

Temporal Node Centrality in Complex Networks

Hyounghick Kim and Ross Anderson

Computer Laboratory, University of Cambridge, 15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom

(Dated: October 12, 2011)

Many networks are dynamic in that their topology changes rapidly – on the same time-scale as the communications of interest between network nodes. Examples are the human contact networks involved in the transmission of disease, ad-hoc radio networks between moving vehicles, and the transactions between principals in a market. While we have good models of static networks, so far these have been lacking for the dynamic case. In this paper we present a simple but powerful model, the *time-ordered graph*, which reduces a dynamic network to a static network with directed flows. This enables us to extend network properties such as vertex degree, closeness and betweenness centrality metrics in a very natural way to the dynamic case. We then demonstrate how our new model applies to a number of interesting edge cases, such as where the network connectivity depends on a small number of highly mobile vertices or edges, and show that our new centrality definition allows us to graph the evolution of connectivity. Finally we apply our model and techniques to two real-world dynamic graphs of human contact networks and then discuss the implication of temporal centrality metrics in the real-world.

INTRODUCTION

Many important phenomena depend on networks, from the spread of disease in a population through systems of metabolic processes to explicit networks such as the Internet and the World Wide Web. Recent advances in the theory of networks have provided us with the mathematical and computational tools to understand them better [1]. Often the topology of a network has distinctive features, such as vertex order distribution, clustering and mean path length, which can be explained in terms of its evolution and which in turn explain some aspects of its behaviour. For example, networks that grow by preferential attachment may acquire a power-law distribution of vertex order which in turn makes them robust against random node failure – yet vulnerable to attacks targeted on high-degree nodes [2]. Insights like this can inform activities from public health to counterterrorism [3].

So most analysis and models have assumed that networks are static, typically represented in graph form as a number of nodes connected by edges. However in real life many networks are dynamic. New nodes are added to the graph, some existing ones are removed, and edges come and go too. While researchers have studied these mechanisms as a means of explaining graph topology, the effects of dynamic topology have generally been ignored when considering how topology affects connectivity. Yet there are important networks whose topology changes rapidly, and its dynamic aspects have a significant effect on connectivity:

- In epidemiology, some possibly infective contacts between individuals are long-term (friends, family) but many are fleeting (people in the street or the market place). In medieval times, infectious disease may have been largely transmitted by a small number of merchants travelling between the markets in otherwise largely isolated towns, while in

a modern urban society the super-spreading node may be a school. When faced with an epidemic, it's important to know whether you should impose travel restrictions or close schools.

- There is interest in ad-hoc networks radio networks set up between moving vehicles to transmit information about congestion and to provide emergency communications. Here oncoming vehicles offer a shorter interaction time, but more rapid information dissemination, than vehicles going in the same direction.
- In military communications systems, nodes that act as local exchanges or that provide long-distance backhaul may become conspicuous because of the volumes of traffic they handle, even if the opponent cannot decrypt and understand it, so they may be targeted. So nodes may take turns; a new exchange may be selected frequently and at random.

Thus far, the models and analytic tools used to characterise dynamic network behaviour have been somewhat limited. It's simple and common to look at static snapshots of the network independently, and use the average characteristics of all snapshots; for example, a possible way of estimating a node's topological importance over time is to use the average value on the node's centrality over all static snapshots. Such dynamic analyses, however, are limited since they neglect temporal paths that can cross over multiple temporal snapshots. In terms of dynamic graph, paths between nodes frequently exist by sewing by sewing partial paths between temporal snapshots. Tang et al [4] proposed an analysis method to identify topologically important nodes in dynamic networks based on a temporal version of shortest path. They proposed temporal versions of conventional centrality metrics (e.g. *closeness* and *betweenness*) based on shortest paths in terms of dynamic networks.

We extend their work into a more generalized model: While they simply assumed the shortest path length in the unweighted scenario, our definitions are different and more general. Moreover, the centrality metrics defined by [4] can sometimes be biased since the temporal paths during a time interval alone are considered. We show that the proposed centrality metrics lead to more accurate results since the temporal paths are additionally considered, which can be ignored in the existing definitions.

RELATED WORK

Traditional network analyses and models have used networks which are built as the results of aggregation of interaction between nodes during a certain time interval. But if the rate of topological change in a network is sufficiently high, simple aggregation models may not lead to a better understanding of the network. For example, aggregate networks tend to result in underestimation of the length of the path since they do not consider the time delay for constructing paths in practice. Therefore a number of studies have recently been introduced to overcome these limitations and provide a better understanding of dynamic characteristics of real networks.

Kempe et al. [5] proposed a model of temporal networks as static graphs where every edge is labelled with the time that the interaction took place. Ferreira [6] also views a dynamic network as a sequence of static graphs to define the fundamental networking problems such as routing metrics, connectivity, and spanning trees in terms of dynamic networks. Kostakos [7] independently presented the concept of temporal graphs to compute the shortest path length between nodes in a network over time. Tang et al. [8] tried to develop a more generalized model by introducing a variable representing the speed that a message travels. However, these models do not allow for analysis of graphs with different edge weights; we should assume that all connections have the same propagation delay.

More recently, Tang et al. [4] proposed temporal centrality metrics based on temporal paths in order to effectively measure the importance of a node in a dynamic network. They demonstrated the feasibility of time-aware central node identification methods with several real datasets but the performance of the proposed metrics may be overestimated since a priori knowledge of future contacts, which is not available in practice, is used in the experiments.

Our work is an extension to these proposals, with a focus on the design more generalized and simple definitions. The proposed representation allows us a concise and general formulation of temporal properties of a dynamic network.

MODEL

In this study, we assume that the time during which a network is observed is finite (from the start time t_{start} until the end time t_{end}). Without loss of generality, we set $t_{start} = 0$ and $t_{end} = T$. A *dynamic network* $G_{0,T}^D = (V, E_{0,T})$ on a time interval $[0, T]$ consists of a set of vertices V and a set of temporal edges $E_{0,T}$ where a temporal edge $(u, v)_{i,j} \in E_{0,T}$ exists between vertices u and v on a time interval $[i, j]$ such that $i \leq T$ and $j \geq 0$. In the dynamic network the set of vertices V is always the same while the set of existing edges can be changed over time.

Most characterisations of dynamic networks discretise time by converting temporal information into a sequence of n network “snapshots”. We use w to denote the time duration of each snapshot (or time window size), T/n , expressed in some time unit (e.g., seconds or hours). In other words, a dynamic network can be represented as a series of static graphs G_1, G_2, \dots, G_n . The notation G_t ($1 \leq t \leq n$) represents the aggregate graph which consists of a set of vertices V and a set of edges E_t where an edge $(u, v) \in E_t$ exists only if a temporal edge $(u, v)_{i,j} \in E_{0,T}$ exists between vertices u and v on a time interval $[i, j]$ such that $i \leq w \cdot t$ and $j > w \cdot (t-1)$. In other words, G_t is the t th temporal snapshot of the dynamic network $G_{0,T}^D$ during t th time window. For simplicity, we here assume that $w = 1$.

We introduce the following illustrative example. When $T = 3$, the dynamic network with the set of temporal edges in Table I can be represented as the aggregated graph where all edges are aggregated into a single graph, $G_{1,3}^S$, or the series of static networks, G_1, G_2 and G_3 as we explained. The visual representations are shown in Figure 1. Unlike the aggregated view of the graph $G_{1,3}^S$ in Figure 1(a), the series of static networks, G_1, G_2 and G_3 in Figure 1(b), shows the temporal relationships effectively.

TABLE I. Example contacts in dynamic network.

Edge	Time interval
(A, C)	[1, 1]
(A, D)	[2, 2]
(B, D)	[2, 3]
(C, D)	[3, 3]

Although this time series representation of the graph (see Figure 1(b)) is intuitive, it is not easy to directly analyse the temporal characteristics of a dynamic network from snapshots of the network. For example, when an edge (u, v) in a dynamic network represents the communication channel u and v , we may want the problem to find the shortest possible route from u to v . In order to find a path from A to B, we have to wait at $t = 0$, use the path from A to D at $t = 1$ and then use the path from D to B at $t = 2$. However, how can we find this

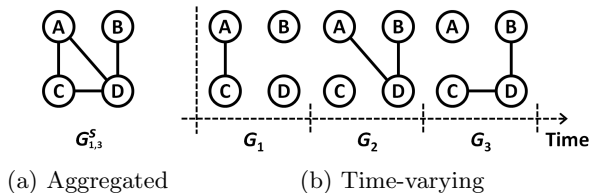


FIG. 1. Comparison of (a) aggregated graph representation and (b) time series representation of the contacts in Table I.

solution more generally?

We now construct the *time-ordered graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ as the asymmetric directed graph shown in Figure 2. Without loss of generality, we assume that the message transmission time is the same as w . In other words, at each time step, we can deliver a message along a single edge. It has a vertex v_t for each $v \in V$ and for each $t \in \{0, 1, \dots, n\}$; it has edges from u_{t-1} to v_t and vice versa for an edge $(u, v) \in E[t]$; and it has edges from v_{t-1} to v_t for all $v \in V$ and for all $t \in \{1, \dots, n\}$.

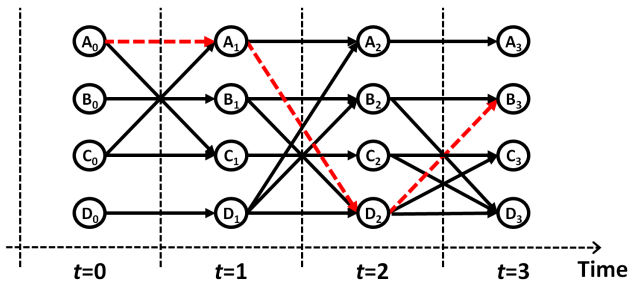


FIG. 2. The corresponding time-ordered graph \mathcal{G} of Table I. The path consisting of red dashed edges represents a temporal shortest path from A to B on the time interval $[0, 3]$.

The point of this construction is that for every path between two nodes for given start and end times in a dynamic network, there is exactly one path between the corresponding vertices in the corresponding time-ordered graph, which thus captures all the connectivity information in the network. It has much finer granularity than an existing model [7, 8] which assumes that a message can be delivered to the nodes within h hops at the same window. Under this assumption, the existing model cannot allow to use different speed of transmission. To make matters worse, if the message transmission time between some nodes is greater than the time window size w , these edges cannot be represented. Unlike the existing model, our model can represent different speed of transmission naturally. In order to show this, we introduce the following example with different speed of transmission. From Figure 3, when the transmission time between A and D is ‘2’ and the other transmission time is ‘1’, \mathcal{G} can be constructed as the weighted graph shown in Figure 3.

Moreover, we can apply conventional graph theory algorithms to analyse the temporal characteristics of the

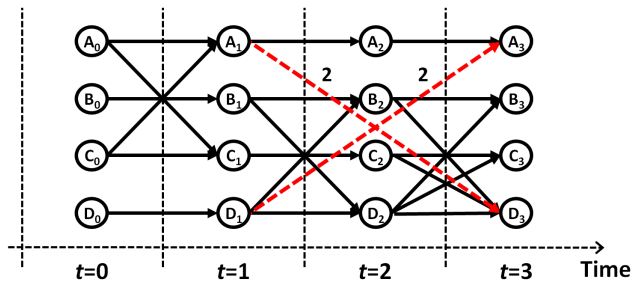


FIG. 3. The corresponding time-ordered graph \mathcal{G} of Table I when the transmission time between A and D is ‘2’. The red dashed edges with edge weight ‘2’ represents the paths between A and D.

dynamic network. For example, given a time-ordered graph \mathcal{G} , a *temporal shortest path* from node u to node v on a time interval $[i, j]$ where $0 \leq i < j \leq n$ is defined as any path $p = \langle u_i, \dots, v_k \rangle$ where $i < k \leq j$ with the path length $|p| = \min_{i < t \leq j} \delta(u_i, v_t)$ where $\delta(u, v)$ is the shortest path distance from u to v in a static graph. Thus in Figure 2 a temporal shortest path from A to B on the time interval $[0, 3]$ is clearly A_0, A_1, D_2 , and B_3 .

Given a time-ordered graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ constructed from n static networks of a *dynamic network* $G_{i,j}^D = (V, E_{i,j})$, we can now offer definitions of centrality metrics (degree, closeness, betweenness, and etc.) to capture the temporal characteristics of dynamic networks as follows.

Temporal degree

The temporal degree $\mathcal{D}_{i,j}(v)$ for a node $v \in V$ on a time interval $[i, j]$ where $0 \leq i < j \leq n$ is the total number of inbound edges to and outbound edges from v on the time interval $[i, j]$, disregarding the ‘self-edges’ from v_{t-1} to v_t for all $t \in \{i+1, \dots, j\}$. This is equal to $\sum_{t=i}^j 2 \cdot \mathcal{D}_t(v)$ where $\mathcal{D}_t(v)$ is the degree of v in G_t .

The temporal degree can be normalised by dividing each node degree by $2 \cdot (|V| - 1) \cdot m$ where $m = j - i$. We note that a node’s normalized temporal degree is the same as the average value of the node’s degree values in the time series of graphs.

Given a time-ordered graph \mathcal{G} derived from $G_{i,j}^D = (V, E_{i,j})$, the temporal degree values of all nodes in V can be computed in $O(|V| + |\mathcal{E}|)$ time by checking the nodes adjacent to each edge in \mathcal{E} .

Temporal closeness

The temporal closeness $\mathcal{C}_{i,j}(v)$ for a node $v \in V$ on a time interval $[i, j]$ where $0 \leq i < j \leq n$ is the sum of inversed temporal shortest path distances to all other nodes in $V \setminus v$ for each time interval in $\{\{t, j\} : i \leq t < j\}$.

We define the temporal closeness by considering m time intervals $\{[t, j] : i \leq t < j\}$ where $m = j - i$ by varying the starting time t of each time interval from i to $j - 1$ instead of one time interval $[i, j]$ with the starting time i . We note that the time interval $[i, j]$ contributes the *temporal shortest paths* only when the starting time is i ; the *temporal shortest paths* from node u to node v mean the paths from node u_i to node v_k which is the first node encountered along a path from u_i to a node in $\{v_{i+1}, \dots, v_j\}$. However, the *temporal shortest paths* from u to v will change as time increases. Therefore, in addition to the case with the starting time i , we also need to consider the *temporal shortest paths* from node u to node v on the additional $m - 1$ time intervals $\{[t, j] : i < t < j\}$ by varying t from $i + 1$ to $j - 1$ to analyse dynamic characteristics of the temporal shortest paths between u and v in a more reasonable manner.

An example in Figure 4 supports our design principle clearly. In this example, if we consider the time interval $[i, j]$ alone, the temporal closeness values of all nodes are identical since all temporal shortest paths are determined during the time interval $[0, 1]$ when the graph is fully connected and the subsequent interactions are ignored in computing the temporal closeness. This is not satisfactory; we can see that the node A is highly connected in the network compared to the other nodes over time. A reasonable temporal centrality metric should capture such dynamics over time. In the existing work [4], however, temporal metrics are defined with the time interval $[i, j]$ alone rather than all the time intervals $\{[t, j] : i \leq t < j\}$.

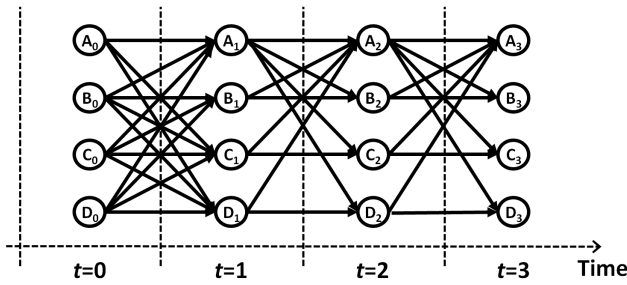


FIG. 4. An example of time-ordered graph \mathcal{G} to explain our design principle for the temporal closeness and betweenness. If we consider the time interval $[i, j]$ only, the temporal shortest paths between all nodes are determined during the time interval $[0, 1]$ alone regardless of the subsequent changes. It is not desirable to analyse the dynamic characteristics of this graph.

Formally, the temporal closeness for a node v is

$$\mathcal{C}_{i,j}(v) = \sum_{i \leq t < j} \sum_{u \in V \setminus v} \frac{1}{\Delta_{t,j}(v, u)}$$

where $\Delta_{t,j}(v, u)$ is the temporal shortest path distance from v to u on a time interval $[t, j]$. If there is no tem-

poral path from v to u on a time interval $[t, j]$, $\Delta_{t,j}(v, u)$ is defined as ∞ . Also, we note that $\Delta_{t,j}(v, u)$ is different from $\Delta_{t,j}(u, v)$ since the time-ordered graph \mathcal{G} is a directed graph.

In order to cover the cases when $\Delta(v, u)$ is infinite, we use the slightly modified definition for closeness which is similar to the definition which was proposed by Opsahl [9] for disconnected graphs. Here we assume $1/\infty = 0$.

The temporal closeness can be normalised by dividing each closeness value by $(|V| - 1) \cdot m$ where $m = j - i$.

Given a time-ordered graph \mathcal{G} derived from $G_{i,j}^{\mathcal{D}} = (V, E_{i,j})$, all-pair temporal shortest path distances can be computed in $O(m \cdot |V|^2)$ time by using dynamic programming with the recurrence $\Delta_{t,j}(v, u) = \Delta_{t+1,j}(k, u) + 1$ if $(v, k) \in \mathcal{E}$; otherwise, $\Delta_{t,j}(v, u) = 0$. With the computed temporal shortest path distances, the temporal closeness value $\mathcal{C}_{i,j}(v)$ of a node v in V can be computed in $O(m \cdot |V|)$, and thus the total running time of the temporal closeness computation for all nodes in V is $O(m \cdot |V|^2)$.

Temporal betweenness

The temporal betweenness $\mathcal{B}_{i,j}(v)$ for a node $v \in V$ on a time interval $[i, j]$ where $0 \leq i < j \leq n$ is the sum of the proportion of all the temporal shortest paths through the vertex v to the total number of temporal shortest paths over all pairs of nodes for each time interval in $\{[t, j] : i \leq t < j\}$. For the same reason as the temporal closeness definition, we consider m time intervals $\{[t, j] : i \leq t < j\}$ where $m = j - i$ instead of one time interval $[i, j]$.

Let $\mathcal{S}_{x,y}(u, v)$ denote the set of temporal shortest paths from source s to destination d on the time interval $[x, y]$ and $\mathcal{S}_{x,y}(s, d, v)$ the subset of $\mathcal{S}_{x,y}(s, d)$ consisting of paths that have v in their interior. Then, the temporal betweenness for a node v is

$$\mathcal{B}_{i,j}(v) = \sum_{i \leq t < j} \sum_{\substack{s \neq v \neq d \in V \\ \sigma_{t,j}(s, d) > 0}} \frac{\sigma_{t,j}(s, d, v)}{\sigma_{t,j}(s, d)}$$

where $\sigma_{t,j}(s, d) \equiv |\mathcal{S}_{t,j}(s, d)|$ and $\sigma_{t,j}(s, d, v) \equiv |\mathcal{S}_{t,j}(s, d, v)|$.

The temporal betweenness can be normalised by dividing each betweenness value by $(V_s^v \cdot V_d^v \cdot m)$ where $m = j - i$ and $V_s^v, V_d^v \subseteq V \setminus v$ such that $\sigma_{t,j}(s, d) > 0$ for each $s \in V_s^v$, for each $d \in V_d^v$ and $i \leq t < j$.

Given a time-ordered graph \mathcal{G} derived from $G_{i,j}^{\mathcal{D}} = (V, E_{i,j})$, the temporal betweenness values for all nodes in V can be efficiently calculated by using dynamic programming: For each node $v \in V$, $\sigma_{t,j}(s, d, v) = \sigma_{t,j}(s, v) \cdot \sigma_{t,j}(v, d)$ if $\Delta_{t,j}(s, d) = \Delta_{t,j}(s, v) + \Delta_{t,j}(v, d)$ where $s \neq d \in V$ and $i \leq t < j$. Since $\mathcal{G} = (V, \mathcal{E})$ is a directed acyclic graph, we compute $\Delta_{t,j}(v, v)$ and

$\sigma_{t,j}(\nu, v)$ between $\nu \in \mathcal{V}$ and $v \in V$ where $i \leq t < j$ in $O(m^3 \cdot |V|^2 + |\mathcal{E}|)$ time. We note that the worst case running time of this computation is $O(m^3 \cdot |V|^2)$ since $|\mathcal{E}|$ is $O(m \cdot |V|^2)$ when all nodes are always completely connected to all nodes at each time step t where $i \leq t < j$. With the computed Δ and σ values, for all nodes $\nu \in \mathcal{V}$, we can compute $\sigma_{t,j}(s, d, \nu)$ in $O(m^3 \cdot |V|^3)$ time for $i \leq t < j$ and thus the total running time of the temporal betweenness computation is $O(m^3 \cdot |V|^3)$. Also this algorithm requires $O(m^2 \cdot |V|^2)$ space to store Δ and σ values.

Analysis of the example

We compute the proposed metrics for the time-ordered graph in Figure 2. For comparison, we also compute each node's the centrality value in the aggregated graph (see Figure 1 (a)) and the average centrality values in G_1 , G_2 , and G_3 (see Figure 1 (b)). The results are shown in Table II. From this table, we can see that A plays a relatively important role compared with C in the dynamic network while the network centrality values of A and C for the aggregated representation and the average centrality value in G_1 , G_2 , and G_3 are exactly identical. Intuitively, our metric seems reasonable as the edge (A, D) alone exists at time $t = 2$.

TABLE II. Comparison with static and average centrality metrics.

Node	Type	degree	closeness	betweenness
A	Temporal	0.222	0.426	0.133
	Aggregated	0.667	0.750	0.000
B	Temporal	0.222	0.370	0.000
	Aggregated	0.333	0.600	0.000
C	Temporal	0.222	0.370	0.000
	Aggregated	0.667	0.750	0.000
D	Temporal	0.444	0.648	0.750
	Aggregated	1.000	1.000	0.667
	Average	0.444	0.444	0.222

In the next two sections we describe how our temporal metrics effectively work.

WHY ARE TEMPORAL METRICS REALLY NECESSARY?

In order to test the effectiveness of the temporal metrics, we define a dynamic network model which we call the *Travelling Merchant Graph* (TMG) to model disease transmission in a traditional society where a few merchants travel but most people stay in their villages. A

TMG $G(\eta, \nu, \gamma, p, b, d)$ is defined by six parameters, the number η of merchants, the number ν of villages, the number γ of residents in each village, and the other three parameters p , b , and d to control the interconnections between residents in a village: start with ν mutually exclusive random graphs $G_{\gamma,p}$ and η merchant nodes which are only connected to a resident node in a random graph, respectively; At every time step, the existing edges are removed and/or the new edges are added as follows:

- **Internal Movement:** For each village $G^i = (V^i, E^i)$ where $i \in \{1, \dots, \nu\}$, an existing edge $e \in E^i$ will *die* with the probability d ; while, a non-existing edge $\hat{e} \notin E^i$ will *appear* with the probability b .
- **External Movement:** For each merchant node v^j where $j \in \{1, \dots, \eta\}$, an existing edge (v^j, v^{old}) will *die* and then a new edge (v^j, v^{new}) will *appear* with the mobility probability $\text{prob}_{\text{mobility}}(v^j)$ where v^{old} and v^{new} are resident nodes in villages, G^{old} and $G^{\text{new}} (\neq G^{\text{new}})$, respectively.

We use the probability $\text{prob}_{\text{mobility}}(\cdot)$ to differentiate the mobility of each merchant. In other words, merchant u moves with the probability $\text{prob}_{\text{mobility}}(u)$. We here set $\text{prob}_{\text{mobility}}(u) \geq 0.5$. Each merchant moves with a probability randomly assigned between 0.5 and 1.

We show that our temporal metrics only capture the merchants' long-distance mobility in TMG effectively. In order to show this, we generate 100 *travelling merchant graphs* with $\eta = 1$, $\nu = 5$, $\gamma = 6$, $p = 0.4$, $b = 0.1$, and $d = 0.1$ during 100 steps. We compute the mean values of the *temporal* degree, closeness and betweenness centrality metrics, respectively, for merchant and resident nodes. For comparison, we also compute the mean values of the centrality metrics in the *aggregated* graph and the *average* centrality metrics in the time series of graphs, respectively. Figure 5 demonstrates these comparison results.



FIG. 5. The comparison between the centrality of the resident and merchant nodes in the *aggregated* graph, the *average* centrality of those in the time series of graphs, and the *temporal* centrality of those for TMGs with $\eta = 1$, $\nu = 5$, $\gamma = 6$, $p = 0.4$, $b = 0.1$, and $d = 0.1$.

If we use each node's *average* centrality values to measure its relative importance then as it is computed from snapshots, the many temporal paths through the merchant nodes are ignored. Even though the centrality

values in the *aggregated* graph can distinguish merchant node from residents nodes well, these centrality values are highly overestimated by ignoring the frequency of interactions (or contacts). In order to show this, we analyse the relation between merchants' centrality values and mobility by calculating the Pearson correlation coefficients among them. The results are shown in Table III.

TABLE III. The correlation analysis between merchants' centrality values and mobility for each metric type.

Type	degree	closeness	betweenness
Temporal	—	0.936	0.883
Aggregated	0.645	0.653	—
Average	—	0.088	—

From the 100 TMGs in the experiments, in the *aggregated* graphs, we can see that the merchant node is always on all shortest paths between all pairs of resident nodes belonging to different villages regardless of its mobility. In other words, the merchant node's betweenness value (0.828) is always fixed in the *aggregated* graphs. We note that the Pearson correlation is defined only if the standard deviations are finite and both of them are non-zero. Thus we cannot define the Pearson correlation coefficients between the merchant nodes' mobility and their betweenness values in the *aggregated* graphs. For the same reason, *temporal* degree, *average* degree and *average* betweenness cannot be defined. In TMGs, the standard deviation of a merchant node's degree is zero since the node is always connected to only one resident node. Also, in the time series representation of a TMG, a merchant node's betweenness is zero in each snapshot since the node is always a leaf node. This result implies that it is difficult to identify the merchant nodes with high mobility from their betweenness values in the *aggregated* graph. In fact, we do not recommend the degree and closeness metrics in *aggregated* graphs, too since a merchant node will be connected to all resident nodes even though it has low mobility if enough time has elapsed. Unlike them, the merchant nodes' *temporal* closeness and betweenness values are highly correlated with their mobility. This is natural; a node with high mobility has high centrality in dynamic networks since the merchant nodes' other conditions are all identical.

We found that the merchant node with the highest mobility can be identified with a high probability in a TMG with multiple merchants by computing their *temporal* closeness or betweenness values. We generated 100 TMGs with $\eta = 4$, $\nu = 5$, $\gamma = 6$, $p = 0.4$, $b = 0.1$, and $d = 0.1$. We tried to identify the merchant node with the highest mobility by selecting the merchant node with the highest centrality value for each metric. When the merchant node with the highest centrality value is not

unique, we arbitrarily choose one of the nodes with the same highest centrality value. Table IV shows the accuracy of such selection based on each centrality metric. In this table, we can see that the high mobility nodes are identified with high probability using *temporal* closeness or betweenness.

TABLE IV. The detection accuracy of the highest mobility merchant node.

Type	degree	closeness	betweenness
Temporal	20%	72%	80%
Aggregated	47%	47%	50%
Average	20%	19%	20%

EFFECTIVENESS OF TEMPORAL CENTRALITY ON REAL DYNAMIC NETWORKS

In order to discuss the effectiveness of temporal centrality metrics in the real-world, we perform tests whether the computed temporal centrality metrics are really meaningful in practice. When the network topology of nodes can change over time as new edges are created or existing ones removed, can we use the temporal centrality computed from the contacts history between nodes to estimate the importance of nodes (in terms of dynamic networks) in future? As an extreme example, if the past human contacts information is totally independent from their future contacts behavior, it will be useless to provide network centrality computed from the past human contacts information. In this section we discuss the implication of temporal centrality metrics in the real-world. For brevity we will only consider the *temporal* closeness and betweenness here.

TABLE V. Experimental Datasets

	CAMBRIDGE	MIT
Number of nodes	12	100
Start Date	25 Jan '05	26 Jul '04
Duration	5 Days	280 days
Avg. contacts per day	846	231
Scanning Rate	2 min	5 min

We use two traces of real mobile device contacts carried by humans: the Bluetooth trace of students at the University of Cambridge, Computer Laboratory [10] and the Bluetooth trace of students and staffs at MIT [11]. We shall refer to these as CAMBRIDGE and MIT, respectively. For MIT, we use the human contacts trace during the first week of the Fall semester (<http://web.mit.edu/registrar/www/calendar0405.html>) representing

a typical week of activity – we note that the only 85 nodes rather than 100 nodes appeared during this period. Table V describes some characteristics of each dataset. These datasets were constructed from mobile device co-location where participants were given Bluetooth enabled mobile devices to carry around. When two devices come into transmission range of the Bluetooth radio, the device logs the co-location with the other device.

We evaluate the effectiveness of the proposed centrality metrics through the computation of ‘message propagation delay’ between nodes on these datasets – we use the first half of the each human contacts trace for training input (i.e. a known historical human contacts trace) and the rest for testing samples (i.e. the future ‘unknown’ human contacts trace). Formally, given a human contacts trace on a time interval $[0, T]$, we use the terms of *training trace* to indicate the first half of the human contacts trace on the time interval $[0, \lfloor T/2 \rfloor]$ and *testing trace* to indicate the human contacts trace on the time interval $[\tau, T]$ where $\lfloor T/2 \rfloor < \tau < T$. In other words, we compute the temporal centrality values of nodes in the dynamic graph $G_{0, \lfloor T/2 \rfloor}^D$ generated from *training trace* and then test whether the computed centrality values are really meaningful with a *testing trace* on the time interval $[\tau, T]$. For example, in CAMBRIDGE, we use the human contact trace during the first 2.5 days for *training trace* and that during the later 2.5 days for *testing trace*.

To measure the testing performance in a quantitative manner, given a *testing trace* on the time interval $[\tau, T]$ to be tested, we formally define the following metrics:

- $\text{PD}(v, u)$: The message propagation delay from node v to node u in the *testing trace*. The message propagation time from v to u specifies how much time has elapsed from the time τ to the time when v first meets u . If there is no contact between v and u , $\text{PD}(v, u)$ is defined as ∞ . We note that $\text{PD}(v, u)$ on the time interval $[\tau, T]$ is different from the temporal shortest path distance $\Delta_{\tau, T}(v, u)$; $\text{PD}(v, u)$ can be computed in a totally independent way, regardless of the underlying dynamic graph. We use $\text{PD}(v, u)$ rather than $\Delta_{\tau, T}(v, u)$ to discuss the effects of the time window size w later since $\Delta_{\tau, T}(v, u)$ is generally changed with w .
- $\text{PD-from}(v)$: The average message propagation delay from node v to all the other nodes in *testing trace*. This can be computed as follows:

$$\text{PD-from}(v) = \frac{1}{|V| - 1} \cdot \sum_{u \in V \setminus v} \frac{1}{\text{PD}(v, u)}$$

Here we assume $1/\infty = 0$. This metric is used to quantify in practical terms how quickly the node

u can communicate with all other nodes at time τ . We test whether $\text{PD-from}(v)$ computed from a *testing trace* increases with v ’s temporal closeness centrality computed from *training trace*.

- $\text{PD-sans}(v)$: The average message propagation delay between all nodes in the *testing trace* except the contacts related to node v . This can be computed as follows:

$$\text{PD-sans}(v) = \frac{1}{(|V| - 1)(|V| - 2)} \cdot \sum_{\substack{u, w \in V \setminus v \\ u \neq w}} \frac{1}{\text{PD}(u, w)}$$

This metric is used to quantify in practical terms how much communication speeds between all pairs of nodes are affected by the node u . We test whether $\text{PD-sans}(v)$ computed from the *testing trace* decreases with v ’s temporal betweenness centrality computed from *training trace*.

We first compute temporal closeness and betweenness centrality of nodes in the dynamic graph $G_{0, \lfloor T/2 \rfloor}^D$ generated from the *training trace* of each dataset where w is set to the finest window granularity, corresponding to the scanning rate of the devices in each dataset (for example, 120 seconds for CAMBRIDGE) and plot them in Figure 6. The computed values are sorted in descending order. This figure clearly shows that there is a small number of nodes which have extremely high temporal centrality, and a large number of nodes that have moderate or low centrality values, across all experiments except for the closeness in CAMBRIDGE. This implies that there exist nodes having high temporal centrality in practice.

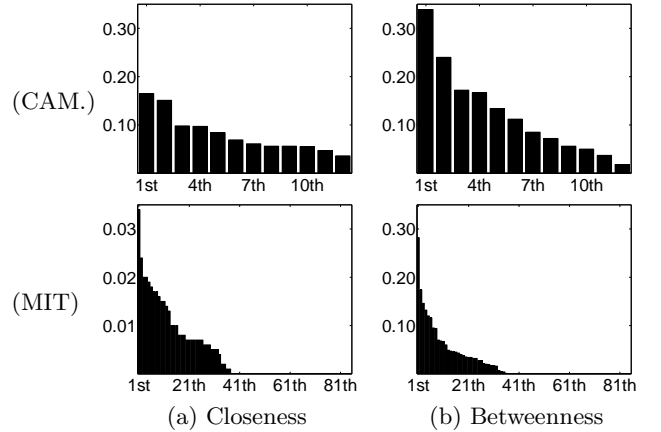


FIG. 6. Temporal centrality distribution of nodes. The computed values are sorted in descending order. For improved visualisation, we use the same range on the y-axis except for the closeness centrality distribution in MIT since the centrality values in MIT are totally different those in the other cases.

We then analyse the correlation between PD-from (or PD-sans) and closeness (or betweenness) centrality over

nodes as to how much useful centrality information is provided. We show that greater temporal centrality in *training trace* is positively related to PD-*from* or PD-*sans* in *testing trace*. We calculate the Kendall tau (rank) correlation coefficients [12] between the PD-*from* (or the reverse of the PD-*sans*) ranking of nodes in *testing trace* on the time interval $[\tau, T]$ and the closeness (or betweenness) centrality ranking in the dynamic graph $G_{0, [T/2]}^D$ generated from *training trace*. The correlation results by varying τ every one hour from $w \cdot (\lfloor T/2 \rfloor + 1)$ are shown in Figure 7. For comparison purpose, we also plot the correlation coefficients between PD-*from* (or PD-*sans*) and the closeness (or betweenness) centrality in the *aggregated* graph and the *average* centrality metrics discussed as above.

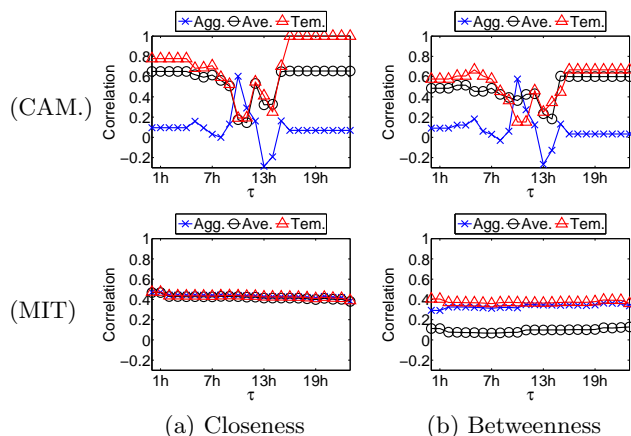


FIG. 7. Kendall tau (rank) correlation coefficients PD-*from* (or PD-*sans*) and closeness (or betweenness) centrality.

For CAMBRIDGE, the temporal centrality metrics are clearly more effective than the centrality in the *aggregated* graph and the *average* centrality metrics; the *closeness* centrality ranking of nodes is particularly identical to their PD-*from* ranking since $w \cdot (\lfloor T/2 \rfloor + 1) + 16$ hours as τ . In MIT, however, the overall correlation coefficients are relatively low (< 0.5). The temporal betweenness centrality is slightly better than the other betweenness centrality metrics while it fails to outperform those for closeness centrality.

Interestingly, for CAMBRIDGE, we found the fluctuation patterns when τ is around $w \cdot (\lfloor T/2 \rfloor + 1) + 13$ hours. We conjecture these fluctuation patterns may result from the human contacts behaviour which is characterized by strong periodicities driven by external calendar cycles. For example, if the network topology during day-time is totally different from that during night-time, the temporal centrality computed from the human contacts trace during day-time is useless to estimate the temporal centrality during night-time. Surely, the use of a long-term training interval to compute temporal centrality will offset this trend. Instead, node centrality is also averaged over time regardless of external calendar cycles. In fact,

the usefulness of temporal centrality is primarily determined by the choice of *training trace*. Thus, given a *testing trace*, we need to choose *training trace* properly so that the temporal centrality computed from *training trace* can be used to estimate PD-*from* or PD-*sans* in *testing trace* well. A reasonable approach for choosing *training trace* is to use the periodic repeatability of human contact patterns. For example, we can use a time interval in the last day as *training trace*, which is the same as the time interval of the given *testing trace*. We plan to study this topic as part of the future work.

Although the estimation of *average* centrality metrics is not as strong as that of *temporal* centrality metrics, they have practically achieved reasonable results compared with *temporal* centrality metrics except for betweenness in MIT. Considering that the time and space complexities of *average* centrality metrics are relatively cheap, we recommend using *average* closeness centrality as an alternative for computing closeness centrality in dynamic networks.

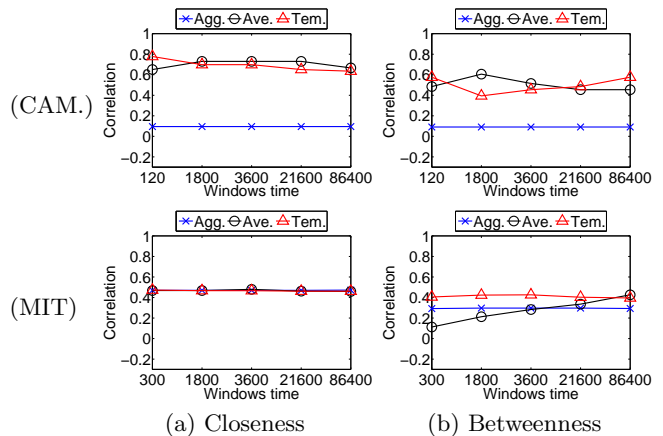


FIG. 8. Kendall tau (rank) correlation coefficients PD-*from* (or PD-*sans*) and the *closeness* (or *betweenness*) centrality by varying w while fixing $\tau = w \cdot (\lfloor T/2 \rfloor + 1)$.

Finally, we discuss the effects of the time window size w which is used for generating a dynamic graph $G_{0, [T/2]}^D$ from *training trace*. As w increases, the temporal characteristics of human contacts are generally underestimated by ignoring time ordering and frequency of contacts within a time window, but the cost of computing centrality values decrease. In particular, the time complexity of the temporal closeness and betweenness centrality computation in a *dynamic network* $G_{i,j}^D = (V, E_{i,j})$ can be dramatically improved since m generally dominates the overall time complexity of the centrality computation where $m = j - i$. If m can become a constant by increasing w sufficiently, the total running time of both the temporal closeness and betweenness computation for all nodes in V is $O(|V|^2)$. Figures 8 shows the effects of varying w from 2 minutes for CAMBRIDGE (or 5 minutes for MIT) to 24 hours. To demonstrate this we fix

$$\tau = w \cdot (\lfloor T/2 \rfloor + 1).$$

Unlike our expectation, the Kendall tau (rank) correlation coefficients between the PD-from (or the reverse of the PD-sans) ranking of nodes and the closeness (or betweenness) centrality ranking are almost stable with w although the correlation coefficients between the PD-from ranking of nodes and the temporal *closeness* centrality ranking for CAMBRIDGE slightly decreases with w . This is an indication that a certain rough approximation of node centrality is comparable to that in a fine-grained dynamic network. Moreover, the *average* centrality metrics produced results similar to the temporal centrality metrics as w increases. In fact, in these datasets, the effects of temporal centrality computed from a historical human contact trace may seem rather limited to deal with identifying the topologically important nodes in future contacts since the network topology of a human contact network changes rapidly over time. So we need to develop techniques to further improve prediction accuracy of centrality.

CONCLUSION

We proposed new temporal centrality metrics based on a simple but powerful model, the *time-ordered graph*, which can reduce a dynamic network to a static network with directed flows. The proposed centrality metrics are designed to overcome the limitation of the existing metrics which require a priori knowledge of future contracts which is not available in practice – the temporal centrality measure should be defined to represent an average of all time aspects in order to avoid the bias observed during a specific time interval.

We demonstrated the feasibility of the proposed temporal centrality metrics by applying them to a number of interesting edge cases, such as where the network connectivity depends on a small number of highly mobile vertices or edges. This intensive simulation results show the concept of temporal centrality is necessary theoretically.

Finally we applied the proposed centrality and techniques to two real-world dynamic graphs of human contact networks and then discuss the implication of temporal centrality metrics in the real-world. However, in practice, the effects of temporal centrality derived from a historical human contact trace may seem rather limited to

deal with identifying the topologically important nodes in future contacts although temporal centrality still better than the other static centrality metrics. We found that it is very hard to estimate node centrality in the human contact network since the network topology of a human contact network changes rapidly over time. As part of the future work, we plan to perform a more comprehensive analysis of the choice of training samples to provide better results of temporal centrality in future network topology.

ACKNOWLEDGMENT

The first author's research is funded by Northrop Grumman Systems Corporation. The contents of this article do not necessarily express the views of Northrop Grumman Systems Corporation.

-
- [1] M. E. J. Newman, A. L. Barabási, and D. J. Watts, eds., *The Structure and Dynamics of Networks* (Princeton University Press, 2010).
 - [2] R. Albert, H. Jeong, and A. Barabasi, *Nature* **406**, 378 (2000).
 - [3] S. Nagaraja and R. Anderson, in *The Fifth Workshop on the Economics of Information Security* (2006).
 - [4] J. Tang, M. Musolesi, C. Mascolo, V. Latora, and V. Nicosia, in *Proceedings of the 3rd Workshop on Social Network Systems, SNS '10* (ACM, New York, NY, USA, 2010) pp. 3:1–3:6.
 - [5] D. Kempe, J. Kleinberg, and A. Kumar, *Journal of Computer and System Sciences* **64**, 820 (2002).
 - [6] A. Ferreira, *Network*, *IEEE* **18**, 24 (2004).
 - [7] V. Kostakos, *Physica A: Statistical Mechanics and its Applications* **388**, 1007 (2009).
 - [8] J. Tang, M. Musolesi, C. Mascolo, and V. Latora, *SIGCOMM Computer Communication Review* **40**, 118 (2010).
 - [9] T. Opsahl, F. Agneessens, and J. Skvoretz, *Social Networks* **32**, 245 (2010).
 - [10] "Haggle Project," <http://www.haggleproject.org> (2009).
 - [11] N. Eagle and A. (Sandy) Pentland, *Personal and Ubiquitous Computing* **10**, 255 (2006).
 - [12] A. D. W. Jerome L. Myers and R. F. L. Jr., *Research Design and Statistical Analysis* (Lawrence Erlbaum Associates, 2003).