Dynamic Graph Properties of Mobile Networks under Levy Walk Mobility

Berk Birand*, Murtaza Zafer†, Gil Zussman* and Kang-Won Lee†
*Department of Electrical Engineering
Columbia University, New York, NY
{berk,gil}@ee.columbia.edu
†IBM T.J. Watson Research Center
Hawthorne, NY
{mzafer,kangwon}@us.ibm.com

***Abstract*—The performance of many algorithms in dynamic networks depends on the properties of the underlying graph representing the network. Since such a graph is inherently time-varying, quantifying the change in its structure is important for understanding the behavior of higher-layer network algorithms. In this paper, we study change in the dynamic graph structure of mobile wireless networks that evolve over time due to node mobility. We define several graph evolution metrics and evaluate them through extensive numerical simulations under Levy Walk mobility, which has been shown previously to have similarities to human mobility patterns. Based on the mean and distribution of these metrics, we obtain important insights into the properties of the evolving graph generated by Levy Walk mobility, and then compare the results to the Random Waypoint mobility model. Finally, we discuss the effects of the rate of graph change on the performance of network applications such as data routing and flooding. Our results suggest that the proposed metrics are viable for quantitatively measuring the magnitude of change in a sequence of evolving graphs.**

***Index Terms*—Mobile networks, Truncated Levy Walk mobility model, evolving graphs, routing overhead, flooding protocol.**

I. Introduction

A mobile wireless network can be naturally represented as a *dynamic graph* whose nodes are mobile devices and whose edges model instantaneous wireless links present between the nodes. Unlike a static network, the structure of a mobile network changes over time. Thus, a mobile network can be abstractly viewed as a time-sequence of graphs.

Wireless networking algorithms, such as scheduling and routing, operate on evolving graphs. For example, throughput optimal transmission scheduling algorithms find a matching or an independent set in a snapshot of the graph [23]. Similarly, routing algorithms seek the shortest path in these graphs as they evolve [12]. Naturally, the properties of the evolving graph directly affect the performance of network algorithms. For example, consider the problem of routing in a mobile network. Several approaches exist ranging from proactive algorithms that store and update routing paths, to reactive algorithms that generate paths for every request [15], [24]. For a given scenario, the performance of these algorithms depend heavily on the mobile activity of the underlying network nodes.

To gain a fundamental understanding into the performance of algorithms in mobile networks, it is therefore important to understand the evolution of the underlying network graph. In this paper, we focus on measures for quantifying graph change. Using these measures, we evaluate the change in evolving graph sequences generated by Levy Walk and Random Waypoint mobility models. The Levy Walk model has been recently shown to have similarities to practical human mobility behavior. For example, [20] found that traces obtained from volunteers moving in outdoor settings could be fitted with a model in which the flight lengths followed a truncated Levy distribution. The Random Waypoint model has been a standard mobility model in the literature [7], [15], [26]. In addition, we also consider an evolving graph sequence generated by a Markovian model in which links evolve over time as a two-state Markov process [10].

The Levy Walk mobility model approximates real mobility patterns well. However, its analysis in the context of evolving graphs is generally not tractable due to the lack of a closed form distribution function for the flight lengths. Hence, we take a numerical approach. We first introduce the new metrics *Graph-Edit-Distance* and *Graph-Inner-Product* for expressing the magnitude of change in evolving graphs. These metrics are motivated by a representation of graphs as vectors and by graph similarity studies in [11], [22]. A visualization method called the *Edit-Graph* is introduced in order to expose the location and frequency of changes in an evolving graph sequence.

We then derive numerical results on the Levy Walk mobility model that illustrate the change in the network graph as seen by higher layer algorithms. In particular, we show a direct relationship between the parameters of the mobility models and our graph metrics. Using our results, the behavior of an evolving graph sequence can be inferred directly from the model parameters. In addition to the Levy Walk model, the results are also compared to graph changes stemming from the Random Waypoint model.

Lastly, we explain the effects of the graph change rate on networking algorithms. Through the use of our graph metrics, we propose an adaptive policy for predicting control packet overhead for various families of routing algorithms. A similar

978-0-7695-4469-4/11 $26.00 © 2011 IEEE
DOI 10.1109/MASS.2011.36

relationship with graph metrics is also explored with respect to the performance of flooding algorithms.

The main contribution of this paper is introducing a new representation of evolving graphs as sequences in vector spaces. The advantage of this view is demonstrated by the introduction of novel graph metrics that are applied to well-known mobility models.

This paper is organized as follows. We begin by an exploration of related work, followed in Section III by introducing our new viewpoint on graph evolution, the associated graph change metrics, and the mobility models that will be studied. Extensive simulation results on the application of these metrics to the mobility models appear in Section IV. We apply the results of this quantification to networking protocols to predict the performance of routing and flooding algorithms in Section V. Finally, Section VI concludes the work and discusses future directions.

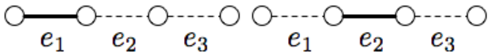
II. Related work

There has been a growing interest in evolving graph models, both from an analytical and an experimental point of view. From the experimental side, several mobility models have been analyzed and compared in the survey paper [7]. Most of the previously known models were variants of the Random Waypoint model [15], or the Random Direction model [19], [21]. However, neither model is believed to adequately represent the behavior of real mobility patterns. This led to the exploration of models with heavy-tail behavior. First, the inter-contact time of most traces were shown to have a heavy-tail distribution [6], [8]. Then, the movement patterns were shown to have flight length and pause times that can be approximated by the Levy Walk [20].

Heavy-tailed models such as the Levy Walk received a lot of attention from an experimental perspective. However, their rigorous analysis is lacking, due to their mathematical intractability. By simplifying the dynamics of evolving graphs, properties of algorithms like gossiping and covering were considered in the context of general Markovian models [2] and adversarial models [16]. Bounds on the flooding time were obtained in more specific models, in which each edge evolves independently of the others and according to a Markov chain [10]. These bounds were further refined in [3].

III. Metrics and Mobility Models

Graphs that change over time can be naturally modeled as an evolving graph sequence. Assume that time is slotted and that a changing network is sampled at discrete time slots. At each such sample, we consider the state of the nodes in the network and derive a network state using a connectivity model such as the disk model (where two nodes are connected by a link, if they are within a distance r of each other) [17][1]. A collection of such static graph snapshots forms an evolving graph sequence [2], [12]. There can be many reasons to the evolution of the graph structure of a mobile network, including changes in transmission power levels or changes in interference, due to communication effects. The focus of this paper is the stochastic change that occurs over time due to the mobility of nodes.



(a) G_0 (b) G_1

Fig. 1. Sample evolution of an evolving graph sequence $G_t \in \mathcal{G}^3$. Existing edges $e_n \in E_t$ are marked in bold, and non-existing edges $e_n \notin E_t$ are dashed.

A. Graphs as Vectors

We first introduce an interpretation that considers graphs as points in a vector space. Under this view, an evolving graph sequence is simply a sequence of points in this vector space. The regular metrics such as norms and inner products can then be defined over this vector space to analyze these sequences.

To illustrate this notion, consider a simple example in Fig. 1. Let $\mathcal{G}^3$ denote the space of all possible 3-node graphs (i.e., the set of graphs with 1-link, 2-links, or 3-links). Consider an evolving graph sequence that cycles between the three graphs consisting of a single link (i.e., at each time slot, only one edge in a 3-node graph is up). Assume that initially edge e_1 is up and denote the corresponding network graph $G_0 \in \mathcal{G}^3$. We associate this graph with the vector $v(G_0) = (1, 0, 0)$ which is a 3-element vector representing the state of all possible links. At time $t = 1$, edge e_2 comes up while e_1 goes down, yielding $v(G_1) = (0, 1, 0)$. As the sequence evolves, we obtain a sequence of graphs $\{G_t\}$ in the space of graphs $\mathcal{G}^3$ and the corresponding sequence of vector points $\{v(G_t)\}$.

Formally, let $\mathcal{G}$ be the space of graphs $G = (V, E)$ with $|V|$ nodes and $|E| = \binom{|V|}{2}$ be the total number of possible edges (labeled $e_1, \cdots, e_{|E|}$). An evolving graph sequence can be represented as the sequence $\{G_t\}$, where each $G_t \in \mathcal{G}$. We define the corresponding function $v(G_t)$ as follows:

Definition 1 (Graph Vector): Let $G \in \mathcal{G}$. Then, $v(G) : \mathcal{G} \to R^{|E|}$ is the vector of edges associated with the graph G. That is, if $v(G_t)_i = 1$, then $e_i \in E_t$.

B. Graph Change Metrics

Using the vector representation, we now present rigorous metrics of graph change based on vector space properties. The notation $|| \cdot ||_2$ denotes the ℓ_2-norm in a vector space.

Definition 2 (Graph Edit Distance [11], [22]): Given two graphs G_1 and G_2, the edit distance between them is given by:

$$\text{edt}(G_1, G_2) := ||v(G_1) - v(G_2)||_2^2.$$

Definition 3 (Graph Normalized Edit Distance): Given two graphs G_1 and G_2, the normalized edit distance between

[1] Although it is well-known that the unit disk model may not capture realistic connectivity relations [18], it still provides a reasonable abstraction. Extending the results to general SINR-based constraints is a subject for further research.

them is given by:

$$\text{nedt}(G_1, G_2) := \frac{||v(G_1) - v(G_2)||_2^2}{|E|},$$

where $|E| = \binom{|V|}{2}$.

The edit distance metric measures the number of link changes that have happened between two graphs G_1 and G_2. Namely, it counts the number of links that have become active in addition to the number of links that have become inactive. The normalized metric expresses this quantity as a value between 0 and 1. If $nedt(G_1, G_2) = 1$, the two graphs G_1 and G_2 must be complements of each other. If $nedt(G_1, G_2) = 0$, the two graphs are identical. The range of values between 0 and 1 represent the order of difference between the two graphs.

Below, we introduce another metric for further characterizing the similarity between two graphs.

Definition 4 (Graph Normalized Inner Product): Given two graphs G_1 and G_2, the normalized inner product between them is given by:

$$G_1 \cdot G_2 := \frac{v(G_1) \cdot v(G_2)}{||v(G_1)||.||v(G_2)||}.$$

The above metric is the cosine of the angle between the two vectors $v(G_1)$ and $v(G_2)$ and satisfies $0 \leq G_1 \cdot G_2 \leq 1$. If the *normalized inner product* is 1, the two corresponding vectors in the vector space are perfectly aligned, and the graphs G_1 and G_2 are identical. On the other hand, if the *normalized inner product* is 0, the two graphs G_1 and G_2 have no common edges.

While the two metrics *normalized edit distance* and *normalized inner product* measure change in the graph there is a subtle difference between them. *Normalized edit distance* measures the magnitude of change that has happened between graphs G_1 and G_2. On the other hand, the *normalized inner product* measures the similarity of two graphs G_1 and G_2. To illustrate this, consider a sparse graph consisting of a large number of nodes, but with very few edges (say just one edge). If this edge goes down and a new edge comes up, the *edit distance* of this graph will be small (only 2) implying a small change in the edges. However, the *inner product* will be zero indicating that the new graph topology has no edges in common with the old topology.

Another variant of the inner product metric defined above is the *link persistence* metric. By normalizing with just the number of edges in the first graph, this metric represents the probability that a randomly selected edge in the first graph also exists in the second graph.

Definition 5 (Link Persistence): Given two graphs G_1 and G_2, the link persistence probability between them is given by:

$$\text{lp}(G_1, G_2) := \frac{v(G_1) \cdot v(G_2)}{||v(G_1)||}.$$

It is immediately possible to see the use of the above metrics. If two graphs are consecutive items of an evolving graph sequence, these metrics represent the quantity of change in the graph sequence. If the metrics indicate a small change from an item of the sequence to the next, the evolving graph sequence changes slowly, whereas if the metric is large, the evolving sequence changes rapidly.

TABLE I
PARAMETERS FOR THE MOBILITY MODELS.

Name	Symbol	Model	Default value
Flight length Levy parameter	α	TLW	1
Pause time Levy parameter	β	TLW	1
Truncation flight	τ_l	TLW	
Truncation pause	τ_p	TLW	
Minimum Velocity	v_{min}	RWP	
Maximum Velocity	v_{max}	RWP	
Communication radius			500
Simulation area			1500

The metrics discussed above represent change between two graphs as a single scalar value. However, they do not provide information about the nature and locality of this change. To capture this, we consider the notion of an *edit graph* defined as follows.

Definition 6 (Edit Graph): The edit graph $\triangle G$ is the symmetric difference between the edge sets of two graphs. Formally, let $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ be two graphs, then, if $e \in \triangle G$, either $e \in E_1 \setminus E_2$ or $e \in E_2 \setminus E_1$.

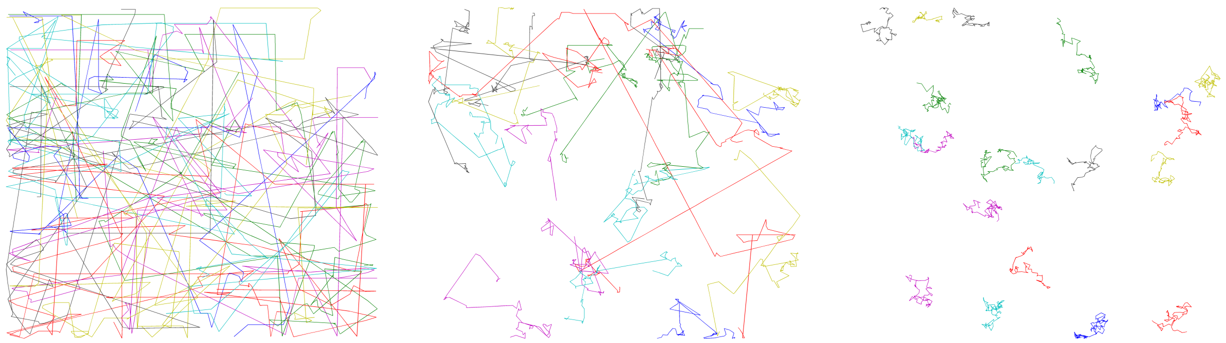
An *edit graph* carries more information about the change between two graphs including the locality of changes. Therefore, its properties affect the performance of some network algorithms which depend not only on the frequency of changes but also on the locations of the changes. This effect will be discussed later when studying flooding time in dynamic networks.

C. Mobility Models

While the metrics are useful to measure change in any sequence of graphs, this paper considers sequences generated by the following three models: Levy Walk mobility, Random Waypoint mobility and the Edge-Markovian model. A survey of several models can be found in [7].

1) Truncated Levy Walk (TLW) Model: Among the many mobility models that attempt to find a balance between analytical tractability and practical relevance, the Levy Walk mobility model has recently received a lot of attention, due to its relatively good representation of reality [5], [13], [20], [25]. The heavy-tail nature most closely captures human motion. Namely, it captures that humans move mostly in a given locality and occasionally make large jumps. The Truncated Levy Walk model is a variant that has been successfully applied to several experimental traces [20].

In this model, we assume a square simulation area in which nodes are free to move. When they reach a corner of the area, they are bounced off with the same angle with which they arrived. The Truncated Levy Walk model is described by a tuple of four variables $(l, \theta, \Delta t_f, \Delta t_p)$. The flight length is captured by a heavy-tailed random variable l. This variable



(a) $\alpha = 0.1$

(b) $\alpha = 1$

(c) $\alpha = 2$

Fig. 2. Motion traces of the Truncated Levy Walk model for different scale parameter α values. In all three instances, the initial locations of the nodes are identical.

is sampled from a Levy distribution with parameter α and its probability distribution function $p(\cdot)$ has the following property:

$$p(l) \sim |l|^{-(1+\alpha)},\ l < \tau_l,\ 0 < \alpha \leq 2.$$

If the sampled value of l is greater than the truncation factor τ_l, a new value is generated. The time it takes for the node to cover the flight length is taken from the relationship $\Delta t_f = kl^{1-\rho}, 0 \leq \rho \leq 1$, with k and ρ constants. The direction of the flight length θ is taken to be uniform from $[0, 2\pi]$.

Once the node arrives to its destination, its pause time Δt_p is also taken from a truncated Levy distribution with parameter β:

$$p(\Delta t_p) \sim |\Delta t_p|^{-(1+\beta)},\ \Delta t_p < \tau_p,\ 0 < \beta \leq 2.$$

The movement is repeated until the end of the simulation period. For more information about the Truncated Levy Walk model, see [20]. The parameters for the various models that we use are summarized in Table I along with their default values.

2) Random Waypoint (RWP) Model: The Random Waypoint model [15] has been extensively studied in the literature . It is used widely in the mobile networking literature due to its intuitive definition and mathematical tractability [15]. Many algorithms, including DSR and OLSR have been evaluated using this model. It is therefore important to understand its properties with respect to our graph metrics. Our aim in this work is to use this model as a benchmark for the Trucated Levy Walk model.

In RWP, each node picks a destination uniformly in the simulation area and travels to that point from its current position. The velocity is sampled uniformly from an interval $[v_{min}, v_{max}]$ and the duration of the flight is computed from this velocity. Once the destination is reached, a random pause time is selected from a uniform distribution $[0, \tau_p]$. Note that in this paper, we are mostly concerned with the behavior of such algorithms over a finite time-interval. Therefore, steady-state issues mentioned in [26] do not apply. Furthermore, we always set a positive v_{min}, which further helps with having a proper steady state behavior.

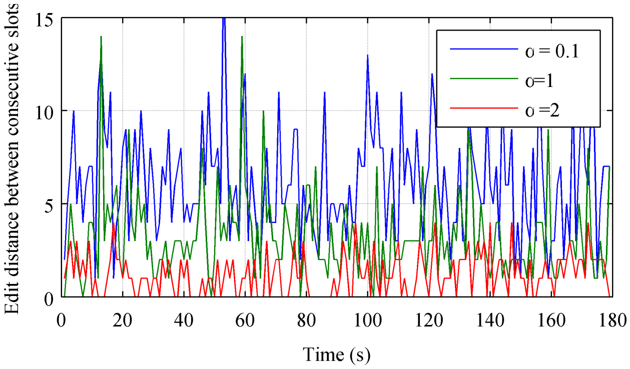


Fig. 3. Evolution of the edit distance over time of the TLW model for different values of the α parameter.

3) Edge-Markovian Model (eMEG): Although not a mobility model, the edge-Markovian graph is a specific evolving graph structure that could be used for modeling change in dynamic networks [9], [10]. In this model, each edge is governed by a simple two-state Markov chain that dictates whether an edge is present or not. The evolution of the chain is Markovian, in the sense that if an edge exists, it disappears in the next slot with probability q, and if it does not exist, it appears with probability p. It is easy to see that at steady-state, this model converges to an Erdös-Renyi random graph model, where an edge exists with probability $p/(p+q)$.

IV. Quantifying Change in Graphs

In this section, we utilize the metrics introduced earlier to study graph change in an evolving graph sequence generated by the Truncated Levy Walk model.

A. Levy Walk Motion

Motion traces of the Truncated Levy Walk model shown in Fig 2(a)-2(c) illustrate the range of motion that can result from this model. These traces have been generated using the Truncated Levy Walk simulator for Matlab provided by [1] that was improved to support longer simulation runs in order to observe steady-state behavior. We plotted the mobility traces

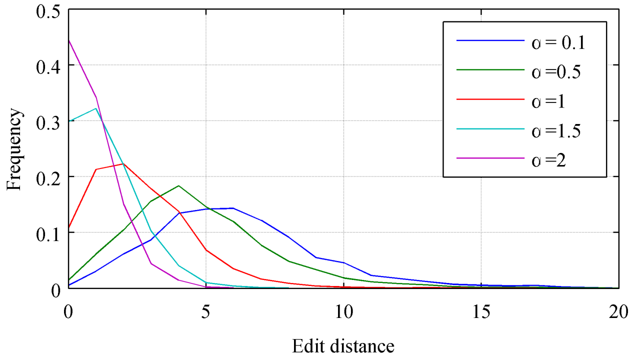


Fig. 4. Distribution of edit distance for different values of α.

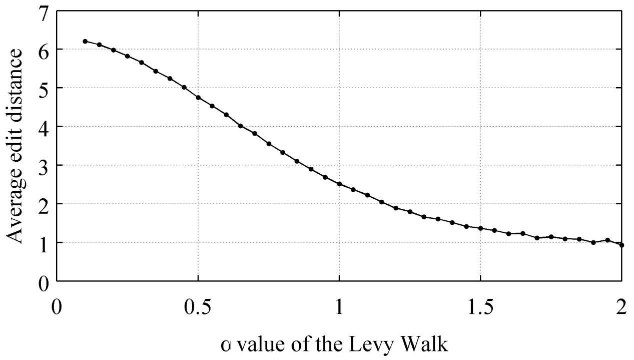


Fig. 5. Average value of the edit distance for given α values for flight length, when $\beta = 1$.

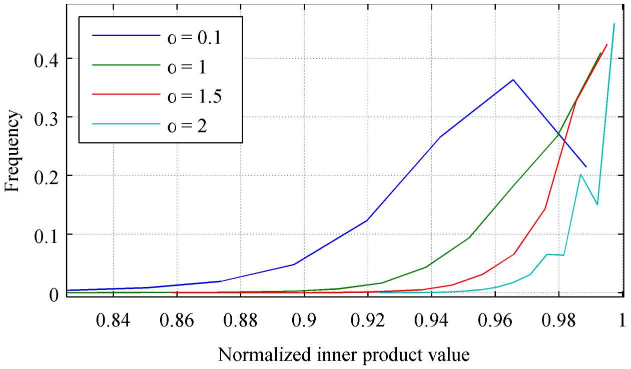


Fig. 6. Distribution of the graph inner product for 4 values of α.

for 20 nodes over a period of three minutes for $\alpha \in \{0.1, 1, 2\}$. The Levy parameter for the pause time is $\beta = 1$, the truncation factors for the flight length and the pause time are chosen to be the simulation area and 60 minutes, respectively.

Note that choosing different values of α can lead to dramatically different kinds of motion under the TLW model. Choosing low values of α makes the movement similar to that of a RWP model or of a Random Direction model [19], [21]. The flights are typically very long and cover the entire simulation area. When α approaches 2, the motion becomes similar to Brownian motion, since for $\alpha = 2$ the Levy distribution reduces to a Gaussian distribution. Hence, in such cases, nodes do not move too far from their initial positions. For values in-between, say $\alpha = 1$, the motion carries the characteristic properties of a Levy walk. They perform mostly short flights around the current position and occasional long flights. In short, for low α values, the nodes have a high degree of mobility and move faster, while as α increases the mobility decreases.

B. Graph Metrics

We analyze the evolving graph produced by Levy Walk mobility using the *graph edit distance* metric. This metric represents by a single value the number of edges that have changed between two instances of a graph sequence. In Fig. 3, we plot the *edit distance* as a function of time for sample realizations of Truncated Levy Walk traces. The edit distance rapidly fluctuates in time with low (high) values of α having larger (smaller) edit-distance values. To better understand this evolution, the distribution of the edit distance for various values of α is shown on Fig. 4. These simulation were run over 2000 simulation minutes to obtain a steady-state distribution, and the other simulation parameters (β, etc) were the same as above. Notice that the mean and the variance are different on all accounts.

In order to further understand the effect of the α parameter on the Levy walk, we plotted in Fig. 5 the mean edit distance versus α. The figure shows the intuitive trend that as α becomes larger, the mean edit distance drops. However, the drop is significant for low α values and decreases as α increases (i.e., the curve flattens out). As α approaches 2, the graph becomes less dynamic, almost completely static in some cases. For values of α that are smaller, the nodes wander further away from the point where they have started. It is worth noting that this is due to the heavy-tail nature of the Levy walk. When $\alpha = 2$, we are essentially sampling from a normal distribution and hence have a Brownian motion pattern.

The findings of Fig. 5 suggest that by knowing the α parameter of the Levy walk, it is possible to anticipate the magnitude of the change in the graph. For certain values, the graph will change dramatically and the protocols running on top of this network must adapt to changes more quickly. As will be shown in Section V, this can have important consequences for predicting the flooding times and on the message overhead of routing algorithms.

We turn our attention to the normalized inner product between two consecutive graphs in a sequence for different values of α. In all instances in this section, the pause time between flights has been set to the same Levy distribution with parameter $\beta = 1$. In order to understand how α affects the inner product, the distribution of the inner products for four values of α is plotted in Fig. 6. We note that small values of α have a lower mean and are, therefore, indicative of a larger difference in the graphs. For $\alpha = 2$, the inner product values are very close to 1, meaning that the graphs are very similar.

Our results suggest that the normalized inner product is a viable metric for quantitatively measuring how "close" two

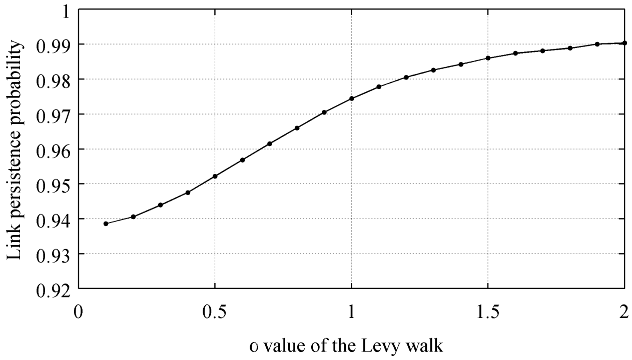
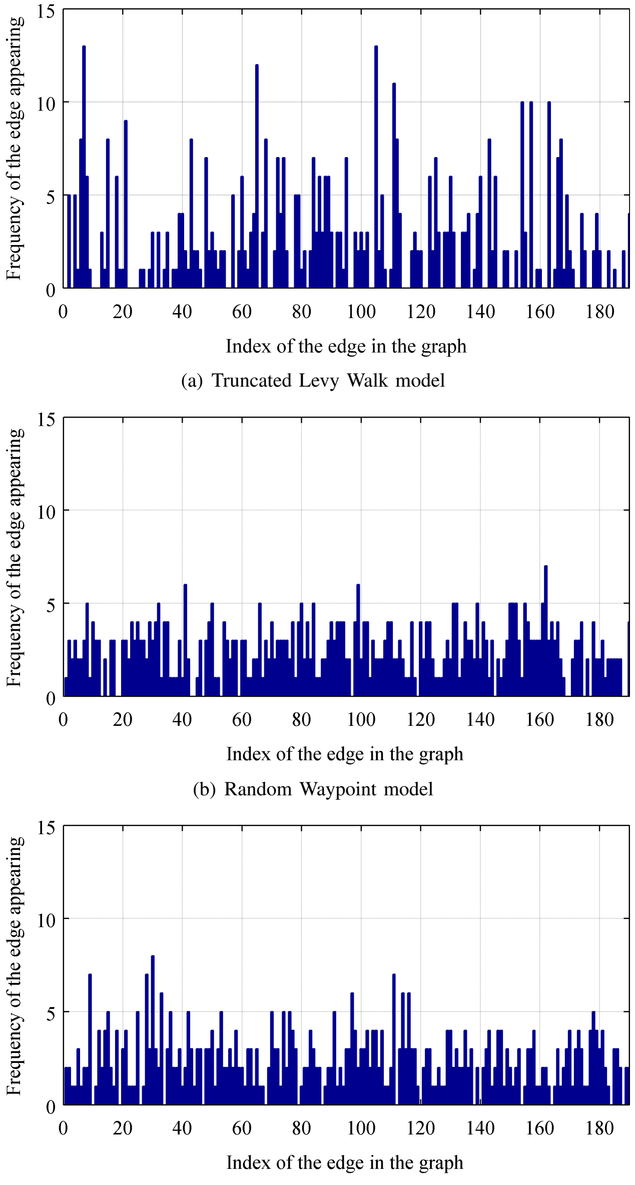


Fig. 7. Average link-persistence probability metric for different values of α.



(a) Truncated Levy Walk model

(b) Random Waypoint model

(c) edge-Markovian model

Fig. 8. Frequency of an edge appearing in the edit graph for the Truncated Levy Walk model, Random Waypoint model and the edge-Markovian model in a simulation that lasts for 3 minutes.

graphs are. Within the context of mobile networks, it is a good metric that can be used to evaluate the effect of the model parameters, such as α in our case, on the underlying evolving graphs sequence.

The link persistence metric also supports the insight obtained by the the normalized inner product. Fig. 7 illustrates the mapping between the α values and the link persistence probability metric. The effect of α on the metric is similar to that of Fig. 5. Moreover, the link persistence probability is fairly high at 0.95 even for small α values. This is indicative that when the nodes take large flights, the link quality does not get affected too much.

C. Edit Graph

The edit graph sequence introduced in Section III further characterizes the change in an evolving graph sequence. While the edit distance is useful in describing the magnitude of changes between time slots, the information about the locality of the changes is not considered. It will be shown in Section IV that the location of the changing edges also plays a significant role in the behavior of some of the higher-layer algorithms.

Since the edit graph sequence is another evolving graph sequence, we will illustrate it using graphical methods[2]. The frequency plot on Fig. 8-(a) shows the frequency of changes in the edit graph of a Truncated Levy Walk with $\alpha = 1$. If an edge appears and disappears very often, it will appear more often in the edit graph and will be represented using a larger bar. We can see from the plots that the distribution of edge appearance is not uniformly distributed in a finite time-window. A small number of edges change very often, while the majority of the edges mostly do not change. This behavior of the Levy walk has implications on networking algorithms that will be discussed in Section V.

As a comparison, we also show the same edge frequency distribution for the edge-Markovian Evolving Graph model introduced in Section III in Fig. 8-(c). The p and q parameters for this model have been chosen such that the edit distance of this graph will be close to that of the Levy walk ($p = q \approx 2.69$). It can be seen that the distribution of the edge appearances for the edge-Markovian sequence is more evenly distributed. Most edges change a predictable number of times.

In order to emphasize the importance of the α parameter and the initial point distribution on the structure of the edit graph, the edge frequency is plotted graphically in Figs. 9(a)-9(c). The nodes are placed according to their initial positions at the beginning of a three minute run interval and the colors of the edges represent the frequency of that edge occuring in the edit graph. A dark color represents an edge that has changed its state very frequently during the run. If an edge does not exist in the edit graph plot, then that edge either has

[2] Animations for the edit graph sequence of this section can be found in [4].

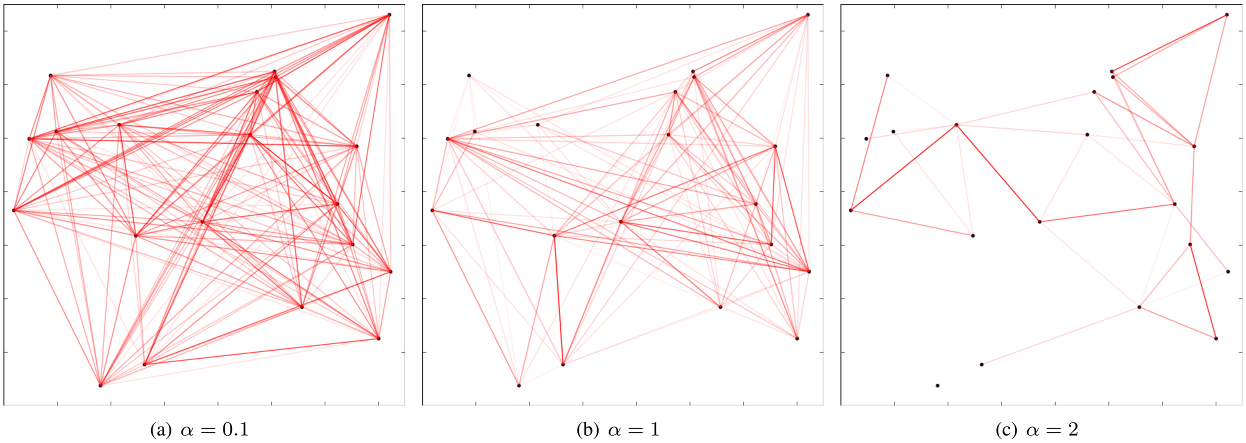

(a) $\alpha = 0.1$
(b) $\alpha = 1$
(c) $\alpha = 2$

Fig. 9. Edges colored according to their frequency in the edit graph for the same initial coordinates and $\alpha \in \{0.1, 1, 2\}$.

never appeared in the original graph or has always existed.

These visualizations highlight the dramatic change that can occur in a graph for different values of α. It was previously established that if α is low, the edit distance will tend to be higher and the graph will change quickly. However, the change is also distributed roughly uniformly among the edges. The same edges keep appearing and disappearing.

As α increases to 1, the graph becomes more dynamic. The changes are more uniformly distributed among the various links but there are still some links that do not change.

Finally, for $\alpha = 2$, both graphs are very sparse. Namely, the nodes that are close to each other initially remain close for the duration of the simulation. The ones that are a mid-distance from each other keep appearing and disappearing, which explains why the edit graph has strong edges even though the total number of edge changes is small. If the initial distance between the nodes is substantial, the nodes either never get connected, or get a very light connection.

The edit graph is therefore an interesting tool that provides statistical information about an evolving graph sequence by conveying information about the locality of the change. Our graph plots indicate which edges are more likely to appear and the effect of the initial distribution of the edges on the path persistence.

D. Comparison with the Random Waypoint Model

In this section, we compare the behavior of the Truncated Levy Walk model with that of the Random Waypoint model. Fig. 10 displays the average edit distance between consecutive time slots. The velocity is sampled uniformly from the interval $[v_{min}, v_{min} + 20]$ and the plot is drawn with respect to v_{min}. The average number of edges that differ between two consecutive instances is roughly linear for the plotted interval. This is an interesting phenomenon, since a priori there is no reason for the number of changes in the graph to be directly proportional to the velocity.

It can be seen in Fig. 11 that the changes in the edit graph sequence are roughly evenly distributed among the edges, which has two implications. First, it is a good predictor that flooding will be performed quickly since nodes will most likely contact each other more frequently, resulting in faster information dissemination. On the other hand, link persistence probabilities will be low for many links and connection quality may suffer. It may therefore not be possible to maintain a connection for a long duration of time which may be harmful to certain applications.

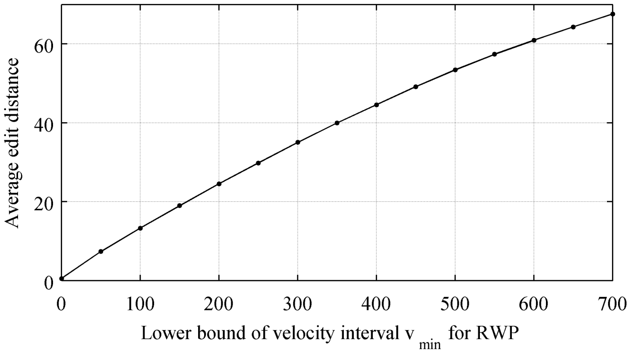


Fig. 10. Edit distance as a function of velocity which is chosen uniformly from $[v_{min}, v_{min} + 20]$.

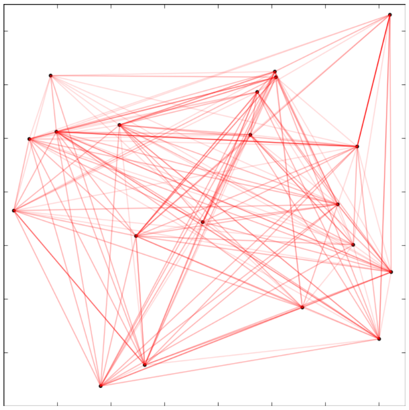


Fig. 11. Edge frequency in the edit graph of the Random Waypoint Model with maximum pause of 10 slots and velocity $v \in [5, 10]$.

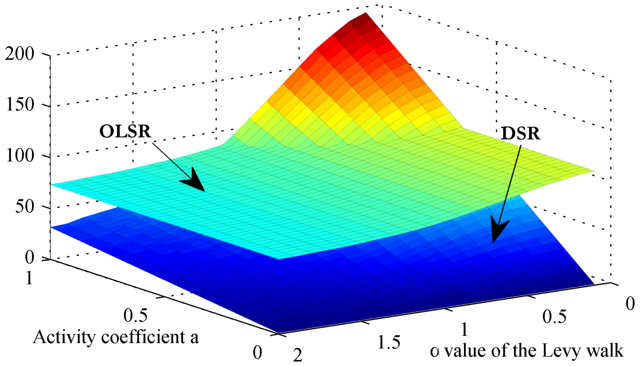


Fig. 12. Number of control packets sent per second for a given α and activity coefficient a for the DSR and OLSR protocols.

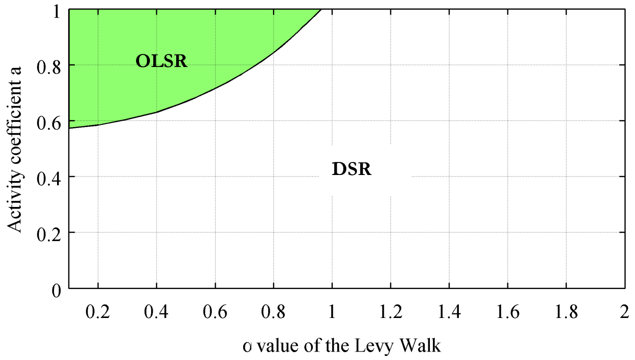


Fig. 13. Areas where OLSR (shaded) has lower control overhead than DSR.

V. Performance Evaluation of Network Protocols

Our motivation for analyzing change in evolving graph sequences is to provide a better understanding of the effects of mobility on network protocols. In this section, we utilize results from the previous sections to study control message overhead of routing algorithms as well as flooding time.

A. *Message Overhead of Routing*

Routing has been extensively studied in the networking literature with the goal of finding the best way to route packets under various network topologies and link costs. In mobile networks, routing algorithms can be broadly classified as reactive or proactive. In reactive protocols, whenever a source has data to send and does not have a cached route, the network is flooded with route-request packets to obtain the best route to the destination. Proactive algorithms, on the other hand, maintain an up-to-date view of the network by regularly exchanging table update messages. In most scenarios, the optimal decision is to use a combination of these two methods through an adaptive policy, an approach that is exploited by hybrid protocols.

In this section, we study control message overhead for the above two protocol classes by utilizing the insights obtained earlier about change in the graphs of mobile networks. Specifically, we consider the Dynamic Source Routing (DSR) [15] and the Optimized Link-State Routing (OLSR) [14] protocols when the underlying mobility model is a Truncated Levy Walk. These are examples of reactive and proactive algorithms, respectively.

A characterization of the control overhead of these two protocols was provided in [24]. For OLSR, the number of control packets generated per second is $K_{OLSR} \cdot \mu + C_{OLSR}$ (i.e., control overhead is proportional to μ), where μ denotes a network mobility parameter which was taken as the link failure probability. For DSR, the number of packets generated is $K_{DSR} \cdot \mu \cdot a + C_{DSR}$, where a represents the node activity defined as the probability that a node is either the source or the destination of an active route. The specific coefficients K_{OLSR}, K_{DSR}, C_{OLSR}, and C_{DSR}, which are non-trivial to estimate, depend on the structure of the network, traffic patterns, and the specific implementation of the algorithm. The coefficient values that we used were obtained experimentally in [24] as $K_{OLSR} = 749$, $C_{OLSR} = 66$, $K_{DSR} = 3150$, and $C_{DSR} = 1$.

The focus of this section is to study the number of control messages produced by these two protocols for various values of α (the Truncated Levy Walk model parameter). This will provide new insights into developing adaptive mechanisms that determines the optimal routing policy solely based on the model parameters α and a. For this purpose, we use the mapping shown in Fig. 7 of Section IV that showcases a mapping from the α-value to the link persistence metric.

Note that while the results are obtained for these specific values, the general shape of the plots would remain the same and the insights would hold. For a given Levy Walk α-value and activity parameter a, it is now possible to estimate whether DSR or OLSR will have lower control overhead.

The number of control messages versus α and a is shown in Fig. 12. Note that the overhead of OLSR is constant over the a values. Using this figure, the selection of protocols that minimizes the overhead is shown in Fig. 13. Under Levy walk mobility, for $\alpha > 1$, DSR is a better option in terms of routing control overhead. This is confirmed by the intuition that when the graph is changing slowly, the regular transmission of control packets is not justified. When $a < 0.6$, DSR has lower overhead than OLSR. As intuitively expected, in networks with low activity or smaller rate of graph change, creating routes on-demand is better than maintaining an up-to-date routing table. In networks that are highly-dynamic and have high activity, OLSR performs better than DSR. As mentioned in [24] since proactive protocols can reuse the routes that have been computed as opposed to recomputing them very quickly.

B. *Flooding Time*

Flooding is used by many network algorithms for quickly disseminating information in networks with unknown topology. Mobile networks form an especially attractive venue for flooding because of their constantly changing nature. It is therefore of both practical and theoretical interest to evaluate the performance of flooding schemes in the context

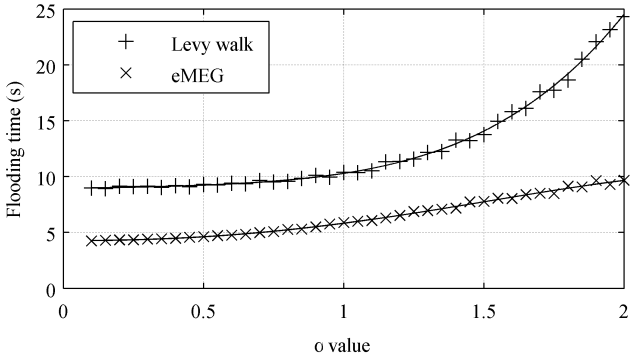


Fig. 14. Flooding time in the Truncated Levy Walk model and the edge-Markovian Evolving Graph model for comparable edit distances.

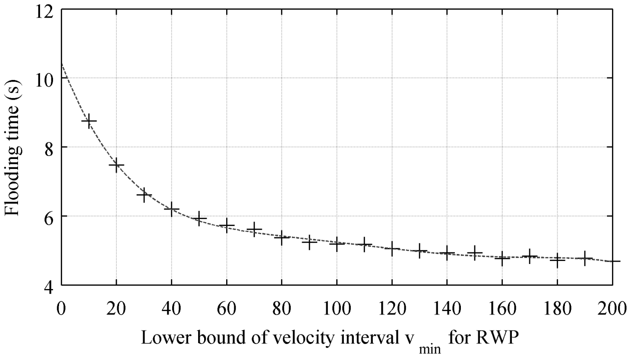


Fig. 15. Flooding time for the Random Waypoint Model.

of changing networks especially for mobility models, such as the Levy Walk.

Intuitively, a network that changes very quickly should, in general, have better flooding performance. This follows from the observation that the more new nodes a given node contacts, the higher the likelihood that it will also run into more nodes that do not have the packet. With respect to graph change metrics, this implies that given a mobility model and some initial conditions, the parameters that cause the graph to have a higher edit distance (or a lower inner product) will have better flooding performance.

This phenomenon can be further explained as follows. The bottleneck in completing flooding in many of our simulations runs were isolated nodes. In most instances, when the graph is disconnected, there is a large connected component that contains most of the nodes and smaller groups of isolated nodes. The flooding time is bounded by the time it takes the isolated nodes to get connected to the main component. If nodes move rapidly, it is more likely that they will get connected to the large component sooner, whereas if they move slowly it will take them longer.

To study flooding, a sequence of evolving graphs was generated and a random node was picked to initiate the flooding. This process was iterated for 2000 simulation runs. The resulting flooding time was observed for various values of the model parameters that govern the amount of change in a graph as expressed by the edit distance.

Fig. 14 presents the results on flooding time for the Truncated Levy Walk model as a function of α. We also plotted the flooding time in an edge-Markovian Evolving Graph model. To make the comparison relevant, the model parameters for the eMEG sequence were computed to have the same edit distance as the TLW model. Thus, for a given α, the magnitude of change on the figure for the TLW and the eMEG models is similar.

The first observation from Fig. 14 is that for larger α values, flooding takes a longer time. This is in line with the results of Section IV-B where larger values of α resulted in smaller edit distances and hence less change. More surprisingly, note that even if they have similar edit distances, the edge-Markovian model always completes flooding faster. This is an important observation and is due to the uniform edge-distribution of the edge-Markovian model outlined in Section IV-C.

Fig.15 presents the results for Random Waypoint model which lead to a similar conclusion. As the velocity of the nodes increases, the edit distance also increases, and the flooding time steadily decreases. Thus, the rate of graph change significantly affects flooding time with a natural inverse relationship where average flooding time decreases as the rate of graph change increases.

VI. Conclusions

We studied the properties of an evolving graph of a mobile network whose structure changes due to node mobility. In particular, we studied the dynamic graph properties of Truncated Levy Walk model and compared it to the Random Waypoint and the edge-Markovian Evolving Graph models. We presented several metrics that quantify graph change in an evolving graph sequence. Through extensive numerical simulations, we computed these metrics for the TLW and RWP mobility models and obtained several new insights into the properties of the underlying dynamic graph. We showed that some parameters of mobility models, such as the α parameter of the TLW model and the velocity of the RWP Model, have a direct relationship with the edit distance metric. For instance, when the α value of the Truncated Levy Walk model approaches 2, the graph becomes less dynamic, and this change is directly captured through our graph edit distance metric. We related our results on the effect of the rate of graph change to the performance of network algorithms, such as control message overhead in data routing and flooding time in dynamic networks. In particular, we showed how flooding time is dependent on the amount and the locality of change in the network, and that it is possible to minimize the control overhead by observing the parameters of the mobility models.

This paper is a first step in understanding the nature of change in evolving graphs sequences. An interesting future research direction would be to evaluate our graph metrics under the Signal-to-Noise-Interference (SINR) constraints in order to have a more realistic model of mobile networks. Although we applied our metrics to mobile networks, evolving graph sequences appear in many other settings. The applicability of our metrics to other domains where change is also pervasive, including social networks and optical networks with link impairments, may guide the design of better algorithms in those domains as well.

Acknowledgments

Research was sponsored in part by the U.S. Army Research Laboratory and the U.K. Ministry of Defense and was accomplished under Agreement Number W911NF-06-3-0001. This work was also partially supported by DTRA grant HDTRA1-09-1-0057, CIAN NSF ERC under grant EEC-0812072, and NSF grants CNS-0916263 and CNS-10-54856.

The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] "Networking research lab," http://research.csc.ncsu.edu/netsrv/.

[2] C. Avin, M. Kouck, and Z. Lotker, "How to explore a fast-changing world (cover time of a simple random walk on evolving graphs)," *Proc. Springer LNCS*, vol. 5125, pp. 121–132.

[3] H. Baumann, P. Crescenzi, and P. Fraigniaud, "Parsimonious flooding in dynamic graphs," in *Proc. ACM PODC'09*, Aug. 2009.

[4] B. Birand, "Mobility animations," http://www.columbia.edu/ bb2408/MobilityAnimations/.

[5] D. Brockmann, L. Hufnagel, and T. Geisel, "The scaling laws of human travel," *Nature*, vol. 439, no. 7075, pp. 462–465, Jan. 2006.

[6] H. Cai and D. Eun, "Crossing over the bounded domain: from exponential to power-law inter-meeting time in MANET," in *Proc. ACM MOBICOM'07*, Sep. 2007.

[7] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Comm. and Mobile Comp.*, vol. 2, no. 5, pp. 483–502, Aug. 2002.

[8] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *IEEE Trans. Mobile Comput.*, vol. 6, no. 6, pp. 606–620, June 2007.

[9] A. E. F. Clementi, F. Pasquale, A. Monti, and R. Silvestri, "Communication in dynamic radio networks," in *Proc. ACM PODC'07*, Aug. 2007.

[10] A. E. Clementi, C. Macci, A. Monti, F. Pasquale, and R. Silvestri, "Flooding time in edge-markovian dynamic graphs," in *Proc. ACM PODC'08*, Aug. 2008.

[11] M. A. Eshera and K.-S. Fu, "A graph distance measure for image analysis," *IEEE Trans. Syst., Man, Cybern.*, vol. 14, no. 3, pp. 398–408, 1984.

[12] A. Ferreira, "Building a reference combinatorial model for manets," *IEEE Network*, vol. 18, no. 5, pp. 24–29, Sep.-Oct. 2004.

[13] M. C. González, C. Hidalgo, and A. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, June 2008.

[14] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Proc. IEEE INMIC'01*, Dec. 2001.

[15] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile computing, Kluwer*, pp. 153–181, 1996.

[16] F. Kuhn, N. Lynch, and R. Oshman, "Distributed computation in dynamic networks," in *Proc. ACM STOC'10*, June 2010.

[17] M. Marathe, H. Breu, H. B. Hunt, S. S. Ravi, and D. J. Rosenkrantz, "Simple heuristics for unit disk graphs," *Networks*, vol. 25, pp. 59–68, 1995.

[18] T. Moscibroda, R. Wattenhofer, and A. Zollinger, "Topology control meets SINR: the scheduling complexity of arbitrary topologies," in *Proc. ACM MOBIHOC'06*, May 2006.

[19] P. Nain, D. Towsley, L. Benyuan, and L. Zhen, "Properties of random direction models," in *Proc. IEEE INFOCOM'05*, Mar. 2005.

[20] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong, "On the levy-walk nature of human mobility," in *Proc. IEEE INFOCOM'08*, Apr. 2008.

[21] E. Royer, P. Melliar-Smith, and L. Moser, "An analysis of the optimum node density for ad hoc mobile networks," in *Proc. IEEE ICC'01*, June 2001.

[22] A. Sanfeliu and K.-S. Fu, "A distance measure between attributed relational graphs for pattern recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. 13, no. 3, pp. 353–362, 1983.

[23] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Automat. Contr.*, vol. 37, no. 12, pp. 1936–1948, 1992.

[24] L. Viennot, P. Jacquet, and T. Clausen, "Analyzing control traffic overhead versus mobility and data traffic activity in mobile ad-hoc network protocols," *ACM/Springer Wireless Networks*, vol. 10, no. 4, pp. 447–455, July 2004.

[25] G. Viswanathan, S. Buldyrev, S. Havlin, M. Da Luz, E. Raposo, and H. Stanley, "Optimizing the success of random searches," *Nature*, vol. 401, no. 6756, pp. 911–914, Oct. 1999.

[26] J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," in *Proc. IEEE INFOCOM'03*, Apr. 2003.