Analysis of Latency of Stateless Opportunistic Forwarding in Intermittently Connected Networks

Chi-Kin Chau, *Member*, *IEEE*, and Prithwish Basu, *Senior Member*, *IEEE*

***Abstract*—Stateless opportunistic forwarding is a simple fault-tolerant distributed scheme for packet delivery, data gathering and information querying in intermittently connected networks, by which packets are forwarded to the next available neighbors in a "random walk" fashion, until they reach their intended destinations or expire. It has been employed in diverse situations, for instance, when (i) the global network topology is not known or is highly dynamic, (ii) the availability of the next-hop neighbors is not easily controllable, or (iii) the relaying nodes are computationally constrained. Data delivery in sensor networks, ad hoc networks, and delay tolerant networks are well-known applications, besides of searching in peer-to-peer networks. A major challenge for stateless opportunistic forwarding is the difficulty to predict the end-to-end latency. To facilitate latency evaluation, we study a simplified model of stateless opportunistic forwarding, namely a "weighted random walk" in a finite graph. This paper makes several contributions towards the analysis of this model: 1) By spectral graph theory we derive a general formula to efficiently compute the exact *hitting* and *commute* times of random walks with *heterogeneous* transition times at relay nodes. Such transition times can model the heterogeneous delivery times and availability periods of the next-hop neighbors. 2) We study a common class of distance-regular networks with a varying number of geographical neighbors, and obtain exact and approximation formulas for the *hitting time* in such networks. 3) Based on these results, we study other sophisticated settings, such as random geographical locations, topology-aware forwarding, and multi-copy random walk forwarding. Our results provide the basic analytical tools for managing and controlling the performance of stateless opportunistic forwarding in finite networks.**

***Index Terms*—Opportunistic Forwarding, Intermittently Connected Networks, Random Walk in Finite Graphs, Spectral Graph Theory**

I. Introduction

Delivering information, gathering data or initiating a query is a common task in most communication and information networks that is handled by the underlying forwarding algorithms. Traditional networks with mostly stationary topology and abundant bandwidth, storage and energy resources can afford a "proactive" routing approach to keep track of the topological changes and construct an accurate routing plan for the forwarding algorithms from time to time. However, in the emerging communication and information networks – so called *intermittently connected networks*, such as wireless sensor networks, a certain class of mobile ad hoc networks, and delay tolerant networks, there may be frequent disruptions of network connectivity. For instance, in sensor networks the relaying transceiver nodes may be performing ad hoc *duty cycling* to save energy, whereas in delay tolerant networks the mobile packet carriers may be temporarily out of reach. In the absence of reliable end-to-end connectivity, flooding or epidemic dissemination to every available neighbor can certainly achieve a good packet delivery ratio. However, these approaches are also very resource-intensive. A simpler approach is to use *stateless opportunistic forwarding*, such that the packet is forwarded to one of the next available neighbors at random, as a "random walk" on the underlying network, regardless of the path traveled by the packet thus far, until it either reaches the desired destination or expires.

This approach is particularly useful in the following situations: (i) the global network topology is not known or is rapidly varying (e.g., in mobile ad hoc networks); (ii) the presence or availability of the next-hop neighbors is not easily controllable (e.g., in *store-carry-forward* networks, and random duty cycling sensor networks); (iii) the relaying nodes are computationally constrained devices that are incapable of supporting highly sophisticated routing strategies (e.g., in sensor networks with low-cost and low-power devices); and (iv) load balancing of traffic and avoidance of single-points of failure are desirable goals. Furthermore, stateless opportunistic forwarding can often serve as a basic building block for devising more sophisticated strategies, such as multi-copy or topology-aware opportunistic forwarding strategies.

Despite the fact that stateless opportunistic forwarding is simple to implement, it is generally difficult to predict its end-to-end latency performance, because packets may (legitimately) travel in loops, and follow different paths from time to time. Although there have been several studies to suggest the usefulness of stateless opportunistic forwarding in intermittently connected networks (e.g., [1]–[4]), there appear to be insufficient analytical tools to assist the management and control of its performance.

We note that the study of stateless opportunistic forwarding in such networks is challenging. In some intermittently connected networks, such as pocket-switched networks, more complicated dynamic graph models have been studied [5], [6]. In contrast, in this paper we aim at providing a simple general non-dynamic model (by a random walk on finite graphs) for

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defense and was accomplished under Agreement Number W911NF-06-3-0002.l. 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. C.-K. Chau is also supported by the Institute for Infocomm Research.

C.-K. Chau is with the Institute for Infocomm Research, Singapore 138632, Singapore, and also with the Computer Laboratory, University of Cambridge, Cambridge CB3 0FD, U.K. (e-mail: ckchau@i2r.a-star.edu.sg).

P. Basu is with Raytheon BBN Technologies, Cambridge MA 02138, USA (e-mail: pbasu@bbn.com).

several representative scenarios of intermittently connected networks specified by certain attributes of the graph edges (as described in the next section).

For this simplified model, we present exact analytical formulas to predict the latency of a generic type of stateless opportunistic forwarding in finite networks. Our approach of using a non-dynamic model to approximately capture the properties of edge dynamics of dynamic networks in an *expected value* sense simplifies the problem of mean latency estimation. The benefit of this style of analysis is to reduce the dependence on simulation to ascertain performance[1], and produce more accurate expressions for more sophisticated design optimization. Furthermore, our exact results are more precise than the results that are yielded by asymptotic analyses in terms of order of magnitude; the latter typically apply to very large networks, whereas most practical wireless networks are often moderate in size. In particular, we derive closed-form expressions for the latency of opportunistic forwarding in specific useful topologies, such as distance-regular networks with varying numbers of geographical neighbors.

A. Motivating Scenarios

To motivate our study, we present three representative scenarios that benefit from stateless opportunistic forwarding, which leads to a general problem formulation presented in Sec. II.

1) *Duty Cycling Sensor Networks*: In a network of battery-powered nodes for a delay-tolerant application, the relaying nodes carry out *random* or *pseudo-random* duty cycle schemes to conserve energy [7]–[9]. For low rate of duty cycling (or wake-up probability), the network is likely to be highly disconnected, and hence traditional routing protocols that rely on freshly gathered topology knowledge may be unsuitable. We are interested in the use of stateless opportunistic forwarding, by which the packets are forwarded to the next-hop neighbor that is the first to wake up[2].
2) *Store-Carry-Forward Networks*: In so-called "pocket switched networks" [1], we consider a network where the nodes are *deposit sites* that can store messages, and have fixed geographical locations. There are some mobile message carriers (e.g., vehicles) that randomly wander from one deposit site to another. Every mobile message carrier has a limited reachable region and the messages can only be forwarded within a certain geographical region by a single mobile message carrier. In such networks, stateless opportunistic forwarding is conveniently employed, by which the packets are picked up by the first nearby carrier that approaches the respective deposit site, and then are off-loaded to the deposit site where the carrier stops next.
3) *Searching in Intermittent Peer-to-Peer Networks*: Consider a collection of peer nodes that may be online or offline momentarily. A query generated by a node may be forwarded to another node that is online simultaneously, provided by an *a priori* member list at each peer node. By stateless opportunistic forwarding, a query will be opportunistically forwarded among the online peer nodes, even when the initiating node stays offline.

We remark that there are other scenarios of intermittently connected networks where stateless opportunistic forwarding can be used. The above scenarios can be modeled naturally by a random walk in a finite graph. We focus on deriving analytical results for its latency, such as:

1) *Hitting time* (or access time): The expected time of a packet from the source to reach (or *hit*) a certain destination. This can model the average packet delivery time.
2) *Commute time*: The expected round-trip time experienced by a packet between the source and the destination (and back to the source). This can model the query return time to a specific node.
3) *Cover time*: The expected time of a packet from the source to visit every other node in the network. This can give an upper bound on the packet delivery time to an unknown group of destinations in the network.

B. Related Work and Our Contributions

The latency of random walk has been studied extensively in the literature. In this section, we present the related work and highlight our contributions in the context of stateless opportunistic forwarding in intermittently connected networks.

There have been a number of proposals of using random walk as a viable and robust forwarding mechanism in various intermittently connected networks [**?**], [2], [3], [10]–[12]. In the theory literature, the hitting, commute and cover times of random walk have been studied by [13]–[20]. A major approach is based on *spectral graph theory* [13], [19]–[21], in addition to effective resistance methods [16]–[18].

We particularly study a general setting of random walk with weighted edges and *heterogeneous* transition times at relaying nodes. For instance, these can correspond to the heterogeneous duty cycling rates in sensor networks, or the heterogeneous delivery times in store-carry-forward networks. We generalize the formulas in [13], [19] to compute the *exact* hitting and commute time in the presence of heterogeneous transition times.

Furthermore, we study random walks in a common class of distance-regular networks with varying numbers of nearest neighbors, which captures the notion of geographical proximity in sensor networks and delay tolerant networks. For instance, in an r-nearest neighbor cycle (or torus), there is an edge between every pair of neighbors that are r-hops away on a cycle (or torus). The varying numbers of nearest neighbors (by r's) can capture the levels of overhead for maintaining local neighbors (e.g., the transmission power in sensor networks, or the reachable regions of carriers in delay tolerant networks).

[1] In our experience, one has to run up to thousands of simulation trials to get a good estimate of the mean latency even in small networks.

[2] Note that the *a priori* shared knowledge of the neighbors' pseudo-random number generators (seed and cycle position) enables a transmitter to wake up in the precise time slot, in which a chosen receiver is supposed to wake up [7], thus saving precious energy at the transmitter node.

One of our aims is to enable the optimization of the latency-overhead trade-off in these distance-regular networks.

We note that there has been an asymptotic study in [22] about the order of magnitude of the commute and cover time for the r-nearest neighbor torus (called k-fuzz of torus) based on effective resistance techniques. However, this paper provides the exact formulas based on spectral graph theory to compute the hitting and commute time in r-nearest neighbor torus. Our results concern the exact analysis of the latency of stateless opportunistic forwarding in finite general graphs, and these are unsurprisingly more precise than asymptotic analysis. Our formulas are useful for solving optimization problems, such as latency-overhead optimization.

There are many other studies of hitting time in 1-nearest neighbor torus that consider only homogeneous transition times in various applications [23]–[27]. In contrast, our work presents the more general results of hitting time considering heterogeneous transition and r-nearest neighbor torus. Furthermore, we also extend our study to various more sophisticated settings, such as multiple-copy and topology-aware forwarding strategies, and networks resulting from random geographical locations.

In summary, our contributions are outlined as follows:

1) In Sec. III, we use spectral graph theory to derive a general formula to efficiently compute the exact hitting and commute time of weighted random walks with heterogeneous transition times at relaying nodes.
2) In Sec. IV, we study a common class of distance-regular networks with varying numbers of geographical neighbors, and obtain exact and approximation formulas of hitting time in such networks.
3) In Sec. V, we apply the results of distance-regular networks to estimate the latency of random walk considering random geographical locations of nodes.
4) In Sec. VI, we study other sophisticated settings, such as topology-aware forwarding and multi-copy strategy to obtain new insights on latency performance.

Due to paucity of space, additional discussion and proofs are deferred to the full technical report [28].

II. Problem Formulation

In this section, we formulate a model to capture a general type of stateless opportunistic forwarding by random walk in finite graphs. Also, we formulate a common class of distance-regular networks with varying numbers of nearest neighbors to model the geographical proximity in opportunistic forwarding.

A. Random Walk in a Finite Graph

To model stateless opportunistic forwarding, we consider a finite *connected* undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes (e.g., relaying devices or deposit sites) and $\mathcal{E}$ is the set of edges. Each edge is interpreted as a possible one-hop forwarding action between a pair of nodes (e.g., by radio transmission, packet carriers, or table lookup in a list of known neighbors). Let $n \triangleq |\mathcal{V}|$, $m \triangleq |\mathcal{E}|$. We also let $\mathcal{N}_u \subseteq \mathcal{V}$ be the set of neighbors of u, and its degree be $d_u \triangleq |\mathcal{N}_u|$.

We consider the setting of slotted time. In random walk based forwarding, a packet is stored at a node for certain time slots before there is an opportunity to be forwarded to its next hop neighbor. We assume that the forwarding operation in the current hop is stateless, which is independent of the forwarding operations of the previous hops.

For each neighboring pair of nodes $u, v \in \mathcal{V}$, we let $\ell_{u,v}$ be the expected transition time of a packet that travels from u to v, and $\rho_{u,v}$ be the forwarding probability that u will forward packets to $v \in \mathcal{N}_u$.

Then, the hitting time $\mathbf{H}_{u,v}$ from source u to reach destination v can be recursively expressed as:

$$\mathbf{H}_{u,v} = \begin{cases} \sum\limits_{w \in \mathcal{N}_u} \rho_{u,w}(\ell_{u,w} + \mathbf{H}_{w,v}) = \ell_u + \sum\limits_{w \in \mathcal{N}_u} \rho_{u,w} \mathbf{H}_{w,v} & \text{if } u \neq v \\ 0 & \text{if } u = v \end{cases} \tag{1}$$

where $\ell_u \triangleq \sum_{w \in \mathcal{N}_u} \rho_{u,w} \cdot \ell_{u,w}$ is the expected transition time at node u. Note that Eqn. (1) also holds for random transition times. In such case, $\ell_{u,v}$ represents the mean transition time.

Also, it is possible to consider self-loops at each node. In such a case, $\ell_u \triangleq \sum_{w \in \mathcal{N}_u} \rho_{u,w} \cdot \ell_{u,w} + \rho_{u,u} \cdot \ell_{u,u}$, where $\ell_{u,v}$ is the sojourn time at u, and $\rho_{u,u}$ is the probability of the existence of a self-loop.

We consider a simple setting of forwarding probability $\rho_{u,v}$, where we assign a weight $w_{u,v}$ to each edge (u, v) to indicate the availability of an opportunistic forwarding operation between the pair of nodes. Then $\rho_{u,v}$ is defined as:

$$\rho_{u,v} \triangleq \frac{w_{u,v}}{w_u}, \qquad \text{where} \quad w_u \triangleq \sum_{v' \in \mathcal{N}_u} w_{u,v'} \tag{2}$$

See Fig. 1 for an illustration. Note that in this paper, we assume the *symmetric* case: $w_{u,v} = w_{v,u}$. Therefore, this defines a random walk in a weighted graph, or equivalently, a reversible Markov chain [14]. Next, we give some concrete examples of our model as follows.

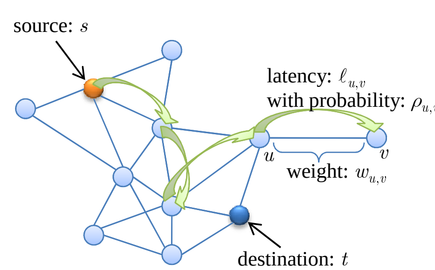


Fig. 1. An illustration of a weighted random walk in a finite graph with transition times given by $\{\ell_{u,v}\}$.

Example 1 (*I.I.D. Random Duty Cycling Sensor Networks*): Let ρ_{dc} be the duty cycling rate of all nodes, such that in one time slot, each node is awake with i.i.d.[3] probability ρ_{dc}, and is dormant with probability 1-ρ_{dc}. Hence, every neighbor has an equal chance of receiving a packet by opportunistic forwarding (namely, $w_{u,v} = 1$). Then the waiting time for both u and v to be awake in the same time slot is a

[3]We also address the setting of heterogeneous random duty cycles in the full technical report [28].

geometric random variable with parameter ρ_{dc}^2. As shown in Lemma 7 in the Appendix, the expected transition time at u (the waiting time that a packet is forwarded to the earliest wake-up neighbor) is $\ell_u = \frac{1}{1-(1-\rho_{\mathsf{dc}}^2)^{d_u}}$

Example 2 (*I.I.D. Pseudo-Random Duty Cycling Sensor Networks*): As proposed in [7], the neighboring nodes can first exchange the seed of pseudo-random sequence that generates the random duty cycling. Then a node can predict the exact awake time slots of its neighbors. Let ρ_{dc} be the duty cycling rate. Similarly, $w_{u,v} = 1$. Then the waiting time for both u and v to be awake in the same time slot is a geometric random variable with parameter ρ_{dc}. As shown in Lemma 7, the expected transition time at u is $\ell_u = \frac{1}{1-(1-\rho_{\mathsf{dc}})^{d_u}}$

Example 3 (*Delay Tolerant Networks with Geographical Dependence*): We suppose that each deposit site has its packet carriers randomly commuting among its neighboring sites within a certain bounded reachable region. And we assume a simple setting where the transition time depends linearly on the geographical distance, such that $\ell_{u,v} = ||u - v||$ and $w_{u,v} = 1$. That is, the further away the nodes are, the longer transition time is incurred. Although other mobility models (e.g., random waypoint model) can also be considered, the expressions for $w_{u,v}$ and $\ell_{u,v}$ will be more complicated. We may also set the weight $w_{u,v}$ to depend on the geographical distance $\ell_{u,v}$, such as $w_{u,v} = \ell_{u,v}^{-\alpha}$ or $w_{u,v} = e^{-\beta \ell_{u,v}}$, as to model the random encounters between the packet carriers and deposit sites.

Example 4 (*Searching in Peer-to-peer Networks*): Suppose that the frequency of peer node u to meet node v online simultaneously is captured by weight $w_{u,v}$. Then $\rho_{u,v} = \frac{w_{u,v}}{w_u}$ naturally characterizes the likelihood that u will forward a query message to v. And $\ell_{u,v}$ represents the average waiting time that u waits for v to be online simultaneously.

These examples can be captured by weighted random walk with heterogeneous transition times. In this paper, we suppose that the topology $\mathcal{G}$, weights $\{w_{u,v}\}$ and latency $\{\ell_{u,v}\}$ are a priori given or measured empirically. The natural question is to predict the hitting time of stateless opportunistic forwarding in such a formulation.

B. Nearest Neighbor Networks

We also particularly consider a common class of distance-regular networks that can capture the notion of geographical proximity. For simplicity, we consider a boundary-less space. Suppose that there is a set of n nodes placed evenly in a one dimension as a cycle, or in a two dimensional space as a torus.

We define an r-nearest neighbor cycle as the graph with edges between every pair of nodes within r-hops away in the cycle. This naturally captures the notion geographical proximity, such that neighbors lie within the bounded transmission range in sensor networks, or are reachable by a single carrier wandering in a bounded region in delay tolerant networks. See Fig. 2 (b) for an example of 2-nearest neighbor cycle.

To generalize to the two dimensional case as a torus, we note that there are various ways of defining the nearest neighbors in two dimensions. One may use the L^1 norm such that there is an edge between the nodes whose shortest path is within r hops in the torus (see Fig. 3 (a)), or L^∞ norm such that the vertical and horizontal distance are both within r hops in the torus (see Fig. 3 (b)). Generally, we can consider other norms (e.g., L^2 norm to model circular transmission range), which however are less convenient to study.

III. Hitting and Commute Time of Random Walk

To solve the hitting and commute time of random walk in arbitrary graphs, we present two general techniques: 1) effective resistance and 2) spectral graph theory. Our study will be based on spectral graph theory to derive efficient general formulas for the hitting and commute time of random walk with heterogeneous transition times at relaying nodes.

A. Effective Resistance

We first survey the approach of effective resistance [16]–[18]. Given a finite graph $\mathcal{G}$, we assign each edge $(u, v) \in \mathcal{E}$ a resistance of value $1/w_{u,v}$. For any pair of nodes $u, v \in \mathcal{V}$ (which may not necessarily be neighbors), we define the effective resistance $\mathfrak{R}_{u,v}$ as the voltage difference between u and v, when a unit current is injected at u and removed from v.

Denote the commute time between node u and node v as $\mathbf{C}_{u,v}$. Thus, $\mathbf{C}_{u,v} = \mathbf{H}_{u,v} + \mathbf{H}_{v,u}$ (i.e., the sum of hitting time from u to v and vice versa).

In [18], it is shown that

$$\mathbf{C}_{u,v} = \mathfrak{R}_{u,v} \cdot \sum_{(u,v)\in\mathcal{E}} w_{u,v}(\ell_{u,v} + \ell_{v,u}) \tag{3}$$

In the case of homogeneous transition times (i.e., $\ell_{u,v} = 1$ for all $u, v \in \mathcal{V}$), [17] has shown that

$$\mathbf{H}_{u,v} = \frac{1}{2}\Big(\mathbf{C}_{u,v} + \sum_{w\in\mathcal{V}} \pi_w(\mathbf{C}_{w,v} - \mathbf{C}_{w,u})\Big) \tag{4}$$

where we define a vector $\pi = (\pi_u)_{u\in\mathcal{V}}$ such that

$$\pi_u \triangleq \frac{w_u}{W} \qquad \text{where} \quad W \triangleq \sum_{u'\in\mathcal{V}} w_{u'} \tag{5}$$

By standard Markov chain theorem [14], π is the unique stationary distribution of random walk in $\mathcal{G}$.

However, it is not clear how to obtain the hitting time from effective resistance, with arbitrary $\ell_{u,v}$. In this paper, we rely on an alternate approach based on spectral graph theory.

B. Spectral Graph Theory

First, we define some notations. Denote the adjacency matrix of $\mathcal{G}$ as $\mathbf{A}$ such that

$$\mathbf{A}_{u,v} \triangleq \begin{cases} w_{u,v} & \text{if } (u,v) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

We also denote the adjacency matrix of a given graph $\mathcal{G}$ as $\mathbf{A}[\mathcal{G}]$. Define the diagonal matrix of $\mathcal{G}$ as $\mathbf{D}$ such that

$$\mathbf{D}_{u,v} \triangleq \begin{cases} w_u & \text{if } u = v \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

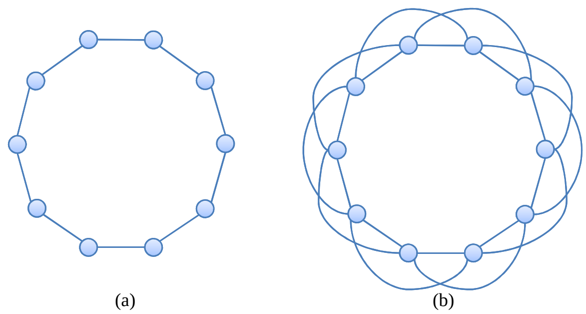


Fig. 2. (a) A cycle. (b) A 2-nearest neighbor cycle.

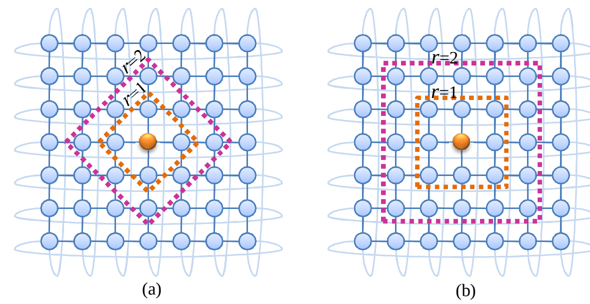


Fig. 3. (a) L^1 Nearest neighbor tori. (b) L^∞ Nearest neighbor tori. The dotted boxes indicate the neighborhoods of the centre node.

And define the normal matrix as $\mathbf{N} \triangleq \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$. That is,

$$\mathbf{N}_{u,v} \triangleq \begin{cases} \frac{w_{u,v}}{\sqrt{w_u w_v}} & \text{if } (u,v) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Note that $\mathbf{N}$ is symmetric, and hence there exist real eigenvalues and eigenvectors of $\mathbf{N}$. Let λ_k and $\boldsymbol{\nu}_k$ be the $(k+1)$-th eigenvalue[4] and the corresponding eigenvector of $\mathbf{N}$.

B.1. Homogeneous Transition Times

Consider homogeneous transition times: $\ell_u = 1$ for all u. In [13] (Theorem 3.1), Lovasz solved Eqn. (1) with a solution:

$$\mathbf{H}_{u,v} = \sum_{k:\lambda_k \neq 1} \frac{W}{1-\lambda_k}\Big(\frac{\boldsymbol{\nu}_{k,v}^2}{w_v} - \frac{\boldsymbol{\nu}_{k,u}\boldsymbol{\nu}_{k,v}}{\sqrt{w_u w_v}}\Big) \tag{9}$$

where $\boldsymbol{\nu}_{k,u}$ is the u-th entry of vector $\boldsymbol{\nu}_k$.

By $\mathbf{C}_{u,v} = \mathbf{H}_{u,v} + \mathbf{H}_{v,u}$,

$$\mathbf{C}_{u,v} = \sum_{k:\lambda_k \neq 1} \frac{W}{1-\lambda_k}\Big(\frac{\boldsymbol{\nu}_{k,u}}{\sqrt{w_u}} - \frac{\boldsymbol{\nu}_{k,v}}{\sqrt{w_v}}\Big)^2 \tag{10}$$

Although Lovasz considered unweighted graphs where $w_u = d_u$ (the degree of u) and $W = 2m$ (twice as the number of edges), it can be easily generalized to weighted graphs.

In [19] (Theorem 8), Chung and Yau considered the (normalized) Laplacian of a graph defined as $\mathbf{L} \triangleq \mathbf{I} - \mathbf{N}$, and independently proved Eqn. (9) via discrete Green's function[5].

The complexity of Eqn. (9) or (10) is $O(n)$, and solving the eigen spectrum of a symmetric matrix is $O(n^3)$. Since the eigen spectrum of $\mathbf{N}$ can be reused for all source-destination pairs. Hence, the complexity of computing the hitting and commute time for all n^2 source-destination pairs is $O(n^3)$.

B.2. Heterogeneous Transition Times

In this paper, we consider a general setting with heterogeneous transition times $\{\ell_u\}$ at different nodes.

We are interested in obtaining similar formulas as Eqns. (9)-(10) for computing the hitting time, whose time complexity takes only $O(n^3)$. A major contribution of this paper is Theorem 1, which extends the formulas in [13], [19] to the setting of heterogeneous transition times. Furthermore, we will study the eigenvalues and eigenvectors for specific network topologies in the next section.

[4]The order of eigenvalues does not matter.

[5]The proof given by Lovasz in [13] is rather sketchy. Hence, our results are based on the one in [19].

We define a latency matrix $\mathbf{S}$ as:

$$\mathbf{S}_{u,v} \triangleq \begin{cases} \ell_u & \text{if } u = v \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

and define "generalized" Laplacian $\tilde{\mathbf{L}} \triangleq \mathbf{S}^{-\frac{1}{2}}(\mathbf{I}-\mathbf{N})\mathbf{S}^{-\frac{1}{2}}$ as:

$$\tilde{\mathbf{L}}_{u,v} \triangleq \begin{cases} \frac{1}{\ell_u} & \text{if } u = v \\ -\frac{w_{u,v}}{\sqrt{w_u \ell_u w_v \ell_v}} & \text{if } (u,v) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Note that $\tilde{\mathbf{L}}$ is also symmetric, hence there exist real eigenvalues and eigenvectors of $\tilde{\mathbf{L}}$. Let σ_k and $\boldsymbol{\mu}_k$ be the $(k+1)$-th eigenvalue and the corresponding eigenvector of $\tilde{\mathbf{L}}$.

Theorem 1: Given arbitrary transition time $\ell_u > 0$ for each u, the hitting time and commute time from u to v can be computed by:

$$\mathbf{H}_{u,v} = \sum_{k:\sigma_k \neq 0} \frac{\tilde{W}}{\sigma_k}\Big(\frac{\boldsymbol{\mu}_{k,v}^2}{w_v \ell_v} - \frac{\boldsymbol{\mu}_{k,u}\boldsymbol{\mu}_{k,v}}{\sqrt{w_u \ell_u w_v \ell_v}}\Big) \tag{13}$$

$$\mathbf{C}_{u,v} = \sum_{k:\sigma_k \neq 0} \frac{\tilde{W}}{\sigma_k}\Big(\frac{\boldsymbol{\mu}_{k,u}}{\sqrt{w_u \ell_u}} - \frac{\boldsymbol{\mu}_{k,v}}{\sqrt{w_v \ell_v}}\Big)^2 \tag{14}$$

where $\boldsymbol{\mu}_{k,u}$ is the u-th entry of vector $\boldsymbol{\mu}_k$, and $\tilde{W} \triangleq \sum_{(u,v)\in\mathcal{E}} w_{u,v}(\ell_{u,v} + \ell_{v,u}) = \sum_{u'\in\mathcal{V}} w_{u'}\ell_{u'}$.

Proof: The idea is to generalize the one in [19] to a setting with heterogeneous transition times. See the Appendix. ∎

When $\ell_{u,v} = 1$ for all $(u,v) \in \mathcal{E}$, it is easy to see that $\sigma_k = 1 - \lambda_k$ and $\boldsymbol{\mu}_k = \boldsymbol{\nu}_k$. Hence, Eqns. (13)-(14) reduce to Eqns. (9)-(10). The time complexity of computing the hitting and commute time with heterogeneous transition times by Eqn. (9) for all n^2 source-destination pairs is also $O(n^3)$.

We present a numerical example of *Example* 3 in Fig. 4 with hitting time computed by Eqn. (13). We have verified that the hitting times in Fig. 4 indeed satisfy Eqn. (1).

Finally, we remark that [29] recently generalizes some of our results of the hitting time for irreversible Markov chains, using the notion of pseudo-inverse of Laplacian.

IV. r-Nearest Neighbor Networks

In this section, we especially study the Laplacian of the nearest neighbor networks and obtain specific formulas for computing the hitting and commute time. For clarity, in this

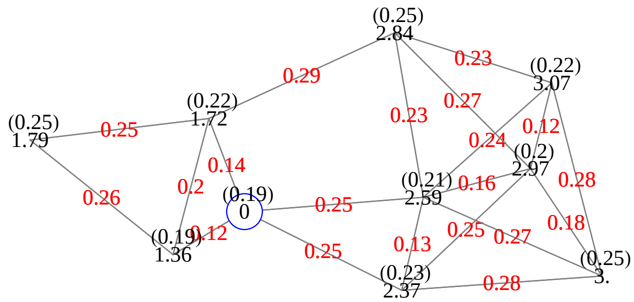


Fig. 4. A randomly generated instance of *Example* 3, where the radius of one-hop reachable region is 0.3. On each node is the hitting time to hit the blue circled node, where the red number on each edge is the Euclidean distance of its end nodes, and the number in bracket is the expected transition time at each node.

section we consider the homogeneous case: $\ell_{u,v} = 1$ for all $(u, v) \in \mathcal{E}$.

A. r-Nearest Neighbor Cycles

Denote a cycle of n nodes as $\mathcal{C}_n$. We construct an r-nearest neighbor cycle (denoted as $\mathcal{C}_n^r$) as the graph with edges between nodes and their r-nearest left and r-nearest right neighbors on $\mathcal{C}_n$. We label the nodes by the order around the cycle as: $0, 1, ..., n-1$. We consider the uniform symmetric case, such that $w_{u,u+j} = w_{u,n-j+u} = a_j$ for $1 \leq j \leq r$.

Lemma 1: The $(k+1)$-th eigenvalue and eigenvector of the Laplacian $\tilde{\mathbf{L}}$ of r-nearest neighbor cycle $\mathcal{C}_n^r$ are:

$$\boldsymbol{\sigma}_k = 1 - \frac{\sum_{j=1}^{r} a_j \cos\left(\frac{2\pi jk}{n}\right)}{\sum_{j'=1}^{r} a_{j'}}, \ \boldsymbol{\mu}_k = \frac{1}{\sqrt{n}}\left(1, \epsilon^k, ..., \epsilon^{(n-1)k}\right)^T$$

where ϵ is a complex number defined as: $\epsilon \triangleq \cos\left(\frac{2\pi}{n}\right) + \mathbf{i}\sin\left(\frac{2\pi}{n}\right)$.

Proof: We show that $1 - \sigma_k$ and $\boldsymbol{\mu}_k$ are the eigenvalue and eigenvector of normal matrix $\mathbf{N}$ of $\mathcal{C}_n^r$. See the Appendix. ■

Lemma 1 is an extension to [30] Lemma 11.

Theorem 2: Suppose $a_j = 1$ for $1 \leq j \leq r$. Without loss of generality, we consider the hitting time from u to 0 in r-nearest neighbor cycle $\mathcal{C}_n^r$. Then, it can be computed by:

$$\mathbf{H}_{u,0} = 2r \sum_{k=1}^{n-1} \frac{1 - \cos\left(\frac{2\pi ku}{n}\right)}{(2r+1) - \frac{\sin\left(\frac{\pi k(2r+1)}{n}\right)}{\sin\left(\frac{\pi k}{n}\right)}} \tag{15}$$

Proof: Note that the degree $d_u = 2r$ is a constant for all u. Thus, $W = 2rn$. By Lemma 1 and Eqn. (9), we obtain:

$$\mathbf{H}_{u,0} = 2rn \sum_{k=1}^{n-1} \frac{1 - \cos\left(\frac{2\pi ku}{n}\right) - \mathbf{i}\sin\left(\frac{2\pi ku}{n}\right)}{2rn\left(1 - \frac{1}{r}\sum_{j=1}^{r}\cos\left(\frac{2\pi jk}{n}\right)\right)} \tag{16}$$

$$= \sum_{k=1}^{n-1} \frac{1 - \cos\left(\frac{2\pi ku}{n}\right)}{1 - \frac{1}{r}\sum_{j=1}^{r}\cos\left(\frac{2\pi jk}{n}\right)} \tag{17}$$

This is due to the identities: $\cos\left(\frac{2\pi jk}{n}\right) = \cos\left(\frac{2\pi j(n-k)}{n}\right)$ and $\sin\left(\frac{2\pi ku}{n}\right) = -\sin\left(\frac{2\pi(n-k)u}{n}\right)$. Finally, Eqn. (15) follows from the trigonometric identity of Dirichlet kernel from Lemma 5 in Appendix. ■

Theorem 2 generalizes our previous results in [21] for only the *maximum* hitting time in r-nearest neighbor cycle.

Theorem 3: When n is even, the maximum hitting time in an r-nearest neighbor cycle can be approximated by:

$$\mathbf{H}_{\frac{n}{2},0} \approx \frac{3n^2}{2(1+r)(1+2r)} \tag{18}$$

Proof: It is based on the careful approximation of the Taylor series expansion of Eqn. (15). See the Appendix. ■

When $r = 1$, it is well known that via effective resistance [14], [15] the hitting time for a pair of farthest nodes in an n-node cycle is indeed $\frac{n^2}{4}$. Hence, Theorem 3 is accurate for 1-nearest neighbor cycles. For r-nearest neighbor cycles, Fig. 5 numerically shows that Eqn. (18) gives a relatively accurate approximation to the exact computation based on Eqn. (15).

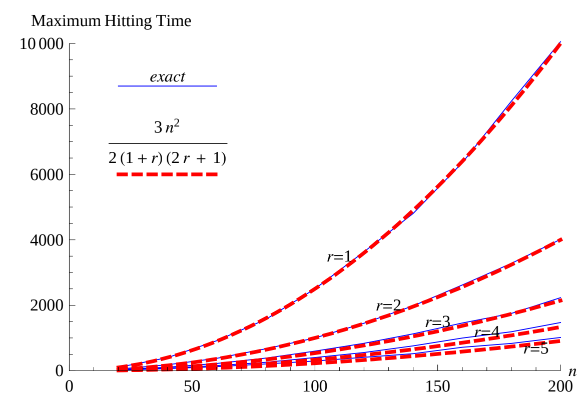


Fig. 5. $\mathbf{H}_{\frac{n}{2},0}$ computed exactly by Eqn. (15) is plotted against the approximation using Eqn. (18).

B. r-Nearest Neighbor Tori

For convenience of presentation, we label the nodes on a torus by $(0, 0)$ to (n, n). Namely, there are n^2 nodes, whereas there are n nodes in the cycle case.

Theorem 4: The hitting time $\mathbf{H}_{(u,v),(0,0)}$ from node (u, v) to node $(0, 0)$ can be computed by:

1) (*Torus*):

$$\sum_{(k,l)\neq(0,0)} \frac{1 - \cos\left(\frac{2\pi(ku+lv)}{n}\right)}{1 - \frac{1}{2}\left(\cos\left(\frac{2\pi k}{n}\right) + \cos\left(\frac{2\pi l}{n}\right)\right)} \tag{19}$$

2) (L[1] *r-Nearest Neighbor Torus*):

$$\sum_{(k,l)\neq(0,0)} \frac{1 - \cos\left(\frac{2\pi(ku+lv)}{n}\right)}{1 - \frac{1}{2r^2+2r}\left(\sum_{i=-r}^{r}\sum_{j=-r+|i|}^{r-|i|}\cos\left(\frac{2\pi ik}{n}\right)\cos\left(\frac{2\pi jl}{n}\right) - 1\right)} \tag{20}$$

3) (L^∞ *r-Nearest Neighbor Torus*):

$$\sum_{(k,l)\neq(0,0)} \frac{1-\cos\left(\frac{2\pi(ku+lv)}{n}\right)}{1-\frac{1}{4r^2+4r}\left(\sum_{i=-r}^{r}\sum_{j=-r}^{r}\cos\left(\frac{2\pi ik}{n}\right)\cos\left(\frac{2\pi jl}{n}\right)-1\right)} \tag{21}$$

(*Proof idea*): We first give the basic idea as follows. The proof relies on the notion of two types of composed graphs. Recall that the adjacency matrix of graph $\mathcal{G}$ is $\mathbf{A}[\mathcal{G}]$. We define two types of composed graphs as follows:

1) *Union Graph*: Given $\mathcal{G}=(\mathcal{V},\mathcal{E})$ and $\mathcal{H}=(\mathcal{V},\mathcal{F})$, define a graph, $\mathcal{G}\cup\mathcal{H}$, such that the set of nodes is $\mathcal{V}$ and its adjacency matrix is just the sum of those of $\mathcal{G}$ and $\mathcal{H}$:

$$\mathbf{A}[\mathcal{G}\cup\mathcal{H}] = \mathbf{A}[\mathcal{G}] + \mathbf{A}[\mathcal{H}] \tag{22}$$

Namely, the edge weights are the sum of those of $\mathcal{G},\mathcal{H}$.

2) *Tensor Product Graph*: Given $\mathcal{G}=(\mathcal{V},\mathcal{E})$ and $\mathcal{G}'=(\mathcal{V}',\mathcal{E}')$, define a graph, $\mathcal{G}\times\mathcal{G}'$, such that the set of nodes is $\mathcal{V}\times\mathcal{V}'$, and its adjacency matrix is the tensor product[6] of those of $\mathcal{G},\mathcal{G}'$:

$$\mathbf{A}[\mathcal{G}\times\mathcal{G}'] = \mathbf{A}[\mathcal{G}] \otimes \mathbf{A}[\mathcal{G}'] \tag{23}$$

See the Appendix for the formal definition of tensor product of matrices. Namely, (u,u') and (v,v') are adjacent in $\mathcal{G}\times\mathcal{G}'$, if $(u,v)\in\mathcal{E}$ and $(u',v')\in\mathcal{E}'$. Particularly, if $n=|\mathcal{V}|$ and $\mathbf{I}$ is the $n\times n$ identity matrix, then $\mathbf{A}[\mathcal{G}]\otimes\mathbf{I}$ defines n disjoint copies of $\mathcal{G}$.

The combinations of union graphs and tensor product graphs over nearest neighbor cycles can generate a large class of nearest neighbor tori with arbitrary norms. We next give some examples of such constructions in Fig. 6 as follows.

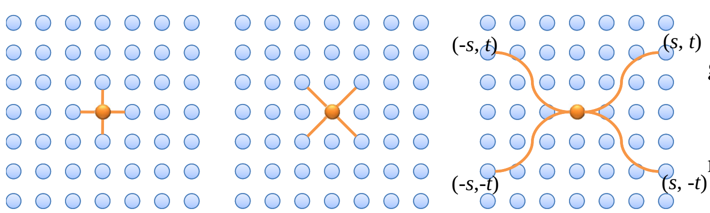


Fig. 6. Three different settings of neighbors of the center node, as explained in (n1)-(n3)

We consider n^2 nodes placed evenly on the two dimensional boundary-free surface of a torus. We assume that the setting of neighbors is uniform to all nodes. We label the nodes by the coordinates: (u,v), where $0\le u,v\le n-1$ on the surface. We explain the examples in Fig. 6 as follows:

- n1 In Fig. 6 (a), each node has four neighbors (two horizontal and two vertical). This indeed forms a torus, whose adjacency matrix is: $\mathbf{A}[\mathcal{C}_n]\otimes\mathbf{I}+\mathbf{I}\otimes\mathbf{A}[\mathcal{C}_n]$
- n2 In Fig. 6 (b), each node has four neighbors (all diagonal). If n is odd, then this forms a connected graph. This defines a graph, with adjacency matrix as: $\mathbf{A}[\mathcal{C}_n]\otimes\mathbf{A}[\mathcal{C}_n]$
- n3 Let $\mathcal{C}_n^r(s)$ be an r-nearest neighbor cycle such that $a_s=1$ and $a_j=0$ for $j\neq s$. In Fig. 6 (c), each node (u,v) has only four neighbors as $(u\pm s \mod n, v\pm t \mod n)$. This defines a graph, with adjacency matrix as: $\mathbf{A}[\mathcal{C}_n^r(s)]\otimes\mathbf{A}[\mathcal{C}_n^r(t)]$

[6]The definition of tensor product of matrices can be found in Appendix.

In fact, a nearest neighbor torus defined by arbitrary norm can be regarded as a union graph of a collection of graphs with suitable $\mathbf{A}[\mathcal{C}_n^r(s)]\otimes\mathbf{A}[\mathcal{C}_n^r(t)]$.

Note that, for $0\le k,l\le n-1$, we define a vector $\boldsymbol{\mu}_{(k,l)}$:

$$\boldsymbol{\mu}_{(k,l)} \triangleq \frac{1}{n}\left(1,\epsilon^k,...,\epsilon^{(n-1)k}\right)^T \otimes \left(1,\epsilon^l,...,\epsilon^{(n-1)l}\right)^T \tag{24}$$

By Lemma 1, it follows that $\boldsymbol{\mu}_{(k,l)}$ is the $(kn+l+1)$-th eigenvector of the Laplacian of the example in Fig. 6 (c), whose eigenvalue is $1-\frac{1}{2}\left(\cos\left(\frac{2\pi sk}{n}\right)+\cos\left(\frac{2\pi tl}{n}\right)\right)$.

In general, it can be generalized as the following lemma.

Lemma 2: Given a set of tuples: $\{(s_1,t_1),...,(s_r,t_r)\}$ such that $(s_j,t_j)\neq(0,0)$ for $1\le j\le r$. Consider a graph $\mathcal{G}$ with n^2 nodes, such that each node (u,v) has only four neighbors as $(u\pm s_j \mod n, v\pm t_j \mod n)$ for $1\le j\le r$, whose edge weight is 1. Then the $(kn+l+1)$-th eigenvalue of the Laplacian $\tilde{\mathbf{L}}$ of $\mathcal{G}$ is:

$$\sigma_{(k,l)} = 1-\frac{1}{r}\sum_{j=1}^{r}\cos\left(\frac{2\pi s_j k}{n}\right)\cos\left(\frac{2\pi t_j k}{n}\right) \tag{25}$$

and the corresponding eigenvector is $\boldsymbol{\mu}_{(k,l)}$.

Proof: (Theorem 4) Since for a L^1 r-nearest neighbor torus the degree is $2r^2+2r$, and for a L^∞ r-nearest neighbor torus the degree is $4r^2+4r$, applying Lemma 2 to substitute into Eqn. (9), we complete the proof of Theorem 4. ■

Theorem 4 can be extended to even more sophisticated geographical proximity relations, other than L^1 and L^∞ norms.

Theorem 5: The maximum hitting time in an r-nearest neighbor torus is:

$$\mathbf{H}_{(\frac{n}{2},\frac{n}{2}),(0,0)} = \Theta\Big(\frac{n^2\log(n)}{(1+2r)^2}\Big) \tag{26}$$

The proof can be found in the full technical report [28]. We remark that [22] has proven a similar result of Theorem 5 for the commute time in k-fuzz of torus.

Furthermore, we numerically approximate the exact formulas in Theorem 4 (and corroborated in Figs. 7-8):

1) (L^1 *r-Nearest Neighbor Torus*):

$$\mathbf{H}_{(\frac{n}{2},\frac{n}{2}),(0,0)} \approx \frac{3.98n^2\log(n)}{(1+2r)^2}+0.96n^2 \tag{27}$$

2) (L^∞ *r-Nearest Neighbor Torus*):

$$\mathbf{H}_{(\frac{n}{2},\frac{n}{2}),(0,0)} \approx \frac{2.34n^2\log(n)}{(1+2r)^2}+0.96n^2 \tag{28}$$

These equations will be useful for the latency-overhead optimization, and the determination of time-to-live (TTL) value of packets in stateless opportunistic forwarding.

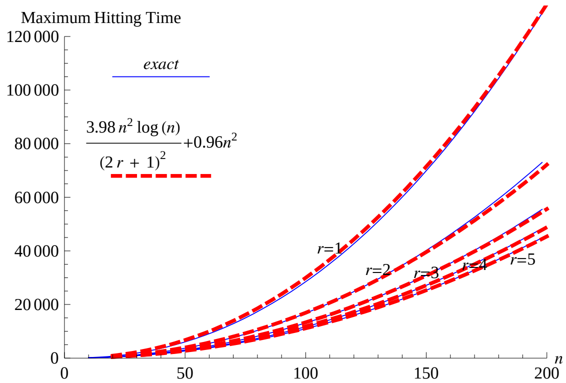


Fig. 7. For L^1 r-nearest neighbor torus, $\mathbf{H}_{(\frac{n}{2},\frac{n}{2}),(0,0)}$ computed exactly by Eqn. (27) is plotted against the approximation using Eqn. (20).

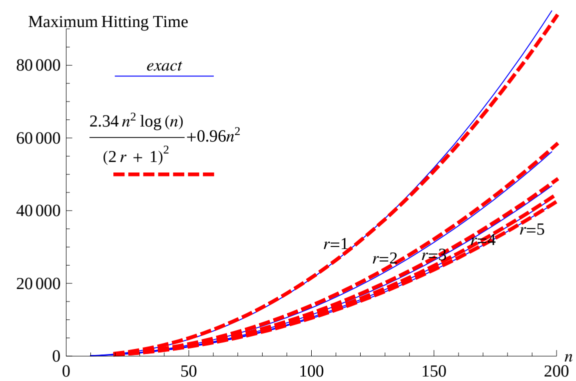


Fig. 8. For L^∞ r-nearest neighbor torus, $\mathbf{H}_{(\frac{n}{2},\frac{n}{2}),(0,0)}$ computed exactly by Eqn. (28) is plotted against the approximation using Eqn. (21).

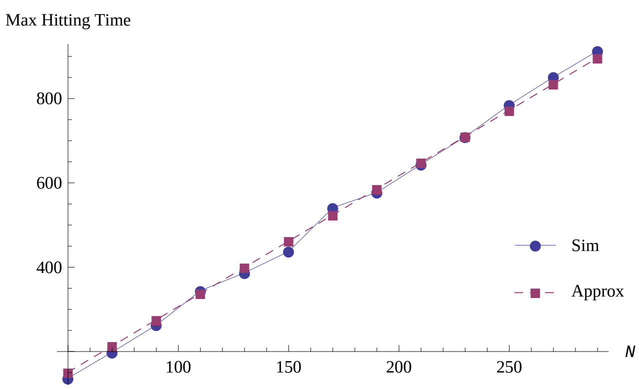


Fig. 9. *Example* 2 (sensor networks with i.i.d. pseudo-random duty cycling) over a random geometric graph $\mathcal{G}_{\mathsf{geo}}(N,\mathsf{R})$, where we set $\rho_{\mathsf{dc}} = 0.1$. We also compare to the approximation by Eqn. (30).

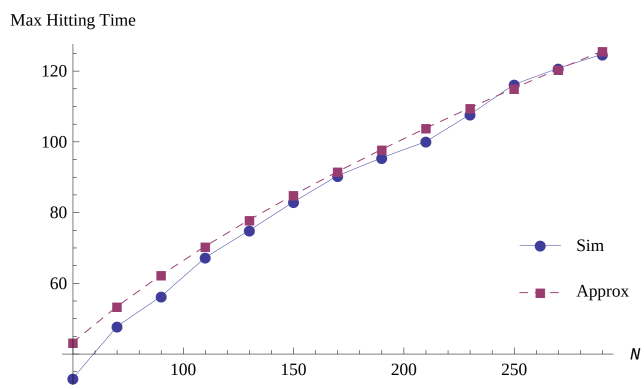


Fig. 10. *Example* 3 (store-carry-forward networks with geographical dependence) over a random geometric graph $\mathcal{G}_{\mathsf{geo}}(N,\mathsf{R})$, where we set $\ell_{u,v} = ||u-v||$ and $w_{u,v} = 1$. We also compare to the approximation by Eqn. (32).

V. Random Geographical Locations

In this section, we apply the results of Sec. IV to the settings of random geographical locations and distance-dependent weights and transition times, using random geometric graphs.

Random geometric graphs have been widely-used for modeling diverse wireless ad hoc networks, in which nodes are randomly placed in a confined area, and the communication links are established between nodes that are within a predefined transmission radius. Here we study the hitting time of random walk in random geometric graphs.

We denote a random geometric graph by $\mathcal{G}_{\mathsf{geo}}(N,\mathsf{R})$, which is an ensemble of N-node graphs, such that the position of each node is independently uniformly distributed on a 2D unit area, and there is an edge between a pair of nodes if they are within the transmission radius R.

Next, we quote a lemma from [22], which shows the distribution of degrees of a random geometric graph is concentrated on the mean degree.

Lemma 3: [22] (Lemma 3.2-3.4) Given a random geometric graph $\mathcal{G}_{\mathsf{geo}}(N,\mathsf{R})$, such that $\mathsf{R} = c\sqrt{\frac{\log N}{N}}$) for some constant c. Then the degree of every node u is:

$$d_u = \Theta(N\pi\mathsf{R}^2) \quad \text{w.h.p.}$$

It is easy to see that the degree distribution of a node follows the Binomial distribution, where $N\pi\mathsf{R}^2$ is the mean. Lemma 3 implies that the degree distribution of random geometric graphs is concentrated around the mean for large N and appropriate R. Hence, it seems reasonable to approximate random geometric graph by a nearest-neighbor network to estimate the hitting time. By simulation, we observe that the degree is close to $N\pi\mathsf{R}^2$ with a negligible variant, when $c \approx 2$.

Next, we consider the settings of *Example* 2 (i.i.d. pseudo-random duty cycling sensor networks) and *Example* 3 (delay tolerant networks with geographical dependence) from Sec. II on a random geometric graph. Figs. 9 and 10 show the numerical results of the maximum hitting time between the farthest pair of nodes for *Examples* 2 and 3 in $\mathcal{G}_{\mathsf{geo}}(N,\mathsf{R})$.

The numerical study was performed as follows. We first randomly generated a geometric graph in $\mathcal{G}_{\mathsf{geo}}(N,\mathsf{R})$, such that R scales as $c\sqrt{\log N/N}$ for $c \approx 2$ to ensure high probability of a connected geometric graph. Then, we applied Eqn. (13) to compute the maximum hitting time between the farthest pair of nodes, averaged over 1000 different graph instances, where $w_{u,v}$ and $\ell_{u,v}$ are set according to *Example* 2 and *Example* 3.

We observe that the maximum hitting time in *Example* 2 scales linearly in N. This observation can be explained by

Theorem 5 in Sec. IV-B. First, we let the number of hops as:

$$L(N, \mathsf{R}) \triangleq \frac{N \log N}{r^2} \tag{29}$$

where r^2 is the number of neighbors. By Lemma 3 we obtain $r^2 \sim \log N$ and $L(N, \mathsf{R}) \sim N$.

Then the maximum hitting time for *Example* 2 can be estimated by:

$$\mathbf{H}_{\max} \sim \frac{L(N, \mathsf{R})}{1 - (1 - \rho_{\mathsf{dc}})^{N \pi \mathsf{R}^2}} \tag{30}$$

which follows from the regularity suggested by Lemma 3.

We next consider *Example* 3. The maximum hitting time can be estimated by:

$$\mathbf{H}_{\max} \sim L(N, \mathsf{R}) \bar{\ell}(N, \mathsf{R}) \tag{31}$$

where $\bar{\ell}(N, \mathsf{R})$ is the expected per-hop latency at each node, which can be approximated by R. Thus, the maximum hitting time is:

$$\mathbf{H}_{\max} \sim \sqrt{N \log N} \tag{32}$$

We observe that both Eqns. (30) and (32) match reasonably well with the trend obtained from simulations, as shown in Figs. 9-10, even for moderate values of N (≤ 250). This gives useful scaling laws for stateless opportunistic forwarding on uniform random geometric networks.

VI. Advanced Stateless Opportunistic Forwarding

In this section, we consider the latency in various more sophisticated settings and extend the work in Sec. III. First, we study the usefulness of partial topology information by a hybrid approach, combining opportunistic forwarding and shortest path routing, and obtain useful insights on the performance. Second, we study the case of multi-copy strategy, such that multiple instances of stateless opportunistic forwarding are initiated. Third, we evaluate the cover time considering heterogeneous transition times.

A. Topology-Aware Forwarding

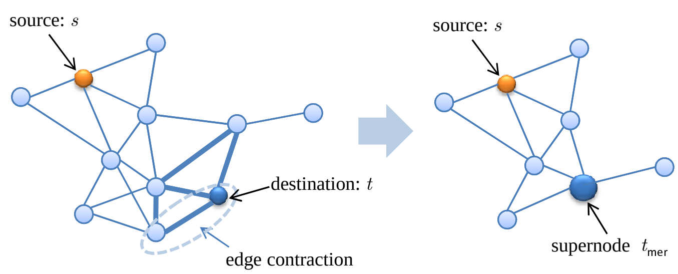
If the local topology information (or a decent estimate) is available, opportunistic forwarding can leverage on such information to improve performance. Particularly, we consider the following situations:

1) *Mobile Ad Hoc Networks*: Smart link state protocols, such as Hazy Sighted Link State routing (HSLS) [31], disseminate the link state information more frequent to nearby nodes and less frequently to farther nodes. Such biased dissemination can reduce the control overhead.
2) *Duty Cycling Sensor Networks*: For energy efficiency, a sink will only broadcast its reachability information (e.g., the full availability schedules of itself and its neighbors) to the nodes within a limited scope. Limited local broadcast can conserve more energy.

In these situations, we consider a hybrid approach of forwarding. First, stateless opportunistic forwarding is used until reaching one of the *topology-aware* nodes (e.g., the nodes have received information state in HSLS, or the full availability schedules in duty cycling sensor networks). Then, (deterministic) shortest path routing is used to minimize the expected latency from the topology-aware node to the destination. We call this *k-hop topology-aware opportunistic forwarding*, if only the neighbors within k hops away from the destination will carry out shortest path routing.

In this section, we especially study the latency of k-hop topology-aware opportunistic forwarding in the setting of *Example* 2 (i.i.d. pseudo-random duty cycling sensor networks) with duty cycling rate ρ_{dc}. In the setting of mobile ad hoc networks with HSLS, one can simply set $\rho_{\mathsf{dc}} = 1$.

Next, we apply the generalized Lovasz formula Eqn. (13) to compute the hitting time of k-hop topology-aware opportunistic forwarding. First, one can simply replace the set of topology-aware neighbors by a super node, and the stateless opportunistic forwarding hitting any topology-aware neighbors is equivalent to hitting the supernode (see Fig. 11).



(a) $\mathcal{G}$ (b) $\mathcal{G}_{\mathsf{mer}}(t,1)$

Fig. 11. An illustration of 1-hop neighbors replaced by a super node.

Formally, we define *edge contraction*. Given a graph $\mathcal{G}$, we write $\mathcal{G} \backslash e$ as the resultant graph of the edge contraction on edge e in $\mathcal{G}$, such that the edge $e = (u, v)$ is removed from $\mathcal{G}$ by merging the nodes u and v, and all other edges incident at either u or v become incident at the merged node. Then given a destination t in $\mathcal{G}$, we define $\mathcal{G}_{\mathsf{mer}}(t, k)$:

$$\mathcal{G}_{\mathsf{mer}}(t, k) \triangleq \mathcal{G} \backslash \{ e = (u, v) \mid v \in \mathcal{N}_t^k \text{ and } u \in \mathcal{N}_t^k \} \tag{33}$$

where $\mathcal{N}_t^k$ is the set of neighbors within k hops from t (including t). See an illustration in Fig. 11. We refer to the *supernode* as t_{mer} in $\mathcal{G}_{\mathsf{mer}}(t, k)$. Another important distinction between $\mathcal{G}$ and $\mathcal{G}_{\mathsf{mer}}(t, k)$ is that in the latter graph, the weights of the edges connecting t_{mer} to its neighbors may have non-unit edge weights even though $\mathcal{G}$ had unit edge costs. Specifically, the new weight will be set as:

$$w_{\mathcal{G}_{\mathsf{mer}}(t,k)}(u, t_{\mathsf{mer}}) \triangleq \sum_{v \in \mathcal{N}_t^k \backslash \mathcal{N}_t^{k-1}} w_{\mathcal{G}}(u, v) \tag{34}$$

Hence, the hitting to destination by k-hop topology-aware opportunistic forwarding becomes:

$$\mathbf{H}_{s,t}(\mathcal{G}) = \begin{cases} \mathbf{H}_{s, t_{\mathsf{mer}}}(\mathcal{G}_{\mathsf{mer}}(t, k)) + \frac{k}{\rho_{\mathsf{dc}}} & \text{if } s \notin \mathcal{N}_t^k \\ \frac{k'}{\rho_{\mathsf{dc}}} & \text{if } s \in \mathcal{N}_t^{k'} \text{ and } k' \leq k \end{cases} \tag{35}$$

It follows from the fact that the latency of shortest path from a k-hop neighbor to the destination is just $\frac{k}{\rho_{\mathsf{dc}}}$. In the following, we also make use of Eqn. (35) to numerically study the properties of latency for obtaining insights of performance.

Numerical Study 1: First, we investigate how the latency of topology-aware opportunistic forwarding varies as a function of k and ρ_{dc}. Fig. 12 shows the average maximum hitting time of topology-aware opportunistic forwarding in a random geometric graph of 100 nodes in a similar setting of Sec. V.

There are two key observations from this numerical study:

1) For each fixed value of k, decreasing ρ_{dc} has a certain effect to reduce the hitting time. But the gain is only marginal beyond a certain value of ρ_{dc}. We also note that for sparser topologies such as grid networks, the hitting time curve does not appear as rapidly flattening with increasing ρ_{dc}.
2) For each fixed value of ρ_{dc}, there is a significant reduction in latency as k is increased from 0 to 2.

The second observation indicates that even though with little topology knowledge, e.g., link state in the 2-hop neighborhood of every node, we can still achieve latency almost as low as shortest path routing. Although this is not generally true for arbitrary network topologies, our simulation with random geometric graphs suggests that this can be a vital heuristic for controlling the latency of topology-aware opportunistic forwarding.

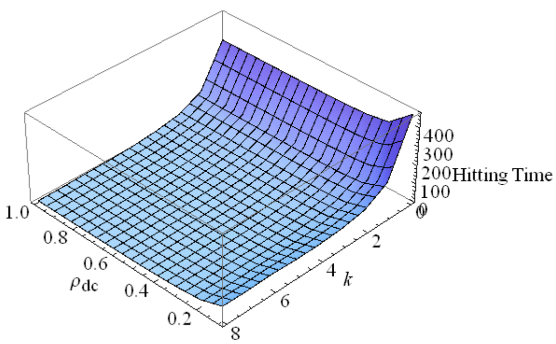


Fig. 12. The average maximum hitting of k-hop topology-aware opportunistic forwarding in a random geometric graph of 100 nodes as a function of the duty cycling rate ρ_{dc} and the scope k.

Numerical Study 2: Second, we study how the average maximum hitting time varies as the topology dissemination overhead in such random networks as a function of scope k. We focus on the $\rho_{\mathsf{dc}} = 1.0$ scenario since this applies to general routing and not just duty cycling. We define a simple measure of dissemination overhead as follows. Given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, for each node v, compute the subgraph $\mathcal{G}^k_v$ induced by the nodes that are within k hops from v. We define the *normalized mean overhead per node* as:

$$\mathrm{Overhead}(\mathcal{G}, k) \triangleq \sum_{v \in \mathcal{V}} \frac{|\mathcal{E}(\mathcal{G}^k_v)|}{|\mathcal{V}| \cdot |\mathcal{E}|} \tag{36}$$

where $\mathcal{E}(\mathcal{G}^k_v)$ is the set of edges in $\mathcal{G}^k_v$.

We then plot both metrics computed (and averaged) over 10 random instances of random geometric graphs with 100 nodes in Figs. 13 and 14. We observe that as the dissemination scope k goes up from 0 (i.e., pure stateless opportunistic forwarding) to 8 (i.e., pure shortest path routing), the latency between the farthest pair of nodes drops significantly and $\mathrm{Overhead}(\mathcal{G}, k)$ increases as k. A key observation from Fig. 14 is that both the latency and overhead curves have an inflection point at $k = 2$ (i.e., the latency drop becomes more gradual after that and the overhead increase becomes slightly sharper). This suggests that $k = 2$ is a good operational value. Ideally, if we can find a general closed form expression (even approximately) for Eqn. (35) for random geometric graphs at critical connectivity radius, then these inflection points could be found analytically, if they exist. This will be pursued in the future research.

B. Multi-Copy Opportunistic Forwarding

The previous sections concern only single-copy stateless opportunistic forwarding. We briefly show how to extend these results to the setting of multi-copy forwarding, by the construction of appropriate Cartesian product graphs.

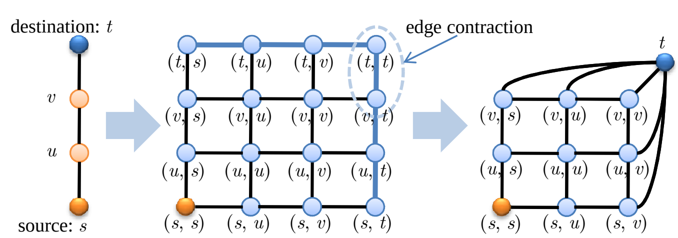
Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we define a *Cartesian product* graph as $\mathcal{G}\Box\mathcal{G}$, such that the set of nodes is $\mathcal{V} \times \mathcal{V}$, and (u, v) and (u', v') are adjacent in $\mathcal{G}\Box\mathcal{G}$, if and only if either one of the two cases is true:

Case (1): $u = u'$ and $(v, v') \in \mathcal{E}$.
Case (2): $v = v'$ and $(u, u') \in \mathcal{E}$.

We then denote $\mathcal{G}^{\Box \mathsf{k}}$ as the graph by taking Cartesian product of k copies of $\mathcal{G}$:

$$\mathcal{G}^{\Box \mathsf{k}} \triangleq \underbrace{\mathcal{G}\Box\mathcal{G}\Box\cdots\Box\mathcal{G}}_{\mathsf{k}\ \text{copies}} \tag{37}$$

It is easy to see that k copies of independent random walks in $\mathcal{G}$ is equivalent to a single random walk in $\mathcal{G}^{\Box \mathsf{k}}$. Suppose that $t \in \mathcal{V}$ is the destination. In k-copy stateless opportunistic forwarding, the hitting time to t is defined as the expected time that *at least one* of k packets hits t, when each packet follows an independent random walk. To incorporate this property into $\mathcal{G}^{\Box \mathsf{k}}$, we also make use of edge contraction, $\mathcal{G}\backslash e$ (see an illustration in Fig. 15).



(a) $\mathcal{G}$ (b) $\mathcal{G}^{\Box 2}$ (c) $\mathcal{G}^{\Box 2}_{\mathsf{mer}}(t)$

Fig. 15. An illustration of the Cartesian product of graphs $\mathcal{G}^{\Box 2}$, and edge contractions for all the edges with at least one coordinate as t.

Next, we apply edge contraction to all the edges in $\mathcal{G}^{\Box \mathsf{k}}$ that have t in one of the coordinates. Denote $\mathcal{E}(\mathcal{G}^{\Box \mathsf{k}})$ as the set of edges of $\mathcal{G}^{\Box \mathsf{k}}$. We define a new graph $\mathcal{G}^{\Box \mathsf{k}}_{\mathsf{mer}}(t)$ as:

$$\mathcal{G}^{\Box \mathsf{k}}_{\mathsf{mer}}(t) \triangleq \mathcal{G}^{\Box \mathsf{k}} \backslash \Big\{ (u, v) \in \mathcal{E}(\mathcal{G}^{\Box \mathsf{k}}) \mid v = (v_1, ..., v_{\mathsf{k}}), u = (u_1, ..., u_{\mathsf{k}}), \text{ and } u_{\mathsf{i}} = v_{\mathsf{j}} = t \text{ for some } \mathsf{i}., \mathsf{j} \in \{1, ..., \mathsf{k}\} \Big\}$$

That is, there is no difference between a pair of nodes that have t as one of the coordinates in $\mathcal{G}^{\Box \mathsf{k}}_{\mathsf{mer}}(t)$ (see Fig. 15). Hence, a

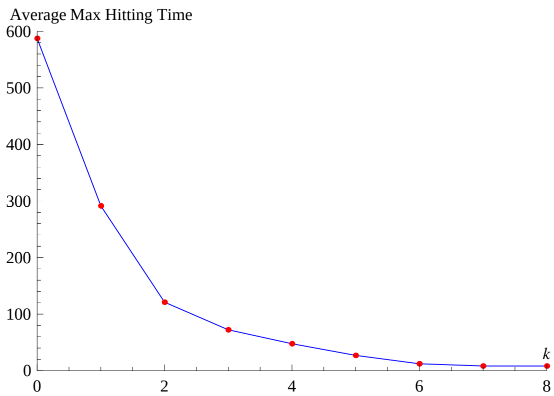


Fig. 13. Hitting time vs. overhead analysis for random geometric graph as a function of topology dissemination scope k for $\rho_{\mathsf{dc}} = 1.0$.

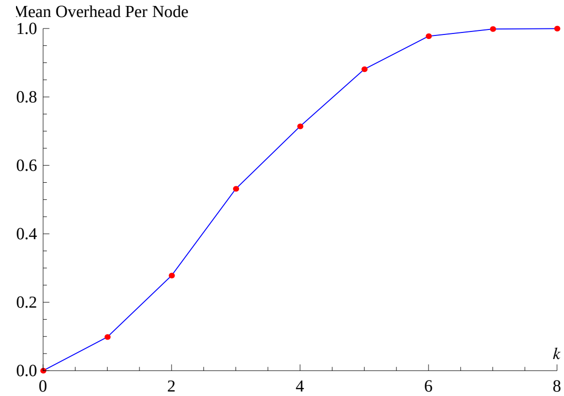


Fig. 14. Mean normalized overhead computed per node. The results are averaged over 10 instances of random geometric graphs with the same parameters.

packet hitting one of these nodes is equivalent to hitting the destination in k-copy opportunistic forwarding.

Without loss of generality, we also denote the merged node in $\mathcal{G}^{\square \mathsf{k}}_{\mathsf{mer}}(t)$ as t. Then, we can apply the Lovasz formula Eqn. (13) to compute the hitting time of k-copy opportunistic forwarding in $\mathcal{G}^{\square \mathsf{k}}_{\mathsf{mer}}(t)$, from node $(s, ..., s)$ to t.

We note that the order of magnitude of Lovasz formula depends on the number of nodes in a graph. In the Cartesian product graph of k graphs, the number of nodes is n^{k}. Hence, the complexity becomes $\mathrm{O}(n^{3\mathsf{k}})$ for a pair of source-destination. $\mathcal{G}^{\square \mathsf{k}}_{\mathsf{mer}}(t)$ does not appear to have simple expressions even for simple topology as cycle, it remains an open question for obtaining a more tractable expression for the hitting time of k-copy stateless opportunistic forwarding.

We remark that recently there are other studies of multi-copy opportunistic forwarding by multiple random walks [32], [33].

C. Cover Time

The cover time, the expected time of a packet from the source to visit every other node by random walk, is more difficult to obtain than hitting and commute time. A viable approach is to estimate based on the hitting time using Matthew's bound [14], [15].

Considering homogeneous transition times (i.e., $\ell_{(u,v)} = 1$), the cover time from node u, $\mathrm{Cover}(u)$, can be estimated by Matthew's bound as follows:

$$\sum_{i=1}^{n} \frac{1}{i} \min_{u,v} \mathbf{H}_{u,v} \le \mathrm{Cover}(u) \le \sum_{i=1}^{n} \frac{1}{i} \max_{u,v} \mathbf{H}_{u,v} \tag{38}$$

We next outline the proof of Matthew's bound and show that it can be also applied to the setting of heterogeneous transition times. For the upper bound, the basic idea is that we pick a random order of nodes, and observe when the random walk hits these nodes sequentially. Let T_k be the random time that the random walk hits all the nodes from the first to k-th node in the random order. Let $\mathbb{E}[T_k - T_{k-1}]$ be the expected difference between T_k and T_{k-1}. If the k-th node is hit before any preceding node in the random order, then $T_k = T_{k-1}$. The probability that such event does not occur is $\frac{1}{k}$. Hence,

$$\begin{aligned}\mathbb{E}[T_k - T_{k-1}] &= \tfrac{1}{k}\mathbb{E}\big[T_k - T_{k-1} \mid k\text{-th node is hit in } k\text{-th place}\big] \\ &\le \tfrac{1}{k} \max_{u,v} \mathbf{H}_{u,v}\end{aligned}$$

This proves the upper bound of Matthew's bound by summing $\mathbb{E}[T_k - T_{k-1}]$ for all k. The lower bound can be proved similarly. Note that such argument of proof does not rely on homogeneous transition times. Hence, Matthew's bound can be also applied to the case of heterogeneous transition times.

VII. Conclusion

We have studied stateless opportunistic forwarding as a random walk in a finite graph, and presented several exact results for the hitting and commute time of random walk in 1) arbitrary finite graphs with heterogeneous transition times at relaying nodes, 2) one dimensional r-nearest neighbor cycles, 3) two dimensional r-nearest neighbor tori. Particularly, we obtained good approximation formulas for the hitting time in r-nearest neighbor cycles and r-nearest neighbor tori. Moreover, we have applied our results to various settings, such as networks resulting from the presence of random geographical locations, and the extensions to topology-aware and multiple-copy forwarding strategies. We also discussed the evaluation of cover time. In the future, we will apply our results to practical applications augmented with empirical measurements, such as in delay tolerant networks and peer-to-peer networks.

Acknowledgment

The authors would like to thank the anonymous reviewers for detailed comments, and Saikat Guha, Don Towsley, Ananthram Swami, and colleagues of International Technology Alliance in Network and Information Science (ITA) program for helpful suggestions and discussions. Chi-Kin Chau is grateful to the Croucher Foundation Hong Kong for a research fellowship award.

References

[1] P. Hui, A. Chaintreau, R. Gass, J. Scott, J. Crowcroft, and C. Diot, "Pocket switched networking: Challenges, feasibility, and implementation issues," in *Proc. Workshop on Autonomic Communication (WAC)*, Greece, October 2005.

[2] S. Dolev, E. Schiller, and J. L. Welch, "Random walk for self-stabilizing group communication in ad hoc networks," *IEEE Trans. Mobile Computing*, vol. 5, no. 7, pp. 893–905, 2006.

[3] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks: algorithms and evaluation," *Performance Evaluation*, vol. 63, no. 3, pp. 241–263, March 2006.

[4] I. Mabrouki, X. Lagrange, and G. Froc, "Random walk based routing protocol for wireless sensor networks," in *Proc. Performance Evaluation Methodologies and Tools (ValueTools)*, 2007.

[5] V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, "Diversity of forwarding paths in pocket switched networks," in *Proc. ACM IMC*, 2007.

[6] C. Avin, M. Koucky, and Z. Lotker, "How to explore a fast-changing world (cover time of a simple random walk on evolving graphs)," in *Proc. ACM ICALP*, 2008.

[7] J. Redi, S. Kolek, K. Manning, C. Partridge, R. Rosales-Hain, R. Ramanathan, and I. Castineyra, "Javelen: An ultra-low energy ad hoc wireless network," *Ad Hoc Networks Journal*, vol. 5, pp. 108–126, January 2008.

[8] S. Nath and P. B. Gibbons, "Communicating via fireflies: Geographic routing on duty cycled sensors," in *Proc. of Intl. Conf. on Information Processing in Sensor Networks (IPSN)*, Cambridge, MA, April 2007.

[9] P. Basu and C.-K. Chau, "Opportunistic forwarding in wireless networks with duty cycling," in *Proc. ACM Workshop on Challenged Networks (CHANTS)*, September 2008.

[10] C.-F. Hsin and M. Liu, "Hitting time analysis for a class of random packet forwarding schemes in ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 3, pp. 500–515, May 2008.

[11] R. Beraldi, "Random walk with long jumps for wireless ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 2, pp. 294–306, March 2009.

[12] E. Gelenbe, "A diffusion model for packet travel time in a random multihop medium," *ACM Trans. Sensor Networks*, vol. 3, no. 2, pp. 1550–4859, June 2007.

[13] S. Guha, P. Basu, C.-K. Chau, and R. Gibbens, "Green wave sleep scheduling: Optimizing latency and throughput in duty cycling wireless networks," *to appear in IEEE J. on Selected Areas in Communications*, 2011.

[14] L. Lovasz, "Random walk on graphs: A survey," *Combinatorics, Paul Erdos is Eighty*, vol. 2, pp. 1–46, 1993.

[15] D. Aldous and J. A. Fill, "Reversible Markov chains and random walks on graphs," http://www.stat.berkeley.edu/~aldous/RWG/book.html, 1999, lecture notes.

[16] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*. AMS, 2009.

[17] P. G. Doyle and J. L. Snell, *Random Walks and Electrical Networks*. Mathematical Association of America, 1984.

[18] P. Tetali, "Random walks and the effective resistance of networks," *Journal of Theoretical Probability*, vol. 4, no. 1, pp. 1–46, 1991.

[19] A. K. Chandra, P. Raghavan, R. Smolensky, and P. Tiwari, "The electrical resistance of a graph captures its commute and cover times," *Computational Complexity*, vol. 6, no. 4, pp. 312–340, 1996.

[20] F. Chung and S.-Y. Yau, "Discrete Green's functions," *Journal of Combinatorial Theory, Series A*, vol. 91, pp. 191–214, 2000.

[21] R. B. Ellis, "Discrete Green's functions for products of regular graphs," 2003, arXiv:math/0309080v2.

[22] P. Basu and C.-K. Chau, "Latency of opportunistic forwarding in finite regular wireless networks," in *Proc. ACM Intl. Workshop on Foundation of Mobile Computing (DIAL M-POMC)*, August 2008.

[23] C. Avin and G. Ercal, "On the cover time and mixing time of random geometric graphs," *Theoretical Computer Science*, vol. 380, pp. 2–22, June 2007.

[24] I. Mabrouki, X. Lagrange, and G. Froc, "Analytical evaluation of unbiased random walks in wireless sensor networks," in *Proc. Autonomous and Spontaneous Networks Symposium*, 2008.

[25] ——, "Design and performance of wireless data gathering networks based on unicast random walk routing," *IEEE/ACM Trans. Networking*, vol. 17, no. 4, pp. 1214–1227, August 2009.

[26] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the single-copy case," *IEEE/ACM Trans. Networking*, vol. 16, no. 1, pp. 63–76, February 2008.

[27] ——, "Efficient routing in intermittently connected mobile networks: the multiple-copy case," *IEEE/ACM Trans. Networking*, vol. 16, no. 1, pp. 77–90, February 2008.

[28] M. Sigelle, I. H. Jermyn, S. Perreau, and A. Jayasuriya, "Lattice green functions and diffusion for modelling traffic routing in ad hoc networks," in *Proc. Workshop on Physics-Inspired Paradigms in Wireless Communications and Networks (PHYSCOMNET)*, 2009.

[29] C.-K. Chau and P. Basu, "An analysis of latency of stateless opportunistic forwarding in intermittently connected networks," Computer Laboratory, University of Cambridge, Tech. Rep., 2010.

[30] Y. Li and Z.-L. Zhang, "Random walks on digraphs: A theoretical framework for estimating transmission costs in wireless routing," in *Proc. IEEE INFOCOM*, March 2010.

[31] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Information Theory*, vol. 52, no. 6, June 2006.

[32] C. Santivanez and R. Ramanathan, "Hazy sighted link state (HSLS) routing: A scalable link state algorithm," BBN Technologies, Tech. Rep., 2001, BBN-TM-1301.

[33] N. Alon, C. Avin, M. Koucky, G. Kozma, Z. Lotker, , and M. Tuttle, "Many random walks are faster than one," in *Proc. ACM SPAA*, 2008.

[34] C. Cooper, A. Frieze, and T. Radzik, "Multiple random walks in random regular graphs," *SIAM Journal on Discrete Mathematics*, vol. 380, no. 4, pp. 1738–1761, 2009.

[35] D. Geller, I. Kra, S. Popescu, and S. Simanca, "On circulant matrices," http://www.math.sunysb.edu/~sorin/, lecture notes.

[36] "Wikipedia: Trigonometric identities," http://en.wikipedia.org/wiki/Trigonometric_identity.

VIII. Appendix

A. Definition of Tensor Product

Suppose $\mathbf{X}$ is an $(r+1)\times(s+1)$ matrix and $\mathbf{Y}$ is a $(p+1)\times(q+1)$ matrix. Define the tensor product matrix as $\mathbf{X}\otimes\mathbf{Y}$, which is a $(r+1)(p+1)\times(s+1)(q+1)$ matrix, such that for $0\le a\le r$, $0\le b\le p$, $0\le c\le s$, $0\le d\le q$, the entry at position $(a(p+1)+b, c(q+1)+d)$ is defined as:

$$(\mathbf{X}\otimes\mathbf{Y})_{a(p+1)+b,\ c(q+1)+d} \triangleq \mathbf{X}_{a,c}\mathbf{Y}_{b,d} \tag{39}$$

B. Proof of Theorems and Lemmas

Lemma 4: Let $\mathbf{J}$ be the all-ones matrix. We reformulate Eqn. (1) as a matrix equation as $\mathbf{\Phi}\triangleq\mathbf{SJ}+(\mathbf{D}^{-1}\mathbf{A}-\mathbf{I})\mathbf{H}$, where the non-diagonal entry $\mathbf{\Phi}_{u,v}$ represents the recursive relationship of hitting times in Eqn. (1). Then,

$$\mathbf{\Phi}_{u,v} = \begin{cases} \frac{\sum_{u'\in\mathcal{V}}\pi_{u'}\ell_{u'}}{\pi_u} & \text{if } u=v \\ 0 & \text{otherwise} \end{cases} \tag{40}$$

Proof: Note that $(\mathbf{SJ})_{u,v}=\ell_u$. Since the non-diagonal entries $\mathbf{\Phi}_{u,v}$ represent Eqn. (1),

$$\mathbf{\Phi}_{u,v}=\mathbf{SJ}+(\mathbf{D}^{-1}\mathbf{A}-\mathbf{I})\mathbf{H}=0, \quad \text{for} \quad u\neq v \tag{41}$$

Namely, $\mathbf{\Phi}$ is a diagonal matrix. Also, we observe that the u-th entry of $\pi^T\mathbf{\Phi}$ is:

$$\left(\pi^T\mathbf{\Phi}\right)^T_u = \left(\pi^T\mathbf{SJ}\right)^T_u = \sum_{u'\in\mathcal{V}}\pi_{u'}\ell_{u'} \tag{42}$$

Therefore, the diagonal entry $\mathbf{\Phi}_{u,u}=\frac{\sum_{u'\in\mathcal{V}}\pi_{u'}\ell_{u'}}{\pi_u}$. ■

Theorem 1: Given arbitrary transition time $\ell_u>0$ for each u, the hitting time and commute time from u to v can be computed by:

$$\mathbf{H}_{u,v} = \sum_{k:\sigma_k\neq 0}\frac{\tilde{W}}{\sigma_k}\Big(\frac{\boldsymbol{\mu}_{k,v}^2}{w_v\ell_v}-\frac{\boldsymbol{\mu}_{k,u}\boldsymbol{\mu}_{k,v}}{\sqrt{w_u\ell_u w_v\ell_v}}\Big) \tag{43}$$

$$\mathbf{C}_{u,v} = \sum_{k:\sigma_k\neq 0}\frac{\tilde{W}}{\sigma_k}\Big(\frac{\boldsymbol{\mu}_{k,u}}{\sqrt{w_u\ell_u}}-\frac{\boldsymbol{\mu}_{k,v}}{\sqrt{w_v\ell_v}}\Big)^2 \tag{44}$$

where $\tilde{W}\triangleq\sum_{(u,v)\in\mathcal{E}}w_{u,v}(\ell_{u,v}+\ell_{v,u})=\sum_{u'\in\mathcal{V}}w_{u'}\ell_{u'}$, and σ_k and $\boldsymbol{\mu}_k$ are the k-th eigenvalue and eigenvector of $\tilde{\mathbf{L}}$.

Proof: First, we let $\boldsymbol{\omega}_u=\sqrt{w_u\ell_u}$. Then we observe that

$$\tilde{\mathbf{L}}_{u,u}\boldsymbol{\omega}_u=\sqrt{\frac{w_u}{\ell_u}} \quad \text{and} \tag{45}$$

$$\sum_{v\in\mathcal{N}_u}\tilde{\mathbf{L}}_{v,u}\boldsymbol{\omega}_v=-\frac{\sum_{v\in\mathcal{N}_u}w_{u,v}}{\sqrt{w_u\ell_u}}=-\sqrt{\frac{w_u}{\ell_u}} \tag{46}$$

Hence, $\boldsymbol{\omega}^T\tilde{\mathbf{L}}=0$. That is, 0 is an eigenvalue, and $\boldsymbol{\omega}/\sqrt{\tilde{W}}$ is the corresponding eigenvector of $\tilde{\mathbf{L}}$. Note that since $\mathcal{G}$ is connected, by Frobenius-Perron Theorem [14], one can show that $\sigma_0=0$ is the unique eigenvalue.

Second, define (generalized) discrete Green's function $\tilde{\mathbf{G}}$:

$$\tilde{\mathbf{G}}\triangleq\sum_{k:\sigma_k\neq 0}\frac{1}{\sigma_k}\boldsymbol{\mu}_k\boldsymbol{\mu}_k^T \tag{47}$$

Note that this is the same definition in [19], when $\ell_u=1$ for all u. Since $\boldsymbol{\omega}/\sqrt{\tilde{W}}$ is an eigenvector of $\tilde{\mathbf{L}}$, it follows that:

$$\boldsymbol{\omega}^T\tilde{\mathbf{G}}=0 \quad\text{and}\quad \tilde{\mathbf{G}}\tilde{\mathbf{L}}=\mathbf{I}-\frac{1}{\tilde{W}}\boldsymbol{\omega}\boldsymbol{\omega}^T \tag{48}$$

Thrid, by Lemma 4 and $\pi_u=\frac{w_u}{W}$, the diagonal entry is:

$$\left(\mathbf{SJ}+(\mathbf{D}^{-1}\mathbf{A}-\mathbf{I})\mathbf{H}\right)_{u,u}=\frac{\sum_{u'\in\mathcal{V}}\pi_{u'}\ell_{u'}}{\pi_u}=\frac{\sum_{u'\in\mathcal{V}}w_{u'}\ell_{u'}}{w_u}=\frac{\tilde{W}}{w_u}$$

Combining these facts, we obtain:

$$\begin{aligned}
\mathbf{SJ}+(\mathbf{D}^{-1}\mathbf{A}-\mathbf{I})\mathbf{H} &= \tilde{W}\mathbf{D}^{-1} && (49)\\
(\mathbf{I}-\mathbf{D}^{-1}\mathbf{A})\mathbf{H} &= \mathbf{SJ}-\tilde{W}\mathbf{D}^{-1} && (50)\\
\mathbf{D}^{\frac{1}{2}}(\mathbf{I}\text{-}\mathbf{D}^{-1}\mathbf{A})\mathbf{D}^{-\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}\mathbf{H}\mathbf{D}^{\frac{1}{2}} &= \mathbf{D}^{\frac{1}{2}}(\mathbf{SJ}\text{-}\tilde{W}\mathbf{D}^{-1})\mathbf{D}^{\frac{1}{2}} && (51)\\
\mathbf{S}^{-\frac{1}{2}}(\mathbf{I}\text{-}\mathbf{N})\mathbf{S}^{-\frac{1}{2}}\mathbf{S}^{\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}\mathbf{H}\mathbf{D}^{\frac{1}{2}} &= \mathbf{S}^{-\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}(\mathbf{SJ}\text{-}\tilde{W}\mathbf{D}^{-1})\mathbf{D}^{\frac{1}{2}} && (52)\\
\tilde{\mathbf{L}}\mathbf{S}^{\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}\mathbf{H}\mathbf{D}^{\frac{1}{2}}\mathbf{S}^{\frac{1}{2}} &= \mathbf{S}^{-\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}(\mathbf{SJ}\text{-}\tilde{W}\mathbf{D}^{-1})\mathbf{D}^{\frac{1}{2}}\mathbf{S}^{\frac{1}{2}} &&\\
\tilde{\mathbf{L}}\mathbf{S}^{\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}\mathbf{H}\mathbf{D}^{\frac{1}{2}}\mathbf{S}^{\frac{1}{2}} &= \tilde{W}(\frac{1}{\tilde{W}}\boldsymbol{\omega}\boldsymbol{\omega}^T\text{-}\mathbf{I}) && (53)\\
\tilde{\mathbf{G}}\tilde{\mathbf{L}}\mathbf{S}^{\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}\mathbf{H}\mathbf{D}^{\frac{1}{2}}\mathbf{S}^{\frac{1}{2}} &= \tilde{W}\tilde{\mathbf{G}}(\frac{1}{\tilde{W}}\boldsymbol{\omega}\boldsymbol{\omega}^T\text{-}\mathbf{I}) && (54)\\
(\mathbf{I}\text{-}\frac{1}{\tilde{W}}\boldsymbol{\omega}\boldsymbol{\omega}^T)\mathbf{S}^{\frac{1}{2}}\mathbf{D}^{\frac{1}{2}}\mathbf{H}\mathbf{D}^{\frac{1}{2}}\mathbf{S}^{\frac{1}{2}} &= -\tilde{W}\tilde{\mathbf{G}} && (55)
\end{aligned}$$

Eqn. (53) is because that $\mathbf{D}$ and $\mathbf{S}$ are diagonal matrices. The diagonal entry (u,u) of the above matrix equation is:

$$w_u\ell_u\mathbf{H}_{u,u}-\frac{w_u\ell_u}{\tilde{W}}\sum_{w\in\mathcal{V}}w_w\ell_w\mathbf{H}_{w,u}=-\tilde{W}\sum_{k:\sigma_k\neq 0}\frac{1}{\sigma_k}\boldsymbol{\mu}_{k,u}^2 \tag{56}$$

while the non-diagonal entry (u,v) is:

$$\sqrt{w_u\ell_u w_v\ell_v}\mathbf{H}_{u,v}-\frac{\sqrt{w_u\ell_u w_v\ell_v}}{\tilde{W}}\sum_{w\in\mathcal{V}}w_w\ell_w\mathbf{H}_{w,u}$$
$$=-\tilde{W}\sum_{k:\sigma_k\neq 0}\frac{1}{\sigma_k}\boldsymbol{\mu}_{k,u}\boldsymbol{\mu}_{k,v} \tag{57}$$

Combining the two equations and using $\mathbf{H}_{u,u}=0$, we obtain Eqn. (13). Eqn. (14) follows from $\mathbf{C}_{u,v}=\mathbf{H}_{u,v}+\mathbf{H}_{v,u}$. ∎

Lemma 1: The $(k+1)$-th eigenvalue and eigenvector of the Laplacian $\tilde{\mathbf{L}}$ of r-nearest neighbor cycle $\mathcal{C}_n^r$:

$$\sigma_k=1-\frac{\sum_{j=1}^r a_j\cos\left(\frac{2\pi jk}{n}\right)}{\sum_{j'=1}^r a_{j'}},\ \boldsymbol{\mu}_k=\frac{1}{\sqrt{n}}\left(1,\epsilon^k,...,\epsilon^{(n-1)k}\right)^T$$

ϵ is complex number defined as: $\epsilon\triangleq\cos\left(\frac{2\pi}{n}\right)+\mathbf{i}\sin\left(\frac{2\pi}{n}\right)$.

Proof: The adjacency matrix $\mathbf{A}[\mathcal{C}_n^r]$ is:

$$\begin{pmatrix} 0 & a_1 & \dots & a_2 & a_1\\ a_1 & 0 & \dots & a_3 & a_2\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ a_1 & a_2 & \dots & a_1 & 0\end{pmatrix} \tag{58}$$

which is a circulant matrix [34]. It is well-known that the k-th eigenvector is $\boldsymbol{\mu}_k$ and the corresponding eigenvalue is:

$$\begin{aligned}&a_1\epsilon^k+\dots+a_r\epsilon^{rk}+a_r\epsilon^{(r-1)k}+\dots+a_1\epsilon^{(n-1)k}\\ &=2\sum_{j=1}^r a_j\cos\left(\frac{2\pi jk}{n}\right)\end{aligned} \tag{59}$$

Finally, $W=2\sum_{j'=1}^r a_{j'}$ and $\tilde{\mathbf{L}}[\mathcal{C}_n^r]=\mathbf{I}-\frac{\mathbf{A}[\mathcal{C}_n^r]}{2\sum_{j'=1,r}a_{j'}}$. ∎

Lemma 5: Trigonometric identity of Dirichlet kernel [35]:

$$1+2\sum_{j=1}^r\cos(jx)=\frac{\sin\left((r+\frac{1}{2})x\right)}{\sin(\frac{x}{2})} \tag{60}$$

Lemma 6: When $x\to 0$,

$$\frac{1}{1-\frac{1}{r}\sum\limits_{j=1}^r\cos(jx)}\approx\frac{12}{(r+1)(2r+1)x^2} \tag{61}$$

Proof: By Lemma 5 and Taylor series expansion at $x=0$,

$$\frac{\sin\left((r+\frac{1}{2})x\right)}{\sin(\frac{x}{2})}=(2r+1)-\frac{1}{6}r(r+1)(2r+1)x^2+O(x^4)$$

∎

By numerical study, we observe that Eqn. (61) gives a relatively good approximation even for $0\leq x\leq\pi$ and small r (see Fig. 16).

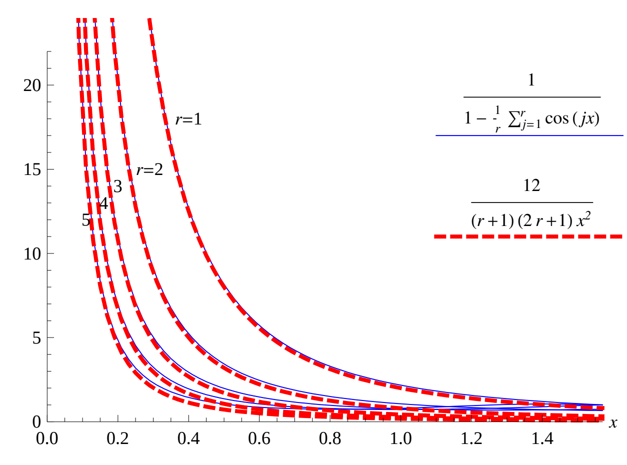


Fig. 16. The comparison of $\frac{1}{1-\frac{1}{r}\sum\limits_{j=1}^r\cos(jx)}$ against $\frac{12}{(r+1)(2r+1)x^2}$ for $0\leq x\leq\pi$.

Theorem 2: When $n\to\infty$, the maximum hitting time in r-nearest neighbor cycle $\mathcal{C}_n^r$ can be approximated by:

$$\mathbf{H}_{\frac{n}{2},0}\approx\frac{3n^2}{2(1+r)(1+2r)} \tag{62}$$

Proof: It suffices to consider the case $n = 4c$ for integer c, because we can always extrapolate in the approximation.

$$\mathbf{H}_{\frac{n}{2},0} = \sum_{k=1}^{n-1} \frac{1 - \cos(\pi k)}{1 - \frac{1}{r}\sum_{j=1}^{r} \cos\left(\frac{2\pi jk}{n}\right)} \quad (63)$$

$$= \sum_{k=0}^{\frac{n}{2}} \frac{2}{1 - \frac{1}{r}\sum_{j=1}^{r} \cos\left(\frac{2\pi j(2k+1)}{n}\right)} \quad (64)$$

$$\overset{(1)}{=} \sum_{k=0}^{\frac{n}{4}} \frac{4}{1 - \frac{1}{r}\sum_{j=1}^{r} \cos\left(\frac{2\pi j(2k+1)}{n}\right)} \quad (65)$$

$$\overset{(2)}{\approx} \sum_{k=0}^{\frac{n}{4}} \frac{12n^2}{\pi^2(r+1)(r+1)(2k+1)^2} \quad (66)$$

The derivation of (1) follows from the fact that $\cos\left(\frac{2\pi j(2k+1)}{n}\right) = \cos\left(\frac{2\pi j(n-2k-1)}{n}\right)$, whereas (2) follows from Lemma 6.

We complete the proof by the following identity:

$$\lim_{n\to\infty} \sum_{k=0}^{n} \frac{1}{(2k+1)^2} = \frac{\pi^2}{8} \quad (67)$$

■

Lemma 7: Consider the settings of *Example 1* and *Example 2* in Sec. II. Let $L^{(u)}$ be the random number of slots for node u before a neighbor becomes active. Therefore, $L^{(u)} = \min\{L_1, L_2, \ldots, L_{d_u}\}$, where L_v is the random waiting time for neighbor $v \in \mathcal{N}_u$ to become active.

1) (*I.I.D. Random Duty Cycling Sensor Networks*):

$$\ell_u = \mathbb{E}[L^{(u)}] = \frac{1}{1 - (1 - \rho_{\mathsf{dc}}^2)^{d_u}} \quad (68)$$

2) (*I.I.D. Pseudo-Random Duty Cycling Sensor Networks*):

$$\ell_u = \mathbb{E}[L^{(u)}] = \frac{1}{1 - (1 - \rho_{\mathsf{dc}})^{d_u}} \quad (69)$$

Proof: We consider pseudo-random duty cycling, as random duty cycling can be proven similarly. Note that each L_v is a geometrically distributed random variable with parameter ρ_{dc}. For $t \geq 1$, the per-hop latency probability distribution is:

$$\mathbb{P}\big(L^{(u)} \geq t\big) = \mathbb{P}\big(\min\{L_1, L_2, \ldots, L_{d_u}\} \geq t\big) \quad (70)$$

$$= \mathbb{P}\big(L_v \geq t, \text{for all } j \in \mathcal{N}_u,\big) \quad (71)$$

$$= (1 - \rho_{\mathsf{dc}})^{(t-1)d_u} \quad (72)$$

Since $L^{(u)}$ is non-negative, we obtain:

$$\mathbb{E}[L^{(u)}] = \sum_{t=1}^{\infty} \mathbb{P}\big(L^{(u)} \geq t\big) \quad (73)$$

$$= \sum_{t=1}^{\infty} (1 - \rho_{\mathsf{dc}})^{(t-1)d_u} = \frac{1}{1 - (1 - \rho_{\mathsf{dc}})^{d_u}} \quad (74)$$

■