artificial swarm intelligence [100], is inspired on an experience by Gross et al. [101] using an ant system. Some example problems where ant algorithms have been used are classical traveling salesman and routing in telecommunications networks.

In [18], the authors proposed Greedy Ant (GrAnt), a prediction-based routing protocol for DTNs, which uses a greedy transition rule of the ACO metaheuristic aiming at exploiting, if available, good previous solutions, and to select the most suitable message forwarder.

To cope with DTN, the following modifications were proposed allowing to differentiate GrAnt from traditional ACO algorithms: (1) to increase the possibility of reaching the destination, forwarder ants, which are ant agents responsible for discovering paths to the destination nodes, are encapsulated into data messages; (2) to find a path to an unknown destination, a dynamic number of forwarder ants, whose computation takes into account the utilities of the already established message forwarders and the success of the message delivery, is used; (3) to provide exploitation of good solutions already found or to forward the message to the most promising node, a greedy ACO transition rule is used while considering heuristic functions and pheromone concentration. The pheromone concentration indicates how useful a global solution was, which serves as a history of the best previous movements of the ants. The heuristic function values indicate an explicit influence towards more useful local information; (4) besides the best path, redundant ones are also allowed due to the dynamics of DTNs. An event-driven evaporation happens if and only if a node detects a new path being constructed to the destination. And, since the pheromone deposited by ants is based on information about nodes in each constructed path, the evaporation process prevents the occurrence of undue convergence of the algorithm to the same subset of paths.

The GrAnt protocol provides modules for (*i*) routing, by determining which route a message should follow to reach its destination. The forwarding decision consists in adopting a greedy transition rule that considers the pheromone at a link in the path to the destination (or local heuristic information, in the absence of pheromone) and the heuristic function associated with an intermediate node in the path to the destination, (*ii*) scheduling, by deciding in which order messages must be transmitted, and (*iii*) buffer management, by indicating which message(s) must be dropped whenever the buffers occupancy limit has been reached.

The heuristic function is based on two criteria: Social_{vt}, representing the social proximity between nodes v and t, and BetwU_{vt}, representing the betweenness utility of node v in relation to the destination t. The node betweenness utility computation is slightly

[10] In [119], an inquiry/response scheme was proposed combining content-based routing with probabilistic-based routing.

different from [3]. In order to have a high betweenness utility to a destination t, a node v must appear with high frequency in paths between any source node and the destination t. So, differently from [3] and [22], no shortest path verification is required by betweenness utility and no list of all previous encounters is exchanged, respectively.

5.6 Kim et al. (2014)

In [28], the authors proposed a routing scheme by using DNI (i.e., node's local contact history) and SNI (i.e., the expanded ego-network betweenness centrality). Since computing the real betweenness involves global network knowledge it is in general impractical in DTNs due to the lack of network-wide end-to-end connectivity. Therefore, each node computes betweenness by means of its local expanded ego network built using the node's social network composed of information of its neighbors and of its neighbors' neighbors. The result is used as an estimate of its true betweenness over the entire network due to their high correlation [102].

Routing is composed of two strategies, namely, edge weight and centrality based strategies. In the former, each node calculates the edge weight [103] using DNI. If the edge weight is high between a pair of nodes it means that there is a high future contact opportunity. Similarly to [103], a node carrying a message to a remote destination forwards it to a given relay node if the edge weight between the relay and the remote destination is higher than the one between him and the destination node. The latter is used to improve routing efficiency. Each node constructs its own social network, and calculates the expanded ego betweenness centrality. As some nodes might get very low edge weights since they hardly meet with other nodes, and if these isolated nodes are the destination, proper relays might be difficult or even impossible to find by the source node using the former strategy. This message would probably be discarded due to Time-To-Live (TTL) expiration. Therefore, the node carrying the message also forwards the message to another node if it presents a higher value of betweenness centrality even though it presents a lower edge weight value, since nodes with higher betweenness centrality values are more socially related to other nodes.

Additionally, a message management scheme was also proposed to reduce the overall delivery cost. A node carrying a message can delete the message from its buffer after forwarding the message to another node with an edge weight higher than all edge weights in its social network.

5.7 LocalCom

Previous works [22,24,104] confirmed that with high probability nodes in DTNs tend to meet more a certain group of nodes than other nodes outside this group, and that the grouping structure remains stable over time. Hence, it is of interest to utilize the grouping structure of DTNs to facilitate message forwarding.

In [26], the authors proposed LocalCom, a community-based epidemic forwarding scheme for routing, which efficiently detects the community structure, using limited information, and improves the forwarding efficiency based on the community structure. In LocalCom, the statistics of the separation period is selected in order to shorten nodes' knowledge. Based on the frequency and length of the node's contacts, each node calculates the average separation period towards its neighbors. It also applies the Gaussian similarity function [105] to represent the closeness in the relationship. A closer relationship is reflected by a shorter average separation period. At the same time, an irregularity in the relationship is reflected by the variance of the separation period. Therefore, closeness and irregularity metrics are used to deduce the similarity metric, which shows the relationship between each pair of nodes in the network. Similarity also captures the core temporal and spatial encounter information.

Differently from [22][24], a distributed scheme was developed and it only requires local information to form communities in LocalCom. It uses an extended clique, which is based on virtual links, to represent underlying community structures. A virtual link allows the representation of a neighboring relationship between a pair of nodes, if at least one path with up to k hops exists between them.

High similarity and short hop-count distances that are some of the desirable properties within a community can aid intra-community communication based on the single-copy source routing. So, packets will be directly forwarded along a virtual link. Through flooding, inter-community packet forwarding is performed using nodes that have direct neighboring relationship with nodes in other communities (also called gateways). Since not all gateways are necessary, some pruning is performed to avoid unnecessary redundancy. Bridges, which are the actual forwarding nodes are selected from gateways using two marking and pruning schemes: static pre-pruning and dynamic pruning. The former is conducted by each gateway based on local information. Nodes marked as bridges during the former further define their role dynamically based on additional information received.

In order to forward a packet between nodes residing in different communities: first, the inter-community forwarding mechanism is used to forward the packet to the current communities' bridges, and then, the bridges forward the packet to other communities they are connected to. Each gateway calculates its centrality for the communities it connects. The betweenness centrality of a gateway v in community A connecting community B is given by

$$c_{GB}(v) = \sum_{s \in A} \sum_{t \in B} \frac{\sum_{p \in P_{st}(v)} \left(\prod_{(i,j) \in p} \omega_{ij} \right)}{\sum_{p \in P_{st}} \left(\prod_{(i,j) \in p} \omega_{ij} \right)}$$

where P_{st} represent the set of all paths between s and t in the neighboring graph, and $P_{st}(v)$ denotes the subset of P_{st} containing all the paths from s to t that pass through v. The numerator and denominator are the sum of all path weights in $P_{st}(v)$ and the sum of all path weights in P_{st}, respectively.

Each gateway should calculate its centrality values, that is, one distinct for each community it connects, and send through the virtual links among them the centrality value to all other nodes in its local community. Therefore, each gateway knows all other gateways in its community connecting to other communities, also knowing their centrality values.

5.8 CAOR

The authors of [31] proposed the community-aware opportunistic routing (CAOR) algorithm for Mobile Social Networks (MSNs) [106] using two social metrics, namely community and centrality. A MSN can be seen as a social DTN since it is composed of mobile nodes with social characteristics[11]. Based on this social characteristic, a home-aware community model was proposed in which mobile users with a common interest form, by themselves, a community where the frequently visited location is their common *home*. Similarly to [106], the authors assume that each home supports a real or virtual throwbox [107], i.e., a local device that can temporarily store and transmit messages.

The rationale behind CAOR is to turn the routing between lots of mobile nodes to the routing between a few community homes. Therefore, message delivery can be turned into the delivery within and between these communities. Two centrality metrics are used to measure the importance of nodes during message delivery, specifically: intra-community centrality and inter-community betweenness metric. The former consists in measuring the capability of each community member to meet and deliver messages to other members, and the node with the largest intra-community centrality in a community has the best capability to deliver messages. The latter consists in measuring the ability of a node set to be taken as a communication bridge between communities. Here, the delivery delay is used to evaluate the inter-community betweenness of a set of nodes.

Let an MSN be composed of $|V|$ nodes $V = \{v|v \in V\}$ moving among $|L|$ locations $L = \{l|l \in L\}$ such that $(|L| \ll |V|)$. For two overlapped communities C_l and $C_{l'}$ and an arbitrary relay set $S(S \subseteq C_l \cap C_{l'})$, the inter-community betweenness is given by

$$c_{B_{l,l'}}(S) = \frac{1}{\sum_{v \in S} \lambda_{v,l}} + \frac{\sum_{v \in S} \lambda_{v,l}/\lambda_{v,l'}}{\sum_{s \in S} \lambda_{v,l}}$$

where $\lambda_{v,l}$ and $\lambda_{v,l'}$ are parameters of the exponential distribution followed by the interval of node v's visits to homes l and l', respectively. In other words, the inter-community betweenness is the expected delay that it takes for a relay node to cooperatively deliver messages by means of an opportunistic routing scheme from one community to another.

The optimal betweenness, which corresponds to the relay set with the smallest betweenness for the message delivery from the community home l to l', is given by

$$c_{\tilde{S}_{l,l'}} = \underset{S \subseteq C_l \cap C_{l'}}{\operatorname{argmin}} c_{B_{l,l'}}(S)$$

The CAOR algorithm consists of an initialization and routing phases. The initialization phase builds $|L|$ community homes from a network with $|V|$ nodes, thus simplifying the network. Then, under the home-aware community model, the routing phase delivers messages based on the optimal opportunistic routing rule, that is, the message sender always delivers messages to the encountered relay that has a smaller minimum expected delay to the destination than itself.

5.9 Hoten

In [33], a forwarding metric, known as Hoten (HOTspot ENtropy), which consists of three social metrics, namely betweenness centrality, similarity and personality, was proposed to improve the performance of routing in opportunistic networks. The authors focused on the integration of social structure into data forwarding algorithms since existing algorithms, such as SimBet, Bubble Rap and People Rank [108], did not fully exploit social structures extracted from real world traces (e.g. human walks [109]). Similarly to [110], the authors confirmed the existence of two known phenomena by analyzing GPS traces of human walks, i.e., on the one hand, people always move around a set of well-known locations, called *public hotspots* (instead of purely random walks), and, on the other hand, each people shows preference for some particular locations, called *personal hotspots*. They also assumed hotspots were more stable than the social structure of existing algorithms, as for example, public hotspots were formed by overlaying personal hotspots together and personal habits were stable over time and across situations [111].

Information theory [112] is used to compute the nodes' social metrics since the entropy represents the degree of disorder or randomness in a system, i.e., the bigger the entropy value is, the more disordered the system is. To compute betweenness centrality, the authors used the relative entropy[12] [113] (also called Kullback-Leibler divergence) between the public hotspots and the personal

[11] For instance, in many real MSNs, mobile users with common interests tend to visit some location (real or virtual) that is related to this interest.

[12] Relative entropy can be used to differentiate the divergence between two random variables [33].

hotspots. Similarity between two nodes was computed by exploiting the inverse symmetrized entropy of the personal hotspots between the nodes. The entropy of personal hotspots of a node is used to estimate its personality.

Let K denote the total number of hotspots in the network and let n_i denote the number of stay points in hotspot i. The weight of the hotspot i is given by $\omega_i = n_i / \sum_{i=1}^{K} n_i$. In the same way, let $n_{\mathrm{p}_i}^j$ denote the number of i^{th} person's stay points in j^{th} hotspot. The weight of j^{th} hotspot influenced by the i^{th} person is given by $\omega_{\mathrm{p}_i}^j = n_{\mathrm{p}_i}^j / \sum_{i=1}^{K} n_{\mathrm{p}_i}^j$.

Let X_i be a random variable denoting the distribution of personal hotspots of node i, and let Y be a random variable denoting the distribution of public hotspots. So, Y = $\omega_1, \omega_2, \dots, \omega_k$ and $X_i = \omega_{\mathrm{p}_i}^1, \omega_{\mathrm{p}_i}^2, \dots, \omega_{\mathrm{p}_i}^k$. The betweenness centrality of node v is given by

$$c_B(v) = \left[\sum_{j=1}^{k} \omega_{\mathrm{p}_v}^j \log\left(\frac{\omega_{\mathrm{p}_v}^j}{\omega_j} \right) \right]^{-1} \tag{11}$$

If equation (11) is compared with equations (1) and (3), one can conclude that it has low time complexity $\mathcal{O}(k)$ since it (*i*) is only related to the top k hotspots and (*ii*) is independent of the number of nodes in the network.

The Hoten routing algorithm works as follows: when a node meets with another node, the node delivers to the other node any message it carries destined to the other node, and removes the message from its messages' queue. If a message is not destined to the other node, both nodes swap their Hoten forwarding metrics (also known as Hoten utility) for that message. If the node's Hoten utility is smaller than that of the other node for the given message, the node delivers the message to the other node and removes the message from its message queue, thus taking a single copy approach.

Table 5 presents a summary and comparison of DTN Routing protocols using betweenness centrality based on the social metrics used, the type of graph used, the main idea of the routing protocol, the type of standard betweenness centrality algorithm, optimizations, the performance evaluation and DTN scenarios and/or applications. Note that the performance evaluation presented, when available, is limited to the surveyed social routing protocols.

6 Discussion

In self-organizing networks, such as DTNs, network's dynamics poses a challenging task to routing protocols and as a result of that end-to-end connectivity between any pairs of nodes might never exist. However, by using a store-carry-and-forward approach, DTN nodes can carry messages with them while moving until an appropriate node is found. In this approach, messages are relayed from one node into another until they reach their destination, or they are discarded. In order to find the most suitable forwarding node, static and dynamic network information is used. Among the available network information, static network information has been adopted by a considerable number of social routing protocols due to its stability tendency over time, hence leveraging the use of social metrics. Still, despite the advantages of using social metrics, single-property social routing protocols may experience difficulties finding the destination node. If, for example, centrality-based metrics are being used, the node carrying a message may not select, as the next message carrier intermediate nodes having lower centrality than the current carrier. But, depending on the network topology, intermediate nodes with low centrality may also have high odds of encountering the destination node. This led most of the surveyed DTN routing protocols to be hybrid, although some properties might be non-social.

In this work, a survey of betweenness centrality concepts, variants and standard algorithms is presented. Additionally, a survey of DTN routing protocols that use betweenness centrality, and a discussion on how the metric, its algorithms are used by the protocols is also provided. Previous work has shown that centrality metrics, which are used to point out the (relative) importance of vertices and edges in networks, are of considerable relevance for DTN routing protocols. Since mathematically these metrics are simple to grasp, their actual calculation is by far much more elaborate, due to the network's size and dynamics. Because of that approximate algorithms are more common means of calculation as an alternative to the exact computation.

The surveyed protocols can be organized in three groups based on the type of algorithms, namely: (*i*) approximate algorithms, (*ii*) exact algorithms, and (*iii*) alternative heuristics.

Ego networks, which fit in the first group, are used to reduce the complexity associated with the computation of betweenness centrality using partial network knowledge. Since ego network analysis is performed locally by each individual node, egocentric betweenness centrality can be seen as a local technique similar to the one described in [87]. In [28], the authors used an expanded ego network, i.e., an ego network where the second degree neighbors of a given node are also considered. In [22], betweenness centrality is only updated upon hello message reception from new nodes.

Table 5
A summary and comparison of DTN Routing protocols using betweenness centrality.

Publication	Social metrics	Graph Type	Main idea	Type of algorithm	Optimizations	Performance evaluation	Scenarios/ Applications
SimBet [22]	Egocentric betweenness and Similarity	Unweighted graph	The message is forwarded to a node structurally more central.	Local Technique	NA	Delivery performance close to Epidemic [120], but without the overhead.	Disconnected Delay-Tolerant MANETs
SimBetAge [23]	Egocentric flow betweenness and Similarity	Aged Graph	It is an extension of SimBet that takes into account the progression of the social network over time.	Local Technique	NA	It outperforms SimBet in terms of delivery rate.	PSNs
Bubble Rap [24]	Betweenness centrality (degree centrality per unit time) and Community	Weighted Temporal Graph	Nodes bubble up messages first using global centrality and then using local centrality.	NA	Controlled message replication. Original carrier deletes the message once the destination community is identified.	Delivery ratio close to SimBet, but much lower resource utilization.	PSNs
PQBCF [25]	Betweenness centrality	NA	A query node sends an inquiry message throughout the network searching for a node with the response message using betweenness centrality as the metric for relay's selection.	NA	The number of inquiry messages is a tradeoff between the query delay and overhead.	Inquiry success ratio and delay better than flooding for high message generation frequency. No social routing protocol was considered.	SONs
GrAnt [18]	Betweenness utility	NA	The next node is chosen using pheromone concentration if available, or local information captured from DTN nodes.	NA	NA	Achieves higher successfully message delivery and lower overhead than Epidemic in community-based movement model. No social routing protocol was considered.	DTNs
LocalCom [26]	Community, similarity and betweenness centrality	Neighboring Graph	The intra-community forwarding mechanism is first used to forward the packet to the current communities' bridges, and then, the bridges forward the packet to other communities they are connected to.	Exact Computation	Community level broadcast if the source and destination are in different communities.	Being simple flooding the upper bound in terms of the delivery ratio, LocalCom outperforms other protocols, (Bubble Rap included). But in terms of number of forwards (overhead), Bubble Rap represents the lower bound for all the scenarios considered.	DTNs
Kim et al. [28]	Expanded ego betweenness centrality	Weighted Contact Graph	Each node first uses DNI to choose a proper relay node. Then, SNI is used to enhance routing efficiency.	Local Technique	Message delivery cost is reduced by deleting messages forwarded to nodes with the highest edge weight.	More delivery efficient routing in comparison to Epidemic.	DTNs
COAR [31]	Betweenness centrality and community	Contact Graph	Build home-aware communities and use optimal opportunistic routing rule to route messages among these communities.	NA	Each home only forwards its messages to the node in its optimal relay set.	It outperforms Bubble Rap and SimBet in terms of delivery rate and average delay.	MSNs
Hoten [33]	Betweenness centrality, similarity and personality	NA	Each node only forwards a message if the Hoten utility for a given destination is smaller.	NA	Single-copy approach	It outperforms SimBet and PeopleRank in terms of delivery rate.	DTNs

The two betweenness centrality metrics (egocentric flow betweenness and directed betweenness) proposed in [23] were envisaged for highly dynamic social networks, as instead of the number of shortest paths, their calculation takes into account all possible paths in a network. The difference between them is that in the latter only paths containing the destination are considered. Also, since both metrics consider nodes in the neighborhood of a given node, these algorithms are based on a local technique. A

Socially-Aware Multi-Phase Opportunistic (SAMPhO) [32] routing protocol was proposed, in which ego betweenness is used according to the conditions of the social environment in the centrality-based forwarding phase. The authors of [34] proposed two distributed EBC protocols (EBC broadcast and gossip) for distributed SONs. EBC broadcast and gossip differ from each other in the update phase of the adjacency matrix. In the former, the adjacency matrix is kept updated by each node, by doing communications with all nodes in its ego network. In the latter, the adjacency matrix is kept updated through specific gossip techniques [114]. A bridging centrality metric [115] that is calculated by multiplying betweenness centrality by a bridging coefficient [115] is used in [35]. But, instead of using the shortest-path version that requires global network knowledge, ego betweenness centrality was used.

The routing proposed in [26] is the only one using an exact algorithm (hence, belonging to the second group) for the betweenness centrality computation on a weighted graph. This routing protocol uses three social metrics, namely similarity, community, and betweenness centrality. Similarity, which is based on closeness and irregularity metrics, is used to build the neighboring graph. With the graph, a distributed scheme is used to identify communities which are used during the intra-community forwarding. If the source and destination nodes are in different communities, flooding is used for intra-community forwarding and betweenness centrality for inter-community through bridges.

Some of the routing protocols proposed use alternative heuristics to compute betweenness centrality. In [25], betweenness centrality is computed using the number of successfully delivered messages. No global network knowledge is necessary as when nodes meet, they synchronize their reserved ratios of successful message delivery values and update their betweenness centrality values to a given destination. In [18] and [30] a metric called betweenness utility was proposed to measure the importance of a given node in delivering messages to a certain destination node. Differently from [3], no shortest path verification is required by the betweenness utility, nor a list of all previous encounters is exchanged, in contrast to [22]. The hybrid protocol proposed in [30] infers the most suitable next node to forward messages by means of opportunistic social information. It also determines the best path to forward each message while limiting message byte redundancy. In [24], the authors approximate centrality using the degree per unit-time, as the two metrics are highly correlated. In [29], a centrality metric was proposed that uses the expected number of packets which can be transmitted from a given node to others within the time constraint, as the centrality metric in [24] only considers the frequency of contacts and disregards their duration. Equally, in [27] a generalized model of the centrality metric was proposed that allows the calculation of the expected delivery performance metrics (delivery latency or delivery cost) of a given message. In [31], the expected delivery delay is used evaluate the inter-community betweenness of a set of nodes, and in [33] relative entropy, from information theory, is used to compute betweenness centrality.

Previously, six definitions of betweenness centrality were presented. They can be used in static or dynamic networks, and use partial or global network information. The most appropriate centrality metrics for DTNs are flow, random-walks, ego and temporal betweenness centrality, since they do not require global knowledge of the network (see Table 2). Among them, flow and ego betweenness have been implemented in DTN routing protocols, as shown in Table 5. Please note that SimBetAge uses an egocentric version of flow betweenness centrality, thus not requiring global network knowledge. Most of the works analyzed compare their approach with Epidemic routing, a non-social routing approach, as it can be seen as the upper bound in terms of delivery ratio. While designing routing protocols, there is always a tradeoff between delivery ratio and overhead. For example, on the one hand, there is LocalCom that outperforms Bubble Rap in terms of delivery ratio by using flooding to increase its probability of successful packet delivery, and the schemes considered could only achieve a delivery ratio of 30% to 40% when the expiration TTL was set to three days in the Reality scenario (MIT Reality Mining [116]) [26]. CAOR significantly outperforms SimBet and Bubble Rap in another scenario (MSN trace from the WiFi campus of Dartmouth College [117]) in terms of delivery ratio and average delay (which was not considered as evaluation metric in [26]). Specifically, when compared with SimBet and Bubble Rap, CAOR increases the delivery ratio by about 89.5% and 35.8%, and reduces the delivery delay by about 49.6% and 22.7%, respectively [31]. On the other hand, there is Bubble Rap that because of using a more conservative replication strategy presents a lower overhead in comparison to LocalCom [26]. But both use betweenness centrality during forwarding as means to reach the destination. Still, the message delivery efficiency, used by Kim et al. [28], incorporates both message delivery ratio and overhead, and because of that can be seen as a good indicator of a protocol overall performance.

In addition, another important aspect that was taken into account in SimBetAge was the fact that social relations and the roles of individual nodes change over time. Likewise, some relations are stronger than others resulting from, for example, a higher contact frequency.

Despite the efforts of innumerous researchers, the use of social metrics by DTN routing in still under research. An interesting point of research is for DTN routing protocols to consider in addition to the shortest paths, the fastest ones in terms of end-to-end duration. This concept is similar to the one defined in the temporal betweenness centrality.

As previously mentioned, betweenness centrality has shown its relevance to problems such as identifying important nodes that control flows of information between separate parts of a network and identifying casual nodes to influence other entities behavior. It has been also used to analyze social and protein networks, to identify and analyze behavior of key bloggers in dynamic networks of blog posts, to identify significant nodes in wireless ad hoc networks, to study online expertise sharing communities, to study the importance and activity of nodes in mobile phone call networks and interaction patterns of players on massively multiplayer online games and to measure network traffic in communication networks. In relation to DTNs, many social routing protocols that use betweenness centrality have been proposed to enhance routing in PSNs, SONs, MSNs, Disconnected Delay-Tolerant MANETs and so on. With the exception of GrAnt and cGrAnt [30], the remaining surveyed routing protocols exploit social characteristics of mobile nodes in those networks. Additionally, betweenness centrality in DTNs has shown its relevance to problems such as the construction of a mobile backbone, the offloading of data in wireless social mobile networks, and information dissemination and content placement in opportunistic networks.

Acknowledgement

This work was partially supported by Fundação Calouste Gulbenkian and by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013.

References

[1] M.M.J.M. Khabbaz, C.M.C. Assi, W.F.W. Fawaz, Disruption-Tolerant Networking: A Comprehensive Survey on Recent Developments and Persisting Challenges, Commun. Surv. Tutorials, IEEE. 14 (2012) 607–640. doi:10.1109/SURV.2011.041911.00093.

[2] K. Wei, X. Liang, K. Xu, A survey of social-aware routing protocols in delay tolerant networks: applications, taxonomy and design-related issues, Commun. Surv. Tutorials, IEEE. 16 (2014) 556–578. doi:10.1109/SURV.2013.042313.00103.

[3] L.C. Freeman, Centrality in social networks conceptual clarification, Soc. Networks. 1 (1978) 215–239. doi:10.1016/0378-8733(78)90021-7.

[4] S.P. Borgatti, M.G. Everett, A Graph-theoretic perspective on centrality, Soc. Networks. 28 (2006) 466–484. doi:10.1016/j.socnet.2005.11.005.

[5] L.C. Freeman, A set of measures of centrality based on betweenness, Sociometry. (1977) 35–41.

[6] J. Anthonisse, The rush in a directed graph, Sticht. Math. Centrum. Math. Besliskd. BN 9/71 (1971) 1–10.

[7] L.C. Freeman, S.P. Borgatti, D.R. White, Centrality in valued graphs: A measure of betweenness based on network flow, Soc. Networks. 13 (1991) 141–154. doi:10.1016/0378-8733(91)90017-N.

[8] K. Stephenson, M. Zelen, Rethinking centrality: Methods and examples, Soc. Networks. 11 (1989) 1–37. doi:10.1016/0378-8733(89)90016-6.

[9] P.S. Dodds, R. Muhamad, D.J. Watts, An experimental study of search in global social networks., Science. 301 (2003) 827–9. doi:10.1126/science.1081058.

[10] T. Spyropoulos, R.N. Bin Rais, T. Turletti, K. Obraczka, A. Vasilakos, Routing for disruption tolerant networks: taxonomy and design, Wirel. Networks. 16 (2010) 2349–2370. doi:10.1007/s11276-010-0276-9.

[11] W. Zhao, M. Ammar, E. Zegura, A message ferrying approach for data delivery in sparse mobile ad hoc networks, in: Proc. 5th ACM Int. Symp. Mob. Ad Hoc Netw. Comput. - MobiHoc '04, ACM Press, New York, New York, USA, 2004: p. 187. doi:10.1145/989459.989483.

[12] R.C. Shah, S. Roy, S. Jain, W. Brunette, Data MULEs: modeling and analysis of a three-tier architecture for sparse sensor networks, Ad Hoc Networks. 1 (2003) 215–233. doi:10.1016/S1570-8705(03)00003-9.

[13] Y. Cao, Z. Sun, Routing in delay/disruption tolerant networks: A taxonomy, survey and challenges, IEEE Commun. Surv. Tutorials. 15 (2013) 654–677.

[14] C. Boldrini, K. Lee, M. Önen, J. Ott, E. Pagani, Opportunistic networks, Comput. Commun. 48 (2014) 1–4. doi:10.1016/j.comcom.2014.04.007.

[15] Y. Zhu, B. Xu, X. Shi, Y. Wang, A survey of social-based routing in delay tolerant networks: positive and negative social effects, IEEE Commun. Surv. Tutorials. 15 (2013) 387–401.

[16] J. Zhang, G. Luo, K. Qin, H. Sun, Encounter-based routing in delay tolerant networks, in: 2011 Int. Conf. Comput. Probl., IEEE, 2011: pp. 338–341. doi:10.1109/ICCPS.2011.6089802.

[17] G. Luo, J. Zhang, H. Huang, K. Qin, H. Sun, Exploiting intercontact time for routing in delay tolerant networks, Trans. Emerg. Telecommun. Technol. 24 (2013) 589–599. doi:10.1002/ett.2553.

[18] A.C.B.K. Vendramin, A. Munaretto, M.R. Delgado, A.C. Viana, A Greedy Ant Colony Optimization for routing in delay tolerant networks, in: 2011 IEEE GLOBECOM Work., IEEE, 2011: pp. 1127–1132. doi:10.1109/GLOCOMW.2011.6162354.

[19] A. Socievole, E. Yoneki, F. De Rango, J. Crowcroft, Opportunistic message routing using multi-layer social networks, in: Proc. 2nd ACM Work. High Perform. Mob. Opportunistic Syst. - HP-MOSys '13, ACM Press, New York, New York, USA, 2013: pp. 39–46. doi:10.1145/2507908.2507923.

[20] E. Yoneki, P. Hui, S. Chan, J. Crowcroft, A socio-aware overlay for publish/subscribe communication in delay tolerant networks, in: Proc. 10th ACM Symp. Model. Anal. Simul. Wirel. Mob. Syst. - MSWiM '07, ACM Press, New York, New York, USA, 2007: p. 225. doi:10.1145/1298126.1298166.

[21] J. Patel, H. Shah, FCER routing using temporal closeness centrality for delay tolerant network, in: 2013 Nirma Univ. Int. Conf. Eng., IEEE, 2013: pp. 1–5. doi:10.1109/NUiCONE.2013.6780071.

[22] E.M. Daly, M. Haahr, Social network analysis for routing in disconnected delay-tolerant MANETs, Proc. 8th ACM Int. Symp. Mob. Ad Hoc Netw. Comput. - MobiHoc '07. (2007) 32. doi:10.1145/1288107.1288113.

[23] J. Ágila, B. Link, N. Viol, A. Goliath, K. Wehrle, J.Á. Bitsch Link, SimBetAge : Utilizing Temporal Changes in Social Networks for Pocket Switched Networks, in: Proc. 1st ACM Work. User-Provided Netw. Challenges Oppor. - U-NET '09, ACM Press, New York, New York, USA, 2009: pp. 13–18. doi:10.1145/1659029.1659034.

[24] P. Hui, J. Crowcroft, E. Yoneki, BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks, IEEE Trans. Mob. Comput. 10 (2011) 1576–1589. doi:10.1109/TMC.2010.246.

[25] J. Niu, Y. Liu, L. Shu, B. Dai, A P2P query algorithm based on Betweenness Centrality Forwarding in opportunistic networks, in: 2013 IEEE Int. Conf. Commun., IEEE, 2013: pp. 3433–3438. doi:10.1109/ICC.2013.6655080.

[26] F. Li, J. Wu, LocalCom: A Community-based Epidemic Forwarding Scheme in Disruption-tolerant Networks, in: 2009 6th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Networks, IEEE, 2009: pp. 1–9. doi:10.1109/SAHCN.2009.5168942.

[27] Jingwei Miao, O. Hasan, S.B. Mokhtar, L. Brunie, An adaptive routing algorithm for mobile delay tolerant networks, in: 2011 14th Int. Symp. Wirel. Pers. Multimed. Commun., 2011: pp. 1–5.

[28] C.-M. Kim, Y.-H. Han, J.-S. Youn, Y.-S. Jeong, A Socially Aware Routing Based on Local Contact Information in Delay-Tolerant Networks, Sci. World J. 2014 (2014) 7 pages. doi:10.1155/2014/408676.

[29] X. Zhuo, Q. Li, W. Gao, G. Cao, Y. Dai, Contact duration aware data replication in Delay Tolerant Networks, in: 2011 19th IEEE Int. Conf. Netw. Protoc., IEEE, 2011: pp. 236–245. doi:10.1109/ICNP.2011.6089057.

[30] A.C.B.K. Vendramin, A. Munaretto, M.R. de B. da S. Delgado, A.C. Viana, CGrAnt: a swarm intelligence-based routing protocol for delay tolerant networks, in: Proc. Fourteenth Int. Conf. Genet. Evol. Comput. Conf. - GECCO '12, ACM Press, New York, New York, USA, 2012: p. 33. doi:10.1145/2330163.2330169.

[31] M. Xiao, J. Wu, L. Huang, Community-Aware Opportunistic Routing in Mobile Social Networks, IEEE Trans. Comput. 63 (2014) 1682–1695. doi:10.1109/TC.2013.55.

[32] N. Vastardis, K. Yang, S. Leng, Socially-Aware Multi-phase Opportunistic Routing for Distributed Mobile Social Networks, Wirel. Pers. Commun. 79 (2014) 1343–1368. doi:10.1007/s11277-014-1933-6.

[33] Peiyan Yuan, Huadong Ma, Opportunistic forwarding with hotspot entropy, in: 2013 IEEE 14th Int. Symp. "A World Wireless, Mob. Multimed. Networks," IEEE, 2013: pp. 1–9. doi:10.1109/WoWMoM.2013.6583393.

[34] B. Guidi, M. Conti, A. Passarella, L. Ricci, Distributed protocols for Ego Betweenness Centrality computation in DOSNs, in: 2014 IEEE Int. Conf. Pervasive Comput. Commun. Work. (PERCOM Work., IEEE, 2014: pp. 539–544. doi:10.1109/PerComW.2014.6815264.

[35] K. Wei, D. Zeng, S. Guo, K. Xu, On Social Delay-Tolerant Networking: Aggregation, Tie Detection, and Routing, IEEE Trans. Parallel Distrib. Syst. 25 (2014) 1563–1573. doi:10.1109/TPDS.2013.264.

[36] N. Kourtellis, T. Alahakoon, R. Simha, A. Iamnitchi, R. Tripathi, Identifying high betweenness centrality nodes in large social networks, Soc. Netw. Anal. Min. 3 (2012) 899–914. doi:10.1007/s13278-012-0076-6.

[37] K.-I. Goh, E. Oh, B. Kahng, D. Kim, Betweenness centrality correlation in social networks, Phys. Rev. E. 67 (2003) 017101. doi:10.1103/PhysRevE.67.017101.

[38] F. Liljeros, C.R. Edling, L.A. Amaral, H.E. Stanley, Y. Aberg, The web of human sexual contacts., Nature. 411 (2001) 907–8. doi:10.1038/35082140.

[39] M.G.R. Ortiz, J.R.C. Hoyos, M.G.R. López, The social networks of academic performance in a student context of poverty in Mexico, Soc. Networks. 26 (2004) 175–188. doi:10.1016/j.socnet.2004.01.010.

[40] H. Jeong, S.P. Mason, A.L. Barabási, Z.N. Oltvai, Lethality and centrality in protein networks., Nature. 411 (2001) 41–2. doi:10.1038/35075138.

[41] S.A. Macskassy, Contextual linking behavior of bloggers: leveraging text mining to enable topic-based analysis, Soc. Netw. Anal. Min. 1 (2011) 355–375. doi:10.1007/s13278-011-0026-8.

[42] L.A. Maglaras, D. Katsaros, New measures for characterizing the significance of nodes in wireless ad hoc networks via localized path-based neighborhood analysis, Soc. Netw. Anal. Min. 2 (2011) 97–106. doi:10.1007/s13278-011-0029-5.

[43] G. Hua, D. Haughton, A network analysis of an online expertise sharing community, Soc. Netw. Anal. Min. 2 (2012) 291–303. doi:10.1007/s13278-012-0047-y.

[44] S. Catanese, E. Ferrara, G. Fiumara, Forensic analysis of phone call networks, Soc. Netw. Anal. Min. 3 (2012) 15–33. doi:10.1007/s13278-012-0060-1.

[45] C.S. Ang, Interaction networks and patterns of guild community in massively multiplayer online games, Soc. Netw. Anal. Min. 1 (2011) 341–353. doi:10.1007/s13278-011-0025-9.

[46] B.K. Singh, N. Gupte, Congestion and decongestion in a communication network, Phys. Rev. E. 71 (2005) 055103. doi:10.1103/PhysRevE.71.055103.

[47] X. Zhang, J.K. Kurose, B.N. Levine, D. Towsley, H. Zhang, Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing, in: Proc. 13th Annu. ACM Int. Conf. Mob. Comput. Netw. - MobiCom '07, ACM Press, New York, New York, USA, 2007: p. 195. doi:10.1145/1287853.1287876.

[48] M. Piorkowski, N. Sarafijanovic-Djukic, M. Grossglauser, A parsimonious model of mobile partitioned networks with clustering, in: 2009 First Int. Commun. Syst. Networks Work., IEEE, 2009: pp. 1–10. doi:10.1109/COMSNETS.2009.4808865.

[49] A. Chaintreau, P. Fraigniaud, E. Lebhar, Opportunistic spatial gossip over mobile social networks, in: Proc. First Work. Online Soc. Networks - WOSP '08, ACM Press, New York, New York, USA, 2008: p. 73. doi:10.1145/1397735.1397752.

[50] F. Chierichetti, S. Lattanzi, A. Panconesi, Gossiping (via mobile?) in social networks, in: Proc. Fifth Int. Work. Found. Mob. Comput. - DIAL M-POMC '08, ACM Press, New York, New York, USA, 2008: p. 27. doi:10.1145/1400863.1400870.

[51] E. Miluzzo, N.D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, et al., Sensing meets mobile social networks, in: Proc. 6th ACM Conf. Embed. Netw. Sens. Syst. - SenSys '08, ACM Press, New York, New York, USA, 2008: p. 337. doi:10.1145/1460412.1460445.

[52] A. Beach, M. Gartrell, S. Akkala, J. Elston, J. Kelley, K. Nishimoto, et al., WhozThat? evolving an ecosystem for context-aware mobile social networks, IEEE Netw. 22 (2008) 50–55. doi:10.1109/MNET.2008.4579771.

[53] Z.-B. Dong, G.-J. Song, K.-Q. Xie, J.-Y. Wang, An experimental study of large-scale mobile social network, in: Proc. 18th Int. Conf. World Wide Web - WWW '09, ACM Press, New York, New York, USA, 2009: p. 1175. doi:10.1145/1526709.1526915.

[54] C.E. Palazzi, A. Bujari, Social-aware delay tolerant networking for mobile-to-mobile file sharing, Int. J. Commun. Syst. 25 (2012) 1281–1299. doi:10.1002/dac.1324.

[55] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, C. Diot, Pocket switched networks and human mobility in conference environments, in: Proceeding 2005 ACM SIGCOMM Work. Delay-Tolerant Netw. - WDTN '05, ACM Press, New York, New York, USA, 2005: pp. 244–251. doi:10.1145/1080139.1080142.

[56] S. Gaito, D. Maggiorini, G.P. Rossi, A. Sala, Bus switched networks: An ad hoc mobile platform enabling urban-wide communications, Ad Hoc Networks. 10 (2012) 931–945. doi:10.1016/j.adhoc.2011.12.005.

[57] T. Liu, Y. Zhu, R. Jiang, B. Li, A sociality-aware approach to computing backbone in mobile opportunistic networks, Ad Hoc Networks. 24 (2015) 46–56. doi:10.1016/j.adhoc.2014.07.007.

[58] M.V. Barbera, J. Stefa, A.C. Viana, M.D. de Amorim, M. Boc, VIP delegation: Enabling VIPs to offload data in wireless social mobile networks, in: 2011 Int. Conf. Distrib. Comput. Sens. Syst. Work., IEEE, 2011: pp. 1–8. doi:10.1109/DCOSS.2011.5982141.

[59] C. Boldrini, A. Passarella, Data dissemination in opportunistic networks, in: S. Basagni, M. Conti, S. Giordano, I. Stojmenovic (Eds.), Mob. Ad Hoc Netw. Cut. Edge Dir., Second Edi, John Wiley & Sons, Inc, Hoboken, NJ, USA, 2013: pp. 453–490. doi:10.1002/9781118511305.ch12.

[60] A. Greede, S.M. Allen, R.M. Whitaker, RDD: Repository-based Data Dissemination protocol for opportunistic networks, in: 2012 Int. Conf. Sel. Top. Mob. Wirel. Netw., IEEE, 2012: pp. 101–106. doi:10.1109/iCOST.2012.6271275.

[61] A. Galati, V. Vukadinovic, M. Olivares, S. Mangold, Analyzing temporal metrics of public transportation for designing scalable delay-tolerant networks, in: Proc. 8th ACM Work. Perform. Monit. Meas. Heterog. Wirel. Wired Networks - PM2HW2N '13, ACM Press, New York, New York, USA, 2013: pp. 37–44. doi:10.1145/2512840.2512846.

[62] P. Pantazopoulos, I. Stavrakakis, A. Passarella, M. Conti, Efficient social-aware content placement in opportunistic networks, in: 2010 Seventh Int. Conf. Wirel. On-Demand Netw. Syst. Serv., IEEE, 2010: pp. 17–24. doi:10.1109/WONS.2010.5437139.

[63] U. Brandes, T. Erlebach, Network Analysis: Methodological Foundations, Springer Science & Business Media, 2005.

[64] J. Harris, J. Hirst, M. Mossinghoff, Combinatorics and graph theory, 2nd ed., Springer Berlin Heidelberg, New York, New York, USA, 2008.

[65] A. Casteigts, P. Flocchini, W. Quattrociocchi, N. Santoro, Time-Varying Graphs and Dynamic Networks, in: Ad-Hoc, Mobile, Wirel. Networks, Springer Berlin Heidelberg, 2010: pp. 1–20.

[66] J. Tang, M. Musolesi, C. Mascolo, V. Latora, V. Nicosia, Analysing Information Flows and Key Mediators through Temporal Centrality Metrics Categories and Subject Descriptors, in: Proc. 3rd Work. Soc. Netw. Syst. - SNS '10, ACM Press, New York, New York, USA, 2010: pp. 1–6. doi:10.1145/1852658.1852661.

[67] S. Milgram, The small world problem, Psychol. Today. 2 (1967) 60–67.

[68] M.E.J. Newman, A measure of betweenness centrality based on random walks, Soc. Networks. 27 (2005) 39–54. doi:10.1016/j.socnet.2004.11.009.

[69] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to algorithms, third edit, MIT Press, 2009.

[70] M. Everett, S.P. Borgatti, Ego network betweenness, Soc. Networks. 27 (2005) 31–38. doi:10.1016/j.socnet.2004.11.007.

[71] R.S. Burt, Structural Holes: The Social Structure of Competition, Harvard University Press, 1995.

[72] R. Geisberger, P. Sanders, D. Schultes, Better Approximation of Betweenness Centrality, ALENEX. (2008) 90–100.

[73] T. Carpenter, G. Karakostas, D. Shallcross, S. Street, Practical Issues and Algorithms for Analyzing Terrorist Networks, in: Proc. West. Simul. MultiConference, 2002.

[74] J. Garreau, Disconnect the dots, Washington Post. 17 (2001) C1. http://www.seity.com/wp-content/themes/seity/pdf/disconnect the dots.pdf.

[75] T.A. Stewart, Six degrees of Mohamed Atta, Business. 2 (2001) 10. http://faculty.cbpp.uaa.alaska.edu/afgjp/PADM610/Six Degrees of Mohamed Atta.pdf.

[76] S. Ressler, Social network analysis as an approach to combat terrorism: past, present, and future research, Homel. Secur. Aff. 2 (2006) 1–10.

[77] C. Demetrescu, G.F. Italiano, Fully dynamic all pairs shortest paths with real edge weights, in: Proc. 2001 IEEE Int. Conf. Clust. Comput., IEEE Comput. Soc, 2001: pp. 260–267. doi:10.1109/SFCS.2001.959900.

[78] C. Demetrescu, G.F. Italiano, A new approach to dynamic all pairs shortest paths, J. ACM. 51 (2004) 968–992. doi:10.1145/1039488.1039492.

[79] U. Brandes, On variants of shortest-path betweenness centrality and their generic computation, Soc. Networks. 30 (2008) 136–145. doi:10.1016/j.socnet.2007.11.001.

[80] T. Opsahl, F. Agneessens, J. Skvoretz, Node centrality in weighted networks: Generalizing degree and shortest paths, Soc. Networks. 32 (2010) 245–251. doi:10.1016/j.socnet.2010.03.006.

[81] U. Brandes, A faster algorithm for betweenness centrality, J. Math. Sociol. 25 (2001) 163–177. doi:10.1080/0022250X.2001.9990249.

[82] M.E.J. Newman, Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality, Phys. Rev. E. 64 (2001) 016132. doi:10.1103/PhysRevE.64.016132.

[83] E.W. Dijkstra, A note on two problems in connexion with graphs, Numer. Math. 1 (1959) 269–271. doi:10.1007/BF01386390.

[84] U. Brandes, C. Pich, Centrality Estimation In Large Networks, Int. J. Bifurc. Chaos. 17 (2007) 2303–2318. doi:10.1142/S0218127407018403.

[85] M. Riondato, E.M. Kornaropoulos, Fast approximation of betweenness centrality through sampling, Proc. 7th ACM Int. Conf. Web Search Data Min. - WSDM '14. (2014) 413–422. doi:10.1145/2556195.2556224.

[86] D.A. Bader, S. Kintali, K. Madduri, M. Mihail, Approximating Betweenness Centrality, Algorithms Model. Web-Graph. (2007) 124–137.

[87] M. Hinne, Local approximation of centrality measures, Radboud University Nijmegen, The Netherlands, 2011.

[88] M.L. Fredman, R.E. Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms, J. ACM. 34 (1987) 596–615.

[89] D. Eppstein, Fast Approximation of Centrality, Graph Algorithms Appl. 8 (2004) 39–45.

[90] W. Hoeffding, Probability Inequalities for Sums of Bounded Random Variables, Am. Stat. Assoc. 58 (1963) 13–30.

[91] S. Har-Peled, M. Sharir, Relative (p,ε)-Approximations in Geometry, Discrete Comput. Geom. 45 (2010) 462–496. doi:10.1007/s00454-010-9248-1.

[92] Y. Li, P.P.M. Long, A. Srinivasan, Improved bounds on the sample complexity of learning, in: Proc. Elev. Annu. ACM-SIAM Symp. Discret. Algorithms, Society for Industrial and Applied Mathematics, 2000: pp. 309–318.

[93] P. V Marsden, Egocentric and sociocentric measures of network centrality, Soc. Networks. 24 (2002) 407–422.

[94] S. Okasha, Altruism, group selection and correlated interaction, Br. J. Philos. Sci. 56 (2005) 703–725.

[95] L.A. Adamic, R.M. Lukose, A.R. Puniyani, B.A. Huberman, Search in power-law networks, Phys. Rev. E. 64 (2001) 046135. doi:10.1103/PhysRevE.64.046135.

[96] P.R. Winters, Forecasting sales by exponentially weighted moving averages, Manage. Sci. 6 (1960) 324–342. doi:10.1287/mnsc.6.3.324.

[97] G. Palla, I. Derenyi, I. Farkas, T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, Nature. 435 (2005) 814–818. doi:10.1038/nature03607.

[98] M.E. Newman, Analysis of weighted networks, Phys. Rev. E. 70 (2004) 056131. doi:10.1103/PhysRevE.70.056131.

[99] M. Dorigo, G. Di Caro, The ant colony optimization meta-heuristic, in: D. Corne, M. Dorigo, F. Glover, D. Dasgupta, P. Moscato, R. Poli, et al. (Eds.), New Ideas Optim., McGraw-Hill Ltd., UK, Maidenhead, UK, England, 1999: pp. 11–32.

[100] E. Bonabeau, M. Dorigo, G. Theraulaz, From Natural to Artificial Swarm Intelligence, Oxford University Press, 1999.

[101] S. Goss, S. Aron, J.L. Deneubourg, J.M. Pasteels, Self-organized shortcuts in the Argentine ant, Naturwissenschaften. 76 (1989) 579–581. doi:10.1007/BF00462870.

[102] C.-M. Kim, Y.-H. Han, J.-S. Youn, Y.-S. Jeong, Betweenness of Expanded Ego Networks in Sociality-Aware Delay Tolerant Networks, in: Y.-H. Han, D.-S. Park, W. Jia, S.-S. Yeo (Eds.), Ubiquitous Inf. Technol. Appl., Springer Netherlands, Dordrecht, 2013: pp. 499–505. doi:10.1007/978-94-007-5857-5.

[103] E. Bulut, B.K. Szymanski, Friendship Based Routing in Delay Tolerant Mobile Social Networks, in: 2010 IEEE Glob. Telecommun. Conf. GLOBECOM 2010, IEEE, 2010: pp. 1–5. doi:10.1109/GLOCOM.2010.5683082.

[104] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, J. Scott, Impact of Human Mobility on Opportunistic Forwarding Algorithms, IEEE Trans. Mob. Comput. 6 (2007) 606–620. doi:10.1109/TMC.2007.1060.

[105] U. Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (2007) 395–416. doi:10.1007/s11222-007-9033-z.

[106] J. Wu, M. Xiao, L. Huang, Homing spread: Community home-based multi-copy routing in mobile social networks, in: 2013 Proc. IEEE INFOCOM, IEEE, 2013: pp. 2319–2327. doi:10.1109/INFCOM.2013.6567036.

[107] M. Ibrahim, P. Nain, I. Carreras, Analysis of relay protocols for throwbox-equipped DTNs, in: 2009 7th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wirel. Networks, IEEE, 2009: pp. 1–9. doi:10.1109/WIOPT.2009.5291625.

[108] A. Mtibaa, M. May, C. Diot, M. Ammar, PeopleRank: Social Opportunistic Forwarding, in: 2010 Proc. IEEE INFOCOM, Ieee, 2010: pp. 1–5. doi:10.1109/INFCOM.2010.5462261.

[109] W. Gao, G. Cao, Fine-grained mobility characterization, in: Proc. Elev. ACM Int. Symp. Mob. Ad Hoc Netw. Comput. - MobiHoc '10, ACM Press, New York, New York, USA, 2010: p. 61. doi:10.1145/1860093.1860103.

[110] K. Lee, S. Hong, S.J. Kim, I. Rhee, S. Chong, Demystifying levy walk patterns in human walks, CS Dept., NCSU, Raleigh, NC, USA, Tech. Rep, 2008.

[111] F.H. Allport, G.W. Allport, Personality Traits: Their Classification and Measurement., J. Abnorm. Psychol. Soc. Psychol. 16 (1921) 6–40. doi:http://dx.doi.org/10.1037/h0069790.

[112] T.M. Cover, J.A. Thomas, Elements of information theory, 2nd ed., John Wiley & Sons, 2012.

[113] K. Anand, G. Bianconi, Entropy measures for networks: Toward an information theory of complex topologies, Phys. Rev. E. 80 (2009) 045102. doi:10.1103/PhysRevE.80.045102.

[114] M. Jelasity, Gossip, in: G. Di Marzo Serugendo, M.-P. Gleizes, A. Karageorgos (Eds.), Self-Organising Softw., Springer Berlin Heidelberg, Berlin, Heidelberg, 2011: pp. 139–162. doi:10.1007/978-3-642-17348-6.

[115] W. Hwang, T. Kim, M. Ramanathan, A. Zhang, Bridging centrality: graph mining from element level to group level, in: Proceeding 14th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD 08, ACM Press, New York, New York, USA, 2008: p. 336. doi:10.1145/1401890.1401934.

[116] N. Eagle, A. Pentland, CRAWDAD data set mit/reality (v. 2005-07-01), (2005). http://crawdad.cs.dartmouth.edu/mit/reality.

[117] D. Kotz, T. Henderson, I. Abyzov, J. Yeo, CRAWDAD data set dartmouth/campus (v. 2009-09-09), (2009). http://crawdad.cs.dartmouth.edu/dartmouth/campus.

[118] D.R. Ford Jr., L.R. and Fulkerson, Maximal Flow Through a Network, in: G.-C. Gessel, Ira and Rota (Ed.), Class. Pap. Comb., Birkhäuser Boston, 1987: pp. 243–248. doi:10.1007/978-0-8176-4842-8_15.

[119] R. Baldoni, R. Beraldi, L. Querzoni, G. Cugola, M. Migliavacca, Content-based routing in highly dynamic mobile ad hoc networks, Int. J. Pervasive Comput. Commun. 1 (2005) 277–288. doi:10.1108/17427370580000131.

[120] A. Vahdat, D. Becker, Epidemic routing for partially connected ad hoc networks, Technical Report CS-200006, Duke University, 2000.