"A Token Based Approach to Distributed Computation in Sensor Networks"

V. Saligrama and M. Alanyali

CISE Technical Report

CISE Technical Report #2010-IR-0010
CISE Technical Report Date: August 2, 2010

Center for Information and Systems Engineering
15 St. Mary's Street, Brookline, MA 02446
617.358.1295

A Token Based Approach to Distributed Computation in Sensor Networks *

Venkatesh Saligrama, Murat Alanyali
Department of Electrical and Computer Engineering
Boston University, Boston, MA 02215, USA
Email: {srv,alanyali}@bu.edu

Abstract

We consider distributed algorithms for data aggregation and function computation in sensor networks. The algorithms perform pairwise computations along edges of an underlying communication graph. A token is associated with each sensor node, which acts as a transmission permit. Nodes with active tokens have transmission permits; they generate messages at a constant rate and send each message to a randomly selected neighbor. By using different strategies to control the transmission permits we can obtain tradeoffs between message and time complexity. Gossip corresponds to the case when all nodes have permits all the time. We study algorithms where permits are revoked after transmission and restored upon reception. Examples of such algorithms include Simple-Random Walk(SRW), Coalescent-Random-Walk(CRW) and Controlled Flooding(CFLD) and their hybrid variants. SRW has a single node permit, which is passed on in the network. CRW, initially initially has a permit for each node but these permits are revoked gradually. The final result for SRW and CRW resides at a single(or few) random node(s) making a direct comparison with GOSSIP difficult. A hybrid two-phase algorithm switching from CRW to CFLD at a suitable pre-determined time can be employed to achieve consensus. We show that such hybrid variants achieve significant gains in both message and time complexity. The per-node message complexity for n-node graphs, such as 2D mesh, torii, and Random geometric graphs, scales as $O(polylog(n))$ and the corresponding time complexity scales as $O(n)$. The reduced per-node message complexity leads to reduced energy utilization in sensor networks.

1 Introduction

Large sensor systems have remarkable potential in a wide range of applications from environmental monitoring to intrusion detection. Such systems are now viable thanks to recent progress in integration and communication technologies, yet their size precludes classical telemetry to collect sensory data and poses algorithmic challenges in handling the high data volume. The principle of gossip offers an appealing and scalable solution approach to this issue: In broad terms, gossip algo-

*This research was supported by NSF Grant 0932114 and NSF CAREER awards ECS-0449194 and CNS-0238397.

rithms are decentralized methods to compute statistics of system-wide data based on asynchronous message passing between sensors that are within immediate communication range. The purpose of this paper is to introduce and analyze token-based gossip algorithms, and put them in perspective with previously studied gossip algorithms with respect to computation time, energy consumption, accuracy, and robustness.

The generic gossip algorithm has been well-studied in the context of computing averages [7, 15, 22], and has roots in load balancing (see, for example, [5, Section7.4]). Here it is assumed that each sensor has a local scalar value and it is of interest to compute the average of these values. Gossip algorithms accomplish this task by randomly choosing two neighboring sensor nodes at each time and replacing their current values by their average. It turns out that under mild conditions this process over time converges to the average of all sensor values at all the sensors, i.e., the sensors asymptotically achieve a consensus. The algorithm executes autonomously at each sensor; it is robust to communication errors and sensor failures; and a final state of consensus provides robustness and convenience in reading the average from the system. Such consensus algorithms have been recently explored in other contexts such as detection [1, 20].

Nevertheless, these algorithms have fundamental disadvantages from an energy efficiency perspective. For example it is well known that in grids and in tori with n nodes, the number of message transmissions *per node* to complete the computational task scales as $\Omega(n)$ [7, 21], which can be significant for a large sensor network. The fundamental reason is that energy efficiency resulting from in-network processing is offset by ad-hoc message passing that results in redundant computations, i.e., the same set (or largely similar set) of nodes repeatedly fuse their information at different points in time. In a related problem involving distributed detection, the significant energy scaling can be attributed to the loopy nature of the network where messages sent from one node repeatedly arrive at the node at different points in time. In order to ensure that no information from any node is forgotten, each node must re-inject its value into the network to reinforce its information [20]. At a fundamental level the significant scaling of energy arises due to the slow mixing rate of large networks, which can be attributed to rather large second eigenvalues of certain connectivity matrices associated with the underlying communication graph [7][Theorem 3].

Another important disadvantage of generic gossip algorithms concerns accuracy. Stopping criterion of gossip is based on tail probabilities of a normalized distance between current system state and state of consensus [7]. In turn, these algorithms provide only probabilistic guarantees on final consensus and therefore they should be considered as approximation algorithms. It should also be noted that even when such a guarantee holds and the final normalized error happens to be small, the actual error may be substantial if the normalizing constant is large. Such situations arise in large networks in which a small set of sensor values significantly influence the sought value.

To put the present paper in perspective with gossip algorithms it helps to consider signal processing and communications *separately*. Here we adopt the objective of *exact* computation of a function of distributed data, and consider performance of a novel class of communication algorithms towards that end. In this view conventional gossip may be considered as a communication algorithm to *ap-*

proximate the same function. This approximation is clearly one specific instantiation; and there are other wide range of approximation criteria that may be useful to consider from a signal processing perspective. From a signal processing point of view it also makes sense to incorporate prior signal information, such as boundedness, distribution of values etc. While these issues are important our focus here is primarily on the communication aspects for exact distributed computation.

1.1 Token-Based Algorithms

We present a novel concept of token-based gossip for distributed computing. This concept preserves the ad-hoc network operation but entails perfect accuracy of computation and exponential savings in energy-consumption over the existing local message passing algorithms. Under the algorithms studied here, a transmitting node becomes inactive and does not transmit further messages until it is reactivated by a message reception from another node. An active node generates messages at constant rate and sends each message to a randomly selected neighbor. Hence network nodes implement interactive sleep-wake schedules. Active nodes are interpreted to hold imaginary tokens that act as transmission permits. The total energy consumption is controlled by managing the number of tokens in the network.

We describe different instances of token based algorithms: (i) Algorithm SRW maintains a single active node (i.e. a single token), whose trajectory is a random walk on the communication graph. Local processing at each active node exploits a decomposability property of the considered function to guarantee that the function is computed when each node becomes active at least once. In turn performance of this algorithm is closely related to the cover time [24] of random walks. (ii) Under algorithm CRW all nodes are initially active but when two active nodes communicate with each other their tokens coalesce and therefore the number of tokens in the system reduces by one. The computation is completed when a single token remains in the system. This latter algorithm is closely related with coalescing random walks [8] that have been studied as duals of a class of interacting particle systems coined as voter models. The two algorithms are illustrated in Figures 1(a)-(b).
A range of distributed algorithms can be obtained by variations of the token-based communication concept. For example, conventional gossip can be considered as a token-based algorithm where each node maintains a token at all times. The local processing upon each message exchange in this case is illustrated in Figure 2(a). Alternatively, one may consider hybrid schemes to improve message and time complexity. One hybrid scheme involves with a fixed but arbitrary number of tokens. Tokens progress from an active node to an inactive node while updating local values exactly as in SRW. If two active nodes interact then they both relax their values as in conventional gossip, and remain active. An illustration of such a scheme is given in Figure 2(b). Another hybrid scheme involves switching at some time t, from CRW to Controlled Flooding (CFLD) [6], which is a token based algorithm where tokens multiply at each local broadcast. In contrast to flooding, CFLD follows additional rules to control the number of transmissions(see Section 5).

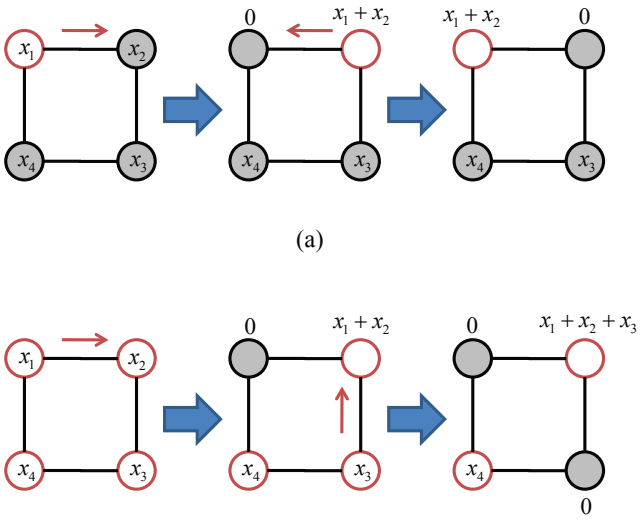


Figure 1: Illustration of token-based algorithms: (a) SRW and (b) CRW. Each node is either active or inactive. Active nodes are indicated with light color. Nodes can transmit information in the active state but not in the inactive state. A node can transition from active to inactive or vice versa based on pre-defined protocol. In SRW there is always one active node. In CRW every node is active initially but the number of active nodes decrease in time, towards the final value one.

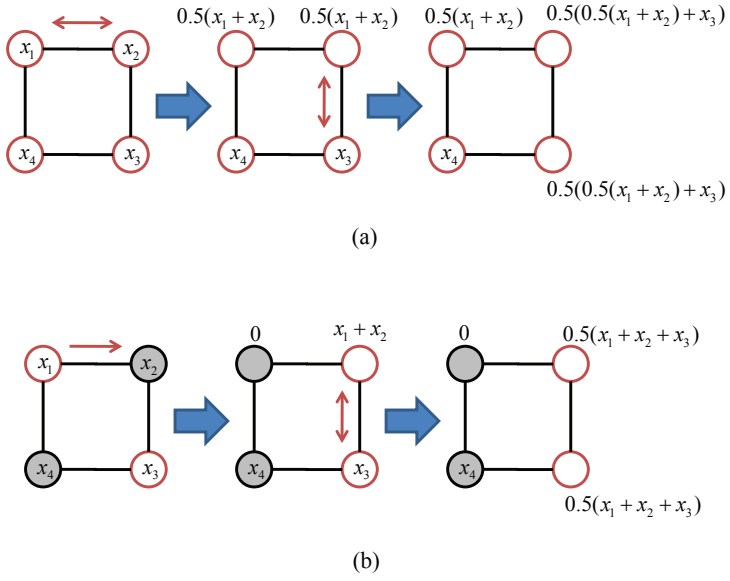


Figure 2: Illustration of two algorithms in the framework of the paper. (a) In conventional gossip all nodes are active all the time. Transmission is two-way and two nodes that share information replace their prior values with the fused values. (b) Hybrid token algorithm that maintains a constant (in this case 2) number of active nodes. Transactions between an active and an inactive node are governed by the rules of SRW, while transactions between two active nodes are governed by rules of conventional gossip.

1.2 Time and Message Complexities

In this section we describe tradeoffs between time and message complexities for regular graphs to illustrate some of the benefits of SRW and CRW. However, a fundamental difference between Gossip algorithm and SRW/CRW makes this comparison potentially difficult. Note that in the standard Gossip algorithm, the fused value is a consensus estimate at all the nodes. In contrast SRW/CRW realize the fused value at a random node in the network. *In practice it is desirable to have access to the fused value at a designated node(s).* Consequently, to meet this requirement, SRW/CRW would have to transmit the fused estimate to the designated fusion center. This raises two fundamental issues:
(A) How can a node recognize that it has the fused estimate?
(B) How to efficiently transmit the fused information to the designated fusion center?
As it turns out, both of these questions can be addressed satisfactorily. To address the first issue we augment the distributed computation problem with a secondary distributed computation procedure. This secondary computation determines when fusion has been realized. To transmit this information to the designated node(s) we *flood* the entire network through CFLD. The overall message complexity for CFLD for a single message scales as the number of communication links in the network. The time complexity scales as the diameter of the network [14]. Furthermore, other choices can result in superior performance. For instance, nodes follow the CRW protocol until a predesignated time t and switch to CFLD after time t.

Table 1 (see Section 4 and 6) illustrates completion times and per-node transmission counts for regular topologies. For the d-dimensional lattice torus with n sensors, we the completion time of CRW is $\Theta(n(\log n)^{\alpha})$ and energy requirement per sensor node is $O((\log n)^{\alpha+1})$ where $\alpha = 1$ for $d = 2$ and $\alpha = 0$ for $d \geq 3$. The algorithm thus has a favorable energy scaling, furthermore its performance is almost insensitive to changes in the network connectivity represented by different values of $d \geq 2$, hinting at the possibility of predictable performance over mesh topologies. Both the time and the per-node energy requirement of the generic gossip algorithm of [7] scale as $\Omega(n)$ on the 2-dimensional torus with n nodes. We also depict results for the two phase CRW + CFLD algorithm. For the 2D torus the time complexity is similar to GOSSIP and the message complexity is similar to CRW. Therefore, this two-phase scheme is an improvement over both GOSSIP and CRW. Recent results indicate that the energy scaling can be improved significantly via variations of gossip that require location awareness capability for each sensor [11, 16]. Similar variations of token-based gossip may also yield reduction in energy complexity, though that direction is not pursued in this paper.

The conclusions of Table 1 (see Sections 4 and 6) for regular topologies is presented in Section 3.2 and is largely based on existing results in applied probability literature. Preliminary work along these lines has also been described by the authors [21]. This paper develops results for general topologies based on a deeper analysis of hitting-time computations on the communication graph. While our approach can lead to conservative estimates for general graphs, it turns out that for

	CRW	SRW	GOSSIP	CRW + CFLD
Clique	$\Theta(\log n)$	$\Theta(\log n)$	$\Theta(n)$	$\Theta(\log n)$
Ring	$\Theta(n)$	$\Theta(n)$	$\Theta(n^2)$	$\Theta(n)$
Torus ($d = 2$)	$O(\log^2 n)$	$\Theta(\log^2 n)$	$\Theta(n)$	$O(\log^2 n)$
Torus ($d \geq 3$)	$O(\log n)$	$\Theta(\log n)$	$\Theta(n^{2/d})$	$O(d + \log n)$

(a)

	CRW	SRW	GOSSIP	CRW + CFLD
Clique	$\Theta(n)$	$\Theta(n \log n)$	$\Theta(n)$	$\Theta(n)$
Ring	$\Theta(n)$	$\Theta(n \log n)$	$\Theta(n^2)$	$\Theta(n)$
Torus ($d = 2$)	$\Theta(n \log n)$	$\Theta(n \log^2 n)$	$\Theta(n)$	$\Theta(n)$
Torus ($d \geq 3$)	$\Theta(n)$	$\Theta(n \log n)$	$\Theta(n^{2/d})$	$\Theta(n)$

(b)

Table 1: (a) Message and (b) time complexities with n nodes.

graphs with local neighborhood structure, such as grids and random geometric graphs(RGG), these estimates are relatively tighter. Our message complexity for RGG scales as $O(\log^2(n))$ and can be combined with CFLD to realize consensus with significant improvement in message and time complexity over GOSSIP.

Robustness: Similar to other token-based communication algorithms such as token rings, robustness of the introduced algorithms suffers from the potential of losing tokens. Packet losses do not contribute to this potential if reliable link protocols are adopted. However permanent node failures may have an impact on system performance if a node fails while holding a token. The issue may be mitigated by running multiple independent instances of a token-based algorithm simultaneously, in order to reduce the likelihood of losing all tokens simultaneously at a failing node, without altering the scaling of time and message complexities. As will be clear in the sequel, a token-bearing node knows explicitly the number of sensor values fused in its current value; and that value may be useful partial information in case of node failures. Alternatively a hybrid scheme with multiple tokens may be invoked, thereby providing a robustness akin to conventional gossip.

Paper Organization: The paper is organized as follows. In Section 2 we formalize a general distributed computation problem that includes computation of statistics of spatially dispersed sensory data. The two token-based gossip algorithms SRW and CRW are formally specified in Section 3 and their correctness is established. Section 3.2 illustrates time and message complexities for regular topologies as summarized by Table 1. Section 5 describes two-phase algorithms that combine CRW with CFLD to realize a consensus fused estimate. Section 6 gives a novel analysis of coalescing random walks on arbitrary graphs and thereby determines the complexity of CRW in the general setting.

2 Problem formulation

We consider a collection of n networked sensors. The communication graph of this collection is an undirected graph $G = (V, E)$ in which each sensor is uniquely represented by a node in V. An edge in E indicates that the two sensors corresponding to its incident nodes have a bidirectional communication link. In order to avoid trivialities G is assumed to be connected; otherwise it is arbitrary.

Each sensor i has a value x_i and we are interested in computing a function $F_n(x_1, x_2, \cdots, x_n)$ of the n sensor values. The function $F_n(\cdot)$ is assumed to be symmetric so that for any permutation π of $\{1, 2, \cdots, n\}$

$$F_n(x_1, x_2, \cdots, x_n) = F_n(x_{\pi_1}, x_{\pi_2}, \cdots, x_{\pi_n}).$$

Furthermore we assume existence of an atomic function $f(\cdot)$ such that for each $1 \leq k \leq n$

$$F_n(x_1, x_2, \cdots, x_n) = f(F_k(x_{\pi_1}, \cdots, x_{\pi_k}), F_{n-k}(x_{\pi_{k+1}}, x_{\pi_{k+2}}, \cdots, x_{\pi_n})). \tag{1}$$

In particular

$$f(x_i, x_j) = F_2(x_i, x_j). \tag{2}$$

For example if $f(x_i, x_j) = \max(x_i, x_j)$ or if $f(x_i, x_j) = x_i + x_j$ then $F_n(\cdot)$ is respectively the maximum or the sum of $x_1, x_2, \cdots, x_n$. If each $x_i = (y_i, w_i)$ is a vector and $f(\cdot)$ is the vector-valued function

$$f(x_i, x_j) = ((w_i + w_j)^{-1}(w_i y_i + w_j y_j)\ ,\ w_i + w_j),$$

then $F_n(\cdot)$ is the tuple

$$F_n(x_1, x_2, \cdots, x_n) = \left(\sum_{i=1}^{n} \frac{w_i}{\sum_{j=1}^{n} w_j} y_i\ ,\ \sum_{i=1}^{n} w_i \right).$$

Weighted average computations of this sort are particularly relevant to applications of Kalman filtering in distributed tracking [19].

We finally assume existence of a special value e that acts as an identity element for $f(\cdot)$ so that for any value x

$$f(x, e) = f(e, x) = f(e, e) = e.$$

This element is not a fundamental requirement, and in fact it makes the upcoming algorithm specifications look somewhat mysterious, but it is useful in giving a concise reasoning about correctness of the algorithms introduced next.

3 Token-based Gossip Algorithms

We specify two algorithms, namely SRW and CRW. A pseudo-code for these algorithms is given in Figure 1. Under each algorithm, each node maintains three variables `value`, `status`, and `count`.

Algorithm 1 Pseudo-code for algorithms SRW and CRW at node i. Send() is activated by the local Poisson clock at the node, and Receive() is activated by message reception from some other node. The two algorithms differ in the initialization of the variable `status`. Each algorithm terminates when `count` is equal to the number of nodes in system.

Variables: `status, value, count`.

Initialize: `value` $\leftarrow x_i$; `count` $\leftarrow 1$;

$$\text{SRW}: \quad \texttt{status} \leftarrow \begin{cases} \text{'active'} & \text{if } i = i_o; \\ \text{'inactive'} & \text{else.} \end{cases}$$

CRW : `status` $\leftarrow$ 'active'.

Procedure **Send**()

```
if( status == 'active' ) {
 choose neighbor;
 send(neighbor , value, count);
 value ← e;
 count ← 0;
 status ← 'inactive';
 }
```

Procedure **Receive**(`value_in, count_in`) {

```
value ← f(value, value_in);
count ← count + count_in;
status ← 'active';
}
```

Content of `value` is an estimate of $F_n(x_1, x_2, \cdots .x_n)$. Initially `value`$= x_i$ and `count`$= 1$ at each node i. The variable `status` is either *'active'* or *'inactive'* and it indicates whether the node is holding a token or not. Let

$$\begin{aligned} v_i(t) &= \text{content of } \texttt{value} \text{ at node } i \text{ at time } t, \\ \xi_i(t) &= \begin{cases} 1 & \text{if } \texttt{status} \text{ of node } i \text{ is 'active' at time } t, \\ 0 & \text{else.} \end{cases} \\ c_i(t) &= \text{content of } \texttt{count} \text{ at node } i \text{ at time } t. \end{aligned}$$

Hence $v_i(0) = x_i$ and $c_i(0) = 1$ for each node i. The initial value $\xi_i(0)$ (i.e. of `status`) depends on the particular algorithm: Under SRW $\xi_i(0) = 1$ for exactly one node, say node i_o, whereas under CRW $\xi_i(0) = 1$ for all nodes i.

Variables `value, status` evolve according to the same rules under both algorithms: Namely, each node has an independent Poisson clock that ticks at unit rate. When the local clock of a node ticks, the node does not take any action unless it is active at that time. If the node is active, then it chooses a neighbor at random, sends its current `value` and `count` to that neighbor, and becomes

inactive. The Send() subroutine of the algorithm maintains `value` and carries the identity e. The `count` is 0 at each inactive node at all times. If the selected neighbor was inactive at the time of reception, then it simply adopts the variables of the sender and it activates itself. Otherwise, the node executes a step towards computation of F_n and adds the received `count` value to its own.

For each time t define $\xi(t) \triangleq (\xi_1(t), \xi_2(t), \cdots, \xi_n(t))$. The process $(\xi_1(t), \xi_2(t), \cdots, \xi_n(t))$ of activity indicators is Markovian due to the randomness of the choice of neighbor by active nodes: For SRW the unique 1 in $(\xi_1(t), \xi_2(t), \cdots, \xi_n(t))$ follows a simple random walk on the communication graph. For CRW $(\xi_1(t), \xi_2(t), \cdots, \xi_n(t))$ indicates sites occupied by n simple random walks each of which evolves independently until it meets another and makes identically the same transitions with that walk afterwards.

3.1 Algorithm Correctness

We first establish correctness of the introduced algorithms by showing that each algorithm computes the quantity $F_n(x_1, x_2, \cdots, x_n)$ in finite time.

Lemma 3.1 *Under both SRW and CRW, and for all $t > 0$,*

$$F_n(v_1(t), v_2(t), \cdots, v_n(t)) = F_n(x_1, x_2, \cdots, x_n).$$

Proof. We prove the lemma by induction. Let $t_0 = 0$ and let t_k be the time of kth message passing in the system. The claim is true at time t_0 due to the initialization of each algorithm. Since the system state remains constant in the interval (t_k, t_{k+1}), it is enough to show the claim holds at time t_{k+1} provided that it holds at time t_k. To this end suppose the claim holds at time t_k. Without loss of generality, suppose that the $k+1$st message has sender 1 and receiver 2. Then $v_1(t_{k+1}) = e,\ v_2(t_{k+1}) = f(v_1(t_k), v_2(t_k))$ and $v_l(t_{k+1}) = v_l(t_k)$ for $3 \le l \le n$. Therefore

$$\begin{aligned} F_n(v_1(t_{k+1}), v_2(t_{k+1}), \cdots, v_n(t_{k+1})) &= F_n(e, f(v_1(t_k), v_2(t_k)), v_3(t_k), \cdots, v_n(t_k)) \\ &= f(\ F_2(e, f(v_1(t_k), v_2(t_k)))\ ,\ F_{n-2}(v_3(t_k), \cdots, v_n(t_k))\ ) \\ &= f(\ F_2(v_1(t_k), v_2(t_k))\ ,\ F_{n-2}(v_3(t_k), \cdots, v_n(t_k))\ ) \\ &= F_n(v_1(t_k), v_2(t_k), \cdots, v_n(t_k)), \end{aligned}$$

where the second and fourth equalities are due to (1) and the third equality is due to (2). This establishes the induction step and in turn the desired conclusion. □

We first consider SRW and define

$$\tau_S = \inf\{t :\ \text{each node becomes active at least once by time } t\ \}.$$

Let $a(t)$ denote the node that is active at time $t \ge \tau_S$. Since each node has been active at least once by time t, every node other than $a(t)$ should have turned inactive by sending the token to a neighbor. Therefore at each node $i \ne a(t)$ the value is $v_i(t) = e$. From Equations (1)–(2) we get

$$F_n(v_1(t), \cdots, v_n(t)) \ =\ F_2(v_{i(t)}(t), e) \ =\ v_{i(t)}(t).$$

Therefore by Lemma 3.1 the value $v_{a(t)}(t)$ of node $a(t)$ is equal to $F_n(x_1, \cdots, x_n)$.

The above argument applies verbatim to CRW also, by taking $a(t)$ to be the unique active node at time $t \geq \tau_C$ where τ_C is defined as

$$\tau_C = \inf\{t : \sum_i \xi_i(t) = 1\}.$$

It should be clear that for SRW τ_S is the cover time of the communication graph G and it is finite with probability 1. Under CRW, $\sum_i \xi_i(t)$ is a non-increasing process with an absorption state at 1. In this case τ_C is the absorption time of this process and it is almost surely finite. Hence under each algorithm the desired quantity $F_n(x_1, \cdots, x_n)$ is available at some node within finite time.

It remains to establish how the termination times τ_S, τ_C can be recognized. Towards this end note that the update mechanism for variable `count` reflects a secondary distributed computation procedure with $f(c_i, c_j) = c_i + c_j$ and

$$F_n(c_1(0), \cdots, c_n(0)) \;=\; \sum_i c_i(0) \;=\; n.$$

The general conclusions obtained above are valid in this special case, and in turn $c_{i(t)}(t) = n$ for $t \geq \tau_S$ under SRW and for $t \geq \tau_C$ under CRW. Therefore at such time instants node i can identify itself as the unique bearer of the desired quantity by verifying the condition $c_i(t) = n$. In other words variable `count` keeps an account of how many sensor values have been fused so far to form the content of variable `value`; this serves as a pilot signal with a known terminal value that signals the end of each algorithm. We collect these observations in the following theorem:

Theorem 3.1 *Both SRW and CRW compute the exact value of $F_n(x_1, \cdots, x_n)$ in finite time. Each algorithm terminates with the correct value when the content of variable* `count` *reaches n, the system size, at some node in the system.*

3.2 Time and Message Complexities

We present execution time and message complexity for SRW and CRW. We begin this section with the definitions of message and time complexities.

Definition 3.1 Average time complexity *of SRW (resp. CRW) refers to $E[\tau_S]$ (resp. $E[\tau_C]$).*

In adopting a measure of messaging complexity, let $\eta_S(t)$ and $\eta_C(t)$ be the total number of transmitted messages in the network by time t under algorithms SRW and CRW respectively:

Definition 3.2 Average per-node message complexity *of SRW (resp. CRW) refers to $n^{-1}E[\eta_S(\tau_S)]$ (resp. $n^{-1}E[\eta_C(\tau_C)]$).*

Note that $(\xi(t) : t \geq 0)$ is a Markov process under both algorithms. More precisely, $(\xi(t) : t \geq 0)$ is a random walk on G under SRW, and a coalescing random walk on G under CRW. In particular

the average time complexity of SRW is the mean cover time of G. Average message complexities of the algorithms are characterized by the following lemma:

Lemma 3.2

$$\begin{aligned} E[\eta_S(\tau_S)] &= E\left[\int_0^{\tau_S} \sum_i \xi_i(t)dt\right] = E[\tau_S], \\ E[\eta_C(\tau_C)] &= E\left[\int_0^{\tau_C} \sum_i \xi_i(t)dt\right]. \end{aligned}$$

Proof. We provide a proof that applies verbatim to both algorithms. Let $\eta(t)$ represent $\eta_S(t)$ for SRW and $\eta_C(t)$ for CRW. Let $\mathcal{F}_t$ denote the sigma-field generated by $(\xi(s) : s \leq t)$ and let $(\phi(t) : t \geq 0)$ be a Poisson process with unit rate. Note that $\eta(t)$ has the same distribution as $\phi(\int_0^t \sum_i \xi_i(s)ds)$ since each carrier node transmits messages at unit rate and inactive nodes do not engage in message transmission, and thus $\sum_i \xi_i(t)$ is the instantaneous rate of message generation in the network at time t. In particular the process $(\mu(t) : t \geq 0)$ with

$$\mu(t) = \eta(t) - \int_0^t \sum_i \xi_i(s)ds \tag{3}$$

is a martingale adapted to $\{\mathcal{F}_t\}$. Both τ_S and τ_C are $\{\mathcal{F}_t\}$-stopping times and they are almost surely finite. In addition $\sup_{t\geq 0} \sum_i \xi_i(t) \leq n$; hence it follows by the optional sampling theorem [12, Theorem 2.2.13] that $E[\mu(\tau_S)] = E[\mu(\tau_C)] = 0$. Using this observation in equality (3) establishes the lemma. □

4 Regular Topologies

Since the two algorithms are closely related to random walks, their time and message complexities for special topologies of the communication graph G can be deduced by referring to related work in applied probability. Before giving an analysis for general topologies in the next section, we consider here the cases when G is completely connected (i.e. clique) and d-dimensional torus for $d \geq 1$. Recall that average time complexity of SRW is the mean cover time of G and by the above lemma this is proportionally related to the mean message complexity. We refer the reader to [2, Chapter 5] for the cover time results, which are summarized in Table 1 in the column SRW. We articulate on the time/message complexity of CRW in more detail, as that entails consideration of coalescing random walks, which are relatively obscure in engineering applications.

Completely connected graph: A completely connected graph is a graph where each vertex has an edge with every other vertex. In such graphs the process $(\sum_i \xi_i(t) \mid t > 0)$ is also Markovian.

This follows from the fact that the transition matrix at any time instant is invariant to permutation. For CRW this process has initial state $\sum_i \xi_i(0) = n$ and at time $t < \tau_C$ it decreases by one at instantaneous rate

$$\begin{pmatrix} \sum_i \xi_i(t) \\ 2 \end{pmatrix},$$

that is, the number of edges that are incident on two actives nodes at time t. This process has been studied in detail in [23] and the results therein are summarized in Table 1. While a completely connected graph reflects a limited set of cases of practical importance, its analysis interestingly sheds considerable light on mesh-type topologies that are considered next.

Ring and d-dimensional torus: A d-dimensional torus is a graph where all the vertices have exactly $2d$ neighbors and it can be formed by joining the facing boundaries of a grid hence yielding a completely symmetric structure. A ring is simply a 1-dimensional torus. Consider $n = N^d$ for a d-dimensional torus. Let

$$s_N \;=\; \begin{cases} N^2 & \text{if } d = 1 \\ N^2 \log N & \text{if } d = 2 \\ N^d & \text{if } d \geq 3. \end{cases}$$

An asymptotic analysis of coalescing random walks on d-dimensional torus for large N is given by Cox [8]. It is established that the time-scaled the process $\sum_i \xi_i(s_N t)$ on a d-dimensional torus converges in distribution to $\sum_i \xi_i(t)$ on a completely connected graph as $N \to \infty$. (Note that $\sum_i \xi_i(t)$ is not even Markovian unless G is completely connected.) This result in turn leads to the following theorem on the time complexity for CRW:

Theorem 4.1 *[8, Theorem 6]*

$$0 \;<\; \liminf_{N\to\infty} E[\tau_C]/s_N \;=\; \limsup_{N\to\infty} E[\tau_C]/s_N \;<\; \infty.$$

The average message complexity of CRW can also be quantified by building on the asymptotic characterization of [8]. Towards that end consider a typical sample path of the process $(\sum_i \xi_i(t) \mid t > 0)$ illustrated in Figure 3. By Lemma 3.2 the average *aggregate* message complexity of the algorithm is the mean of the shaded area under the trajectory of $\sum_i \xi_i(t) : 0 < t \leq \tau_C$. The average per-node message complexity is then this quantity divided by the total number of nodes n.

Let

$$m_N = \begin{cases} N^2 & \text{if } d = 1 \\ N^2(\log N)^2 & \text{if } d = 2 \\ N^d \log N & \text{if } d \geq 3. \end{cases}$$

The following theorem provides an upper bound for the message complexity of CRW.

Theorem 4.2 *[21, Theorem 7]*

$$\limsup_{N\to\infty} E[\eta_C(\tau_C)]/m_N \;<\; \infty.$$

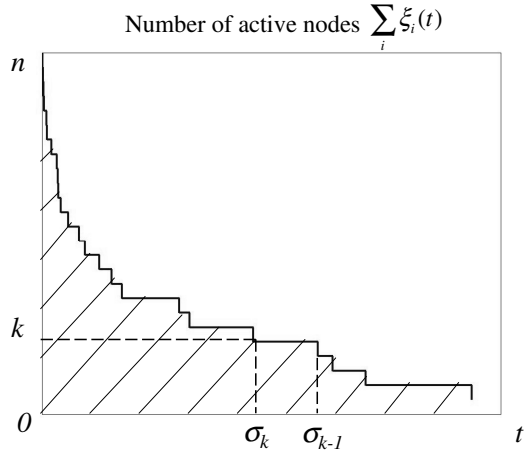


Figure 3: A sample path of the number of active nodes under algorithm CRW. σ_k denotes the first time that k active nodes remain in the network so that $\tau_C = \sigma_1$. The mean of the shaded area is the mean aggregate number of transmitted messages in the network.

To complement Theorem 4.2 a lower on the growth rate of $E[\eta_C(\tau_C)]$ is provided by Theorem 4.1: Since $\sum_i \xi_i(t) \geq 1$ for all t it follows that $E[\eta_C(\tau_C)] \geq E[\tau_C]$; in turn

$$\liminf_{N\to\infty} E[\eta_C(\tau_C)]/s_N \;\geq\; \liminf_{N\to\infty} E[\tau_C]/s_N \;>\; 0.$$

By comparing s_N and m_N we conclude that this bound is tight for a ring, i.e. a 1-d torus, and it is off by at most a factor $\log N$ for $d \geq 2$. For a relatively concrete view of these quantities Figure 4 provides a summary of numerical simulations for time and message complexities of CRW on 2-dimensional tori of varying sizes.

5 Two-Phase Algorithms

The algorithm would work as follows: In the first phase the nodes in the network follow a CRW protocol upto some designated deterministic time t. At this time there are $\eta_C(t)$ tokens left in the system. The set of nodes that have tokens at time time t then *flood* their messages to all the nodes as described below.

Controlled Flooding (CFLD) algorithm assumes no network topology. It works by forwarding the message over all links. From the source node message is sent to all neighbors. Each node, v, receiving its first message from vertex u sends messages to all neighbors except u. Also, each node will transmit its packet at most once [6]. If no message is received then it does nothing. It is well known that the message complexity scales as $\Theta(|edges|)$ since each edge delivers the message either once or twice [6]. The time complexity scales as $\Theta(diam(G))$ since we must reach all nodes [14]. We can adapt CFLD algorithm to our scenario where we have a random number $\eta_C(t)$ tokens left in the system. At time t all nodes cease to implement CRW. Nodes with tokens separately CFLD messages. Each node then fuses the data once all the messages are received.

There are two fundamental questions that arise:
(1) How do nodes realize that they have all the messages;
(2)Can we ensure that consensus is achieved through this process.
The answer to the second question lies in Lemma 3.1, which asserts that the data fusion is invariant to order of reception. Consequently, we are left to address the first requirement. Here we invoke the secondary distributed computation scheme described in Figure 1 and Section 3. Each active node at time t has its individual variable `count`. During the CFLD phase each node forwards this count variable in addition to its fused value. Each node can then determine whether it has received all the messages by updating its private `count` variable.

There are three principal advantages for combining CRW and CFLD:
(**A**) Consensus is obtained in finite time.
(**B**) CRW slows down when there are few tokens increasing time complexity. The two phase algorithm substantially improves time complexity.
(**C**) Analytical bounds for message and time complexities for general graphs can be established. This is because it is easier to determine the expected number of tokens left in the system at any time.

Next we will present message and time complexity for the combined algorithm. Recall the n-node communication graph $G = (V, E)$ with link set E and nodes V; Let d_v denote the degree of node $v \in V$; and $\text{diam}(G)$ denote the diameter of the graph.

To simplify the exposition we denote,

$$N(t) = E\left[\sum_{i=1}^{n} \xi_i(t)\right] \tag{4}$$

Note that $1 \leq N(t) \leq n$ and $N(0) = n$, and $\xi_i(t)$ is as before the state of node i at time t.

We compute the time it takes for the expected number of active tokens to be below some positive integer γ. This leads to the following definition.

Definition 5.1 *γ-time complexity is the time $\mathbf{T}_\gamma$ it takes for the CRW on an n-node graph to have an average of γ active tokens left in the system, i.e.,*

$$\mathbf{T}_\gamma = \min\left\{t \geq 0 \mid N(t) \leq \gamma\right\},\ \gamma = 2, 3, \ldots, n$$

Observe that unlike the termination time, τ_C defined in the Section 3.2, $\mathbf{T}_\gamma$ is no longer a random variable, which as we will see in Section 6 will simplify our analysis.

Define message complexity for the CRW until time t:

$$M_C(t) = \int_0^t N(s)ds. \tag{5}$$

The time complexity, $\mathbf{T}$, is the sum of the time complexities corresponding to the two phases. Consequently, for the case when all the tokens during the CFLD phase are transmitted at a unit

rate we have:

$$\mathbf{T} \leq \mathbf{T}_\gamma + O(\text{diam}(G)) \tag{6}$$

Theorem 5.1 *The overall message complexity, $M(t)$ for the two phase scheme where nodes follow CRW upto time t and CFLD after time t is less than $M(t) + 2(\sum_{v\in V} d_v)N(t)$. For $t = \mathbf{T}_\gamma$, the overall message complexity is less than $M_C(\mathbf{T}_\gamma) + 2(\sum_{v\in V} d_v)N(\mathbf{T}_\gamma)$.*

Proof. Suppose, the CFLD phase starts at time t then the overall number of messages, $\eta(t)$ is the sum of messages transmitted during the CRW phase, $\eta_C(t)$, and that transmitted during the CFLD phase, $\eta_F(t)$ starting at time t. Specifically, we have $\eta(t) = \eta_C(t) + \eta_F(t)$. Taking expectations on both sides we obtain: $E(\eta(t)) = M(t) + E(\eta_F(t))$. We can simplify the expression for the second phase by noting that(see [6]), $E(\eta_F(t)) \leq 2(\sum_{v\in V} d_v)N(t)$. The proof now follows by substitution. □

6 Time and Message Complexities for General Graphs

This section describes techniques for estimating the message and time complexity of CRW for general graphs in both continuous and discrete time settings. Bounds for SRW reduces to computation of cover times. Cover time bounds for many of the graphs considered in this paper are available in the literature and we do not develop these results here.

To develop results for CRW we will follow the two-phase procedure outlined in the previous section. The advantages of the two-phase algorithm has already been outlined in Section 5. We recall one main advantage that is pertinent here, namely, from Theorem 5.1 it follows that we do not have to seek bounds for the stopping times τ_C. Rather we only need to determine the expected number of active tokens at a deterministic time t.

This section is organized as follows. First we establish straightforward results for general graphs based on bounds on the worst-case mean hitting time. We show that the number of active tokens at time t decays as $O(n\exp(-\frac{t}{\sigma}))$, where σ is the worst-case mean hitting time on the graph. We compute σ network-circuit resistance analogy. We then compute complexity bounds for a number of graphs such as expanders and meshes. While this bound is general, it turns out to be conservative in estimating the message complexity. The main reason is that the worst-case mean hitting time is generally large for many graphs and a local analysis is required. This motivates a careful study of message complexity based on local analysis of random walks. Specifically, we consider graphs with geometric structure. We show that for such graphs the message complexity scales as $O(\log^2(n))$ paralleling our results for the torus in Section 4.

As before let X_t denote the state of a random walk on a graph $G = (V, E)$. For continuous time we consider unit rate random walks and assume that multiple random walks are independent. Analogously we also consider discrete time independent simple symmetric random walks and allow self-loops in the graphs. Our definitions and results typically apply to both continuous and

discrete setups based on the so called jump-and-hold description[1] and a continuization argument as described in [2]. Nevertheless, wherever appropriate we will point out specifically whether our results apply to discrete or continuous time scenarios.

Analogous to Equation 5 for the continuous time setup, the message complexity in the discrete setup is given by,

$$M_C(t) = \sum_{s=0}^{t} N(s) \tag{7}$$

We denote the first hitting time of node v by T_v, i.e., $T_v = \inf\{t \geq 0 \mid X_t = v\}$. We also denote by T_{vw} the hitting time for a random walk starting at v and hitting w for the first time.

$$T_{vw} = \inf\{t \geq 0 \mid X_0 = v,\ X_t = w\}$$

The worst-case hitting time is denoted as σ, i.e.,

$$\sigma = \max_{v,w \in V} E(T_{vw})$$

Let C_{vw} be the first time that two independent unit rate continuous time random walks, X_t, Y_t, on graph $G = (V, E)$ started at nodes v and w coalesce(meet), i.e.,

$$C_{vw} = \inf\{t \geq 0 \mid X_t = Y_t, , X_0 = v, Y_0 = w\}$$

The meeting times for two independent copies of random walks in continuous time is related to the worst-case hitting time. Specifically, Aldous [2] (Proposition 5, Chap 14) uses Martingale arguments to show that,

$$\max_{v,w} E(C_{vw}) \leq \sigma \tag{8}$$

Let $\alpha_s(A)$ denote the worst-case coalescing time probability on a subset $A \subset V$, i.e.,

$$\alpha_s(A) = \min_{v,w \in A} Prob(C_{vw} \leq s) \tag{9}$$

Note that by union bounding we obtain, $\alpha_s(A) \geq \alpha_s(V)$. Now through Markov inequality together with Equation 8 we get a bound on the probability of meeting time, i.e.,

$$\alpha_s(A) \geq \alpha_s(V) \geq 1 - \frac{\sigma}{s} \tag{10}$$

[1]Specifically, as described in [2] the continuous walk can be constructed by the two step procedure, namely, (1) Run a discrete time chain with the simple symmetric transition matrix; (2) Given the sequence of states, $v_j \in V,\ j = 1, 2, \ldots, m$ visited by the discrete time chain, the duration of time spent at each state, v_m is a unit rate exponentially distributed random variable. This continuization is particularly useful since useful quantities such as mean hitting times etc. in the continuous case corresponds directly to the mean number of discrete time steps required in the discrete time chain.

We next consider decomposition of the original graph into disjoint subgraphs and bound the total coalescing time by the union of the coalescing times for the subgraphs. Let $\lfloor t \rfloor$ denote the greatest integer smaller than t. Suppose $A_i, i = 1, 2, \ldots, m(t)$ be a partition of the vertices of the graph and $\mathcal{A}_t$ denotes the collection, i.e.,

$$\bigcup_{i=1}^{m(t)} A_i = V, \;\; A_i \bigcap A_j = \emptyset, \; i \neq j, \; \mathcal{A}_t = \{A_1, A_2, \ldots, A_{m(t)}\}$$

The worst-case coalescing time, $\alpha_s(\mathcal{A}_t)$, over this sub-collection is defined by

$$\alpha_s(\mathcal{A}_t) = \min_{1 \leq j \leq m(t)} \alpha_s(A_j) \tag{11}$$

Theorem 6.1 *Consider the partition of the graph into subsets, $\{A_k\}$, as described above. Suppose $1 \leq m(t) \leq \frac{N(t)}{2}$ i.e., the number of partitions is smaller than one half the expected number of active tokens at time t. It follows for both continuous and discrete time setups that,*

$$N(t+s) \leq N(t) \exp\left(-\frac{1}{2}\alpha_s(\mathcal{A}_t)\right); \quad 0 \leq s \leq t, \quad N(t) \geq 2. \tag{12}$$

Furthermore, suppose $t \leq r \leq r+s \leq 2t$ and the number of partitions are chosen such that $1 \leq m(t) \leq \frac{N(t)}{4}$ and $N(t) \leq 2N(2t)$, then it follows that,

$$N(2t) \leq N(t) \exp\left(-\left\lfloor \frac{t}{2s} \right\rfloor \alpha_s(\mathcal{A}_t)\right); \quad 0 \leq s \leq t, \quad N(t) \geq 2. \tag{13}$$

The proof of the theorem appears in the appendix and is based on the arguments presented in Cox [8] for the torus. We exploit the salient steps there to extend it to general graphs.

Observe that if the coalescing time of two walks is a constant then the number of active tokens decreases exponentially fast. However, the meeting time can be large, namely, the probability that two walks meet in a short time can be very small. Note that since $0 \leq \alpha_s(\mathcal{A}_t) \leq 1$ the right hand sides of Equation 12 is larger than $N(t)/\sqrt{e}$. Consequently, Theorem 6.1 is not useful for large incremental times s. Therefore, this result will be used as an intermediate step in an iterative process over many increments to provide useful bounds.

We will now use Theorem 6.1 to prove the γ time and message complexities for arbitrary connected graphs. We have the following theorem.

Theorem 6.2 *Consider the algorithm CRW on an arbitrary connected graph, $G = (V, E)$. The γ time complexity for $\gamma \geq 2$ scales*[2] *as $O(\sigma \log(n/\gamma))$. The γ message complexity for $\gamma \geq 2$ scales as $O(n\sigma \log(n/\gamma))$.*

Proof. In Theorem 6.1 we choose a single partition, i.e., $\mathcal{A}_t = \{V\}$. For this case we note that the worst-case meeting time $\alpha_s(\mathcal{A}_t) = \alpha_s(V)$. Consequently, we can apply Markov inequality described

[2]The $O(\cdot)$ notation here and in the rest of this section for time and message complexity implies that the bound holds for sufficiently large time for a fixed n-node graph.

by Equation 10 to obtain $\alpha_s(V) \geq 1 - \sigma/s$. This bound only makes sense if $s > \sigma$. We choose time increments $s = 2\sigma$ and partition $T = 4\sigma \log(n/\gamma)$ into $2\log(n/\gamma)$ increments. For each increment s we obtain from Equation 13 that,

$$N(t+s) \leq N(t)\exp(-(1-\sigma/s)) = N(t)\exp(-1/2)$$

Repeating this $2\log(n/\gamma)$ times we get

$$N(T) \leq N(0)(\exp(-1/2))^{2\log(n/\gamma)} = \gamma$$

where the last equality follows from the fact that $N(0) = n$. The γ message complexity directly follows from Equation 5. □

In the next section we will now apply Theorem 6.2 for specific graphs to obtain bounds on time and message complexities.

6.1 Time and Message Complexity Based on Hitting Time Characterization

Our goal in this section is to use well known bounds on hitting times for some well known graphs together with Theorem 6.2.

For general graphs Aleliunas et al [3] showed a general upper bound $\sigma = O(|E||V|)$, for the worst-case hitting time, where $|E|$ is the number of edges and $|V|$ is the number of nodes (vertices). If the maximal degree of the graph is $D_{\max}$ then $|E| \leq nD_{\max}$ and $|V| = n$. This implies that the γ time complexity scales as

$$\mathbf{T}_\gamma = O\left(n^2 D_{\max}\log(n/\gamma)\right)$$

We note that this result is generally conservative in comparison to the time complexity of 2D torus described in the previous sections. This is because this hitting time bound is conservative. We invoke resistance characterization of hitting time to obtain sharper bounds.

6.1.1 Resistance characterization for Connected graphs

Chandra et al[9] establish bounds for hitting time between any two nodes based on resistance of electrical networks. Note that the resistance bounds apply generally to discrete time walks. However, note that there is a close relationship between the discrete and continuous time random walks based on the so called jump-and-hold description described earlier, which results in similar results for continuous time with appropriate time scaling.

The electrical network is obtained by replacing each edge in the graph with a one-ohm resistor. It turns out that the worst-case mean hitting time satisfies

$$\sigma \leq \max_{u,v\in V} 2|E|\rho_{uv} \stackrel{\Delta}{=} \rho^* \tag{14}$$

where ρ_{uv} is the effective resistance between nodes u and v. Consequently, if $D_{\max}$ is the maximum degree for the graph and ρ^* is the maximum effective resistance between any two nodes in the network, we get

$$\mathbf{T}_\gamma \leq 2n \log(n/\gamma) D_{\max} \rho^* \tag{15}$$

Expander Graphs: An $(n, D_{\max}, \alpha)$ expander is a graph $G = (V, E)$ on n vertices of maximal degree $D_{\max}$ such that every subset $A \subset V$ satisfying $|A| \leq n/2$ has $|N(A) - A| \geq \alpha |A|$, where

$$N(A) = \{v \in V \mid (u, v) \in E,\, u \in A\} \tag{16}$$

For an $(n, D_{\max}, \alpha)$ expander graph with minimum degree $D_{\min}$, the worst-case resistance is equal to

$$\rho^* = \frac{24}{\alpha^2 (D_{\min} + 1)}.$$

Consequently, the γ time complexity scales as:

$$\mathbf{T}_\gamma \leq 48 \frac{n \log(n/\gamma) D_{\max}}{\alpha^2 (D_{\min} + 1)},\ \gamma \geq 2$$

For an expander graph, where $D_{\min} \approx D_{\max}$ we get a γ time complexity scaling as

$$\mathbf{T}_\gamma = O(n \log(n/\gamma))$$

We note that this result is close to the time complexity bounds obtained for a completely connected network in the previous section.

2D Mesh: From the resistance calculations it turns out that

$$\rho^* = O(\log(n))$$

for 2D mesh [9] with n nodes. Consequently, for the 2D mesh we obtain

$$\mathbf{T}_\gamma = O(n \log^2(n/\gamma)),\ \gamma \geq 2$$

This is within a $\log(n)$ factor of the bound obtained for the 2D torus using more elaborate martingale calculations in the previous section. Note that unlike the 2D torus the 2D mesh is not symmetric and results of the previous section cannot be directly applied here.

Random Geometric Graphs (RGG): A 2D Random Geometric Graph with n nodes and radius $r(n)$, denoted by $G_{r(n)} = (V, E)$, is a graph where nodes are uniformly distributed in the unit square and $(u, v) \in E$ if and only if the Euclidian distance between nodes u and v is smaller than or equal to $r(n)$.

It is well known that when the radius of connectivity is chosen as $r(n) = \sqrt{2 \log n / n}$, the graph is connected with high probability. Furthermore, Avin and Ercal [4] (Theorem 5.3) show that, with high probability, the resistance scales as

$$\rho^* = O(1/nr^2(n))$$

and the number of edges scales as $|E| = O(n^2 r^2(n))$ (see [4] Corollary 3.5) for this choice of connectivity radius. Consequently, the worst-case mean hitting time scales as $\sigma = O(n)$ with high probability. This implies that for geometric random graphs with $r(n) = \sqrt{2 \log n/n}$ the γ time complexity scales as

$$\mathbf{T}_\gamma = O(n \log(n/\gamma)),\ \gamma \geq 2.$$

While the time-complexity bounds obtained using resistance characterization appears to be tight for several cases, the γ-message complexity is overly conservative. This is because the worst-case mean hitting time, σ is $\Omega(n)$ in general. Theorem 6.2 implies that the γ message complexity scales as $O(n^2 \log(n))$ even for a 2D torus. This is significantly weaker than the complexity bounds obtained for the torus in Section 3.2. Motivated by these reasons we develop a new characterization of message and time complexities based on local geometric analysis of random walks.

6.2 Logarithmic Bounds for Message Complexity

The main conservatism in Theorem 6.2 arises from the fact that the meeting time is bounded in terms of the worst-case hitting time. Specifically, if two random walks start relatively close to each other we expect that the meeting time is relatively small, i.e., the meeting time should typically scale with initial distance between the two walks. In this section we develop these ideas further for graphs that have a geometric neighborhood structure. We focus on discrete time walks since the analysis is technically simpler. Each active token follows an independent, simple, symmetric random walks on the graph $G = (V, E)$. Specifically, at each step an active token moves to a neighbor of its current location, chosen uniformly at random and the moves of all the active tokens are synchronized (this assumption is not restrictive since we allow self-loops).

The basic idea is based on local behavior of random walks. Specifically, it turns out that for graphs that are endowed with a geometric neighborhood structure it is possible to characterize the probability that two random walks meet in terms of their initial graph distance. We emphasize that while in general there is always a non-zero probability that two random walks meet, this probability has often been characterized in terms of the entire graph. Indeed this was the basic reason for the conservatism of resistance based bounds derived in the previous section. Therefore, to overcome this issue we will develop results based on local behavior of random walks. Our main result in this section (see Theorem 6.3) will establish that under certain regularity conditions on the graph the expected number of active tokens at time step t decays inversely with t, i.e.,

$$N(t) = O\left(\frac{n \log(t+1)}{t}\right),\ N(t) \geq 2 \tag{17}$$

We again emphasize that the $O(\cdot)$ notation above and in the rest of this section refers to time asymptotics for a fixed n-node graph. The result implies a bound on both γ time complexity and γ message complexity. The γ time complexity scales as

$$\mathbf{T}_\gamma \leq C \frac{n \log(n)}{\gamma},\ \gamma = 2, 3, \ldots, n$$

Note that the γ time complexity bounds are order-wise similar to those derived using resistance arguments in the previous section. However, the main advantage here is that we can now obtain a bound on message complexity based on Equations 7, 21:

$$M(\mathbf{T}_\gamma) \leq C \sum_{t=1}^{\mathbf{T}_\gamma} \frac{n \log(1+t)}{\gamma t} \leq C \frac{n}{\gamma} \left(\log^2(n) + o(\log(n)) \right) \tag{18}$$

Thus the message complexity per node scales as $n^{-1}\eta_\gamma = O(\log^2(n))$.

This result is based on the fact that for many graphs the probability that two walks at a distance R meet in time R^2 is bounded from below by the $1/\log(R)$. To precisely describe these ideas we introduce some notation. Let $d(u, v)$ be the graph distance between the nodes u, $v \in V$, i.e., the minimal number of edges in any edge path connecting u and v. We denote by $B(u, R)$ the ball centered at node u and radius R, i.e.,

$$B(u, R) = \{v \in V \mid d(u, v) < R\}$$

The volume of a set, $A \subset V$, denoted by $Vol(A)$, is the number of edges contained in the ball. The volume of the ball, $B(u, R)$ is denoted by $Vol(u, R)$ for simplicity. Note that if d_v is the degree of node v then we have,

$$Vol(u, R) = Vol(B(u, R)) = \sum_{v \in B(u,R)} d_v$$

Next we denote by $P(u, v)$ the 1-step transition probability of going from node u to node v. Since we consider simple symmetric random walks, this transition probability is the inverse of the degree of node u if u and v are connected and zero otherwise. We also use $P_t(u, v)$ to denote the t-step transition probability for going from u to v. We next present a precise characterization when Equation 17 holds. We will see that this bound holds when one has a geometric neighborhood structure as described below:

Definition 6.1 *A Graph $G = (V, E)$ is said to satisfy a geometric neighborhood structure if there exists constants, C_0, C_1 such that*

$$C_0 R^2 \leq |B(u, R)|; \;\; |B(u, R) \bigcap B(u, R + \Delta)| \leq C_1 \Delta R, \;\; \forall \; u \in V, \; 0 < \Delta \leq R. \tag{19}$$

where, $0 \leq R \leq R_{\max}$ and $R_{\max}$ is the diameter of the graph.

Typically a graph that is approximately regular and has a geometric neighborhood structure satisfies such a property. The geometric random graph described earlier asymptotically satisfies the geometric neighborhood property. Indeed, note that due to the uniform distribution of the nodes in the unit cube this property is satisfied for sufficiently large n with high probability (see [4] for more details). The theorem below will evidently require only the lower bound. However, it turns out that to ensure a suitable bound on the meeting time probability the upperbound will also be necessary.

Theorem 6.3 *Suppose the graph $G = (V, E)$ has a geometric neighborhood structure as described in Equation 19 and the meeting time probability satisfies:*

$$\alpha_{t^2}(B(u,t)) \geq \frac{C_2}{\log(t)},\ t > 0,\ u \in V \tag{20}$$

for some constant C_2 independent of time t. Note that $\alpha_{t^2}(B(u,t))$ is the meeting time probability (see Equation 9) for any two walks starting in the ball $B(u,t)$ in time t^2. Then the expected number of active tokens at time step t satisfies

$$N(t) \leq C\left(\frac{n\log(t+1)}{t}\right),\ t \geq 1 \tag{21}$$

where $C = \frac{16}{C_0}\max(8, \frac{\log(2)}{C_2})$ when the number of active tokens is greater than 4.

Note that smaller the constant C_0 the larger the number of active tokens at time t. We are now left to determine the conditions under which Equation 20 is satisfied. Surprisingly, it turns out that the logarithmic bound holds if:
(a) The t-step transition probability is approximately Gaussian.
(b) Geometric neighborhood property as described in Theorem 6.3 holds.
This result is stated below.

Lemma 6.1 *Consider the graph $G = (V, E)$ satisfying the geometric neighborhood property as described in Equation 19. Suppose the t-step transition probability satisfies the so called Gaussian bound, i.e.,*

$$\frac{C_3}{t}\exp\left(-\frac{d^2(u,v)}{C_4 t}\right) \leq P_t(u,v) + P_{t+1}(u,v);\ \forall\ u,\ v \in V,\ 1 \leq d(u,v) \leq t \tag{22}$$

where C_3, C_4, are positive constants independent of time. Then the probability of meeting time satisfies Equation 20 for some suitable constant C_2. Consequently, these conditions also imply the γ message complexity bound described by Equation 21.

Note that the Gaussian t-step transition estimate bounds the sum of the transitions at t and $t+1$. Note that for bi-partite graphs we must have either P_t or P_{t+1} equal to zero. Therefore, we cannot hope to improve this situation in general. However, if each node has self-loops it turns out that we can lower bound the t step transition probability directly, i.e., for non-bipartite graphs we have

$$\frac{C_3}{t}\exp\left(-\frac{d^2(u,v)}{C_4 t}\right) \leq P_t(u,v);\ \forall\ u,\ v \in V,\ 1 \leq d(u,v) \leq t \tag{23}$$

Our problem now reduces to finding those graphs that satisfy the t-step Gaussian transition property. It turns out that weak homogeneity conditions lead to the Gaussian t-step transition property. We describe what these conditions are next.

Volume Doubling Property: A graph $G = (V, E)$ is said to satisfy *volume doubling property* if volume of a ball centered at any point, u, with increasing radius satisfies

$$Vol(u, 2R) \leq C_5 Vol(u, R), \ \forall \ u \in V, \ R > 0 \tag{24}$$

We again point out that a 2D mesh satisfies such a property. The volume at a graph radius R and $2R$ is smaller than R^2 and $4R^2$ respectively for any R. A similar result holds for random geometric graphs(RGG) due to the so called geo-dense property [4]. For a constant $\mu \geq 1$, a graph is said to be μ-geo-dense if every square bin of size $A \geq r^2(n)/\mu$ (in the unit square) has nA nodes. Recall from Section 6.1.1 that any two nodes at a Euclidean distance $r(n)$ is connected. Lemma 3.2 of [4] shows that with high probability if $r^2(n) = c\mu \log(n)/n$ then RGG is μ geo-dense. Furthermore, if RGG is μ geo-dense then, (i) Each node, v, has degree $d_v = \Theta(nr^2(n))$; (ii) $|E| = \Theta(n^2 r^2(n))$. Consequently, we immediately see that the volume doubling property holds since RGG is evidently close to a 2D mesh in terms of volumes at the different radii except for a $\log(n)$ factor.

Constant Resistance Property: For any subsets $A \subset B \subset V$ consider an electrical network with one-ohm resistors for each edge on the graph $G = (V, E)$. Define the resistance, $\rho(A, B)$, between A and B as the power dissipated when a one-volt potential is applied to all the nodes in A and the nodes in the complement of B, i.e., B^c are all grounded. The graph $G = (V, E)$ is said to satisfy the constant resistance property if:

$$C_6 \leq \rho(B(u, R), B(u, MR)) \leq C_7 \tag{25}$$

where M is any number larger than one and C_6 and C_7 are constants that can depend on M but not on R.

Again consider first the example of a 2D mesh. Due to symmetry all the nodes at distance $R + \Delta$ have the same potential. Consequently we can short all the nodes at this distance. Due to the geometric neighborhood property, there are about $R + \Delta$ nodes connected to nodes at a distance $R + \Delta - 1$. Since this is a parallel set of resistances the effective resistance is $1/(R + \Delta)$. Summing over these resistances we obtain,

$$\rho(B(u, R), B(u, MR)) = \sum_{\Delta=1}^{MR} \frac{1}{R + \Delta} \approx \log(\frac{MR}{R}) = \log(M)$$

which establishes the fact. A similar but more elaborate argument is required for RGG. Basically the short cut principle along with the geo-dense property ensures a lower bound of $\Omega(\log(M))$. To obtain an upper bound we need to construct a flow along the lines of [4] that satisfies the Kirchoff current law.

Uniform Isoperimetry Property: We consider the subgraph, $G(u, R)$, formed by restricting the graph $G = (V, E)$ to the subset of vertices in the ball, $B(u, R)$. Consider any partition of

$G(u,R)$ into S, S^c. We say that the graph $G=(V,E)$ satisfies a uniform isoperimetry property if for every u and every R we have,

$$\frac{C_8}{R} \leq \frac{Cut(S,S^c)}{\min(Vol(S),Vol(S^c))} \tag{26}$$

where C_8 is some constant independent of R and $Cut(S,S^c) = \sum_{u\in S,\, v\in S^c} \mathbf{1}_{\mathbf{u},\mathbf{v}}$.

For the 2D mesh this is a well-known property (see [10]). The corresponding property for an RGG is a direct consequence of Theorem 4.1 of [4].

We are ready to state our result.

Lemma 6.2 *Consider a graph $G=(V,E)$ that is in general infinite and satisfies the properties described in Equations 24, 25, 26, then the t-step transition probability satisfies the Gaussian estimate described in Equation 22.*

Proof. The proof is a direct consequence of the results in Merkov [17] and Grigoryan and Telcs [13]. Theorem 3.1 in Grigoryan and Telcs [13] states that if a graph $G=(V,E)$ satisfies the volume doubling property, the resistance property and the Elliptic Harnack Inequality, the t-step transition matrix satisfies Equation 22. Merkov [17] shows that the isoperimetry property implies the Elliptic Harnack inequality. □

6.3 Message and Time Complexity for Achieving Consensus

We will utilize Theorem 5.1 to characterize message and time complexity for achieving consensus in general graphs. For general graphs we note from the resistance arguments of Equation 15 that,

$$\mathbf{T}_\gamma \leq 2n\log(n/\gamma)D_{\max}\rho^* \implies \mathbf{T} \leq \mathbf{T}_\gamma + O(diam(G)) \leq 2n\log(n/\gamma)D_{\max}\rho^* + O(n)$$

As we described earlier this bound is not useful for characterizing message complexity. To obtain better bounds we restrict our attention to graphs satisfying volume doubling, constant resistance and uniform isometry described in the previous section. We note that the message complexity from Theorem 5.1 can be bounded as:

$$M(\mathbf{T}_\gamma) \leq M_C(\mathbf{T}_\gamma) + 2\gamma\sum_{v\in V} d_v \leq O(\frac{n}{\gamma}\log^2(n)) + 2\gamma\sum_{v\in V} d_v$$

where, d_v is the degree of node v and we have used Equation 18 to determine a bound on $M_C(\mathbf{T}_\gamma)$. We now let $\gamma = \log(n)$. It follows that the message complexity for the two phase scheme is:

$$M(\mathbf{T}_\gamma) \leq O(n\log(n)) + 2\gamma\sum_{v\in V} d_v \implies M(\mathbf{T}_\gamma) \leq O(n\log(n))$$

where in the final inequality we have used the fact that $d_v \leq 4$ for two-dimensional Grid graphs. The time complexity for grid graphs follows from Equation 6,

$$\mathbf{T} \leq \mathbf{T}_\gamma + O(\text{diam}(G)) \leq O(n)$$

where we note that $\mathbf{T}_\gamma$ for $\gamma = O(\log(n))$ scales as $O(n)$ and $O(\text{diam}(G))$ scales as $O(\sqrt{n})$.

For RGG we note that with high probability the number of links is of $O(\log(n))$. Consequently, following along the same lines as the previous computation for 2D grid graphs we obtain

$$M(\mathbf{T}_\gamma) \leq O(n \log^2(n))$$

By noting that for RGG $\text{diam}(G) = O(\sqrt{n/\log(n)})$ we get a bound on the time complexity:

$$\mathbf{T} \leq O(n \log(n/\gamma)) + O(\text{diam}(G)) \leq O(n)$$

6.4 Numerical Results

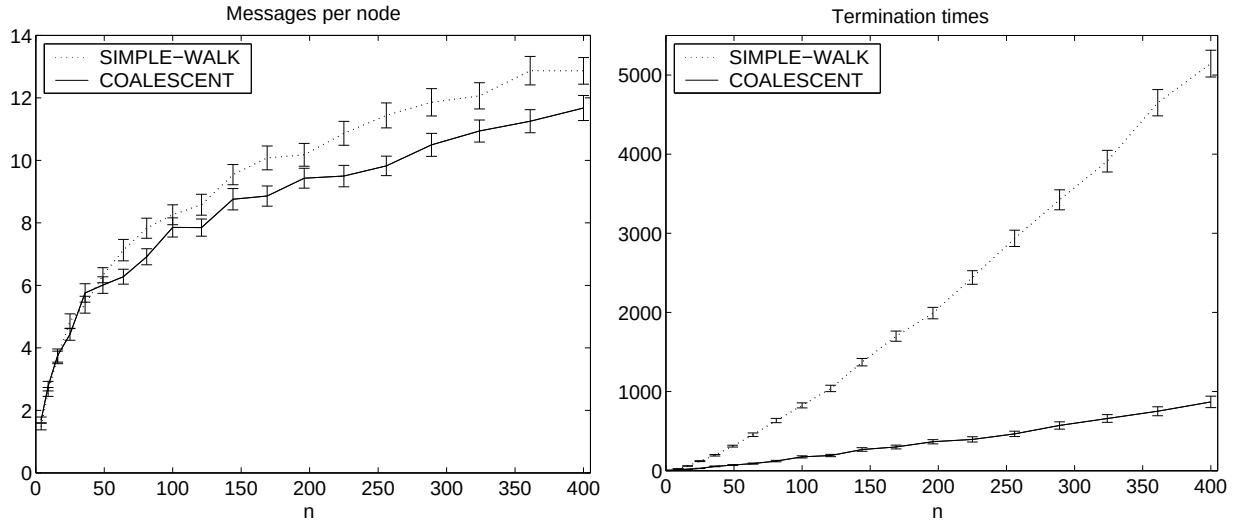
Numerical verification of the analytical results of Table 1 on a $2-d$ torus is presented in Fig. 4. Figure 4 provides a summary of numerical simulations for time and message complexities of CRW on 2-dimensional tori of varying sizes.

An important consideration is that GOSSIP achieves consensus at all nodes while SRW and CRW realize their solution at a random node. Therefore, strictly speaking for the comparisons to be meaningful we need to add the time and message complexities to obtain similar consensus estimates for SRW and CRW. We can obtained consensus through CFLD. The time complexity of CFLD for torii scales as $O(\sqrt{n})$, which is insignificant relative to time complexity of SRW/CRW. Message complexity-per-node of CFLD on torii scales as $O(\log(n))$, which is again insignificant relative to message complexity of CRW $O(\log^2(n))$. Consequently, the qualitative nature of the plots is similar even when we incorporate these additional costs.

We also simulate numerically time and message complexity for random geometric graphs. To simulate a 2 dimensional geometric random graph we distributed n nodes in a unit square and formed edges whenever two nodes were at a distance smaller than $\sqrt{2 \log n/n}$. We discarded graphs that were not connected. Again to compare CRW/SRW against GOSSIP consensus costs must be incorporated. We can obtained consensus through CFLD. The time complexity of CFLD for RGG scales as $O(\sqrt{n/\log(n)})$, which is insignificant relative to time complexity of SRW/CRW. Message complexity-per-node of CFLD on RGG scales as $O(\log(n))$, which is again insignificant relative to message complexity of CRW $O(\log^2(n))$. Consequently, the qualitative nature of the plots is similar even when we incorporate these additional costs.

We next describe Gossip algorithm as studied in [7] for the sake of completion. Gossip algorithms refer to distributed randomized algorithms that are based on pairwise relaxations between randomly chosen node pairs. In the present context a pairwise relaxation refers to averaging of two values available at distinct nodes. In what follows a stochastic matrix $P = [P_{ij}]_{n \times n}$ is called *admissible* for G if $P_{ij} = 0$ unless nodes i and j are neighbors in G. The algorithm is parameterized by such a P:

Algorithm `GOSSIP-AVE`(P): Each node i maintains a real valued variable with initial value $z_i(0) = x_i$. At the tick of a local Poisson clock, say at time t_o, node i chooses a neighbor j with respect to



(a) Per-node message complexities

(b) Time complexities

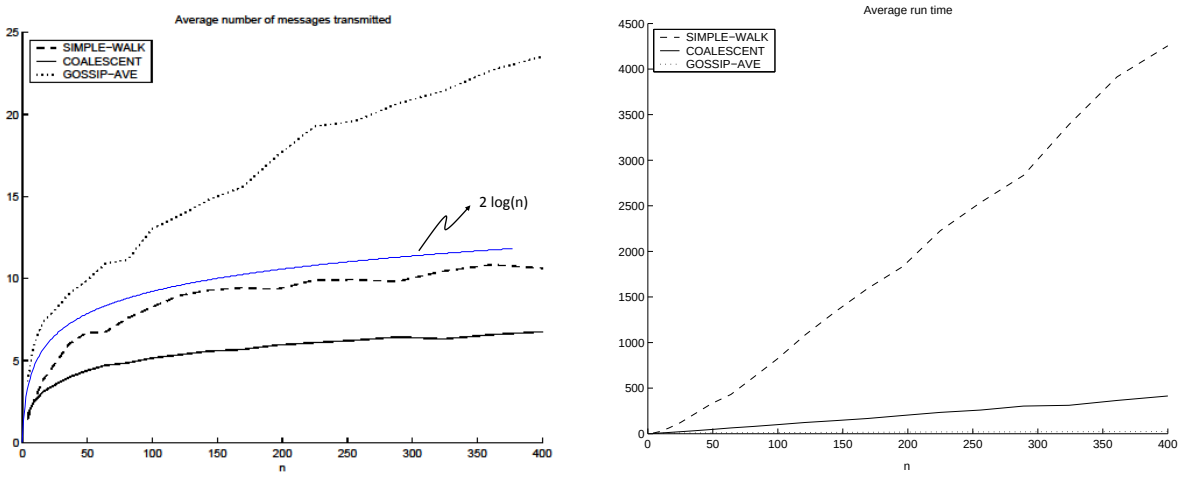
Figure 4: Average execution times and message complexities per node for CRW and GOSSIP on the 2-dimensional torus with n nodes. Note for an accurate comparison time and message complexity of CFLD needs to be added to that of CRW. Nevertheless, both time and message complexity are insignificant relative to that of CRW (see Section 6.3) for the torus.

the distribution $(P_{ij} : j = 1, 2, \cdots, n)$ and both nodes update their internal variables as $z_i(t_o) = z_j(t_o) = (z_i(t_o^-) + z_j(t_o^-))/2$. We associate each node with a real value and consider the problem of computing its mean value. In order to make a fair comparison of `GOSSIP-AVE` with `CRW` and `SRW` we need to use a stopping criterion for `GOSSIP-AVE`. Let $\bar{x}$ denote the average of $x_1, x_2, \cdots, x_n$, let $z(t)$ denote the vector $(z_1(t), z_2(t), \cdots, z_n(t))$ of node values at time t, and $\mathbf{1}$ denote the vector of all 1s. Define τ_k as the kth time instant such that some local clock ticks and thereby triggers messaging in the network. For $\varepsilon > 0$ let the deterministic quantity $K(\varepsilon, P)$ be defined by

$$K(\varepsilon, P) = \sup_{z(0)} \inf \left\{ k \ : \ Pr\left(\frac{\|z(\tau_k) - \bar{x}\mathbf{1}\|_2}{\|z(0)\|_2} \geqslant \varepsilon \right) \leqslant \varepsilon \right\}.$$

In [7] $K(\varepsilon, P)$ is considered as a termination time for Algorithm `GOSSIP-AVE`(P) and minimization of $K(\varepsilon, P)$ is sought by proper choice of P. Here we adopt the same interpretation for comparison purposes. It should perhaps be noted here that this is a fairly weak stopping criterion as $\|z(\tau_{K(\varepsilon,P)}) - \bar{x}\mathbf{1})\|_\infty / |\bar{x}|$ may be much larger than ε.

The numerical results of average number of messages and run times for 2-D torus appear in Figure 4. The corresponding results for geometric random graphs and a illustrative comparison with Gossip is presented in Fig. 5. We have also plotted a bound $2\log(n)$ for comparison purposes. Notice that from the scale of the two plots it should be clear that the bound will have a similar qualitative relationship to the message complexity for the torus. These bounds reveal that the empirical per node message complexity appears to be closer to $O(\log(n))$ which is much smaller than the $O(\log^2(n))$ theoretical message complexity bound of Equation 18. One possibility for this difference is that our theoretical message complexity bound is for worst-case distribution of initial node values,



(a) Per-node message complexities

(b) Time complexities

Figure 5: Average run times and number of messages per node on random geometric graphs (RGG). The solid curve represents simulation results for `CRW`, the dashed curve for `SRW` whereas the dotted curve represent lower bounds for `GOSSIP-AVE`(P) based on a lower bound for $K(\varepsilon, P)$. Note also that exact value of $F_n(x_1, x_2, \cdots, x_n)$ is obtained at the termination of `CRW` or `SRW` whereas no such claim can be made for `GOSSIP-AVE`(P). Note for an accurate comparison, time and message complexity of CFLD needs to be added to that of CRW. Nevertheless, for RGG both time and message complexity are insignificant relative to that of CRW (see Section 6.3).

while the empirical result is for an average case distribution of the node values.

7 Appendix

7.1 Proof of Theorem 6.1

We follow the argument of [8] and provide a detailed proof to point out that the proof goes through for general graphs. The proof applies to both discrete and continuous settings and basically utilizes Markovianity. Let $\Lambda_B(t) : t \geq 0$, $B \subset V$ denote the occupied nodes (state) at time t of a coalescing random walk whose initial state is B. Observe that irrespective of the initial state, B, if $0 \leq s \leq t$, $E(|\Lambda_B(t)|) \leq E(|\Lambda_B(s)|)$.

We then have the following lemma:

Lemma 7.1 *Suppose* $(X_t \mid t \geq 0)$ *is a simple symmetric unit rate continuous time random walk on graph* Γ *and* $B \subset A \subset V$*, then*

$$E[|\Lambda_B(s)|] \leq |B| - (|B| - 1)\alpha_s(A). \tag{27}$$

Proof. If $B = \emptyset$, Equation (27) trivially holds therefore assume B is non-empty. Our approach is to find an upper bound for the number of coalescences occurring in the time interval $[0, s]$. Our

analysis begins with starting random walks with active tokens at $v \in B\backslash w$ and at w, and see whether their paths ever meet. To formalize this approach define an indicator function $\mathcal{I}(\cdot)$ to indicate whether or not active tokens v and w meet in time t. Then,

$$Z(s) = \sum_{v \in B\backslash w} \mathcal{I}(C_{v,w} \leq s) \tag{28}$$

the random quantity $Z(s)$ is the number of active tokens $B\backslash w$ which coalesces with w at some time $0 \leq t \leq s$. It follows through conservation of active tokens that,

$$|\Lambda_B(s)| \leq |B| - Z(s)$$

Now,

$$E(Z(s)) = \sum_{v \neq w} Prob(C_{vw} \leq s) \geq \min_{v,w \in B} Prob(C_{vw} \leq s)(|B| - 1) = \alpha_s(A)(|B| - 1)$$

The result now follows by taking expectations on both sides in Equation 28 and substituting the above expression. □

Now consider the partition $A_i, i = 1, 2, \ldots, m(t)$ of the vertices of the graph as in the hypothesis of the theorem and let $B_j = A_j \cap B$.

Lemma 7.2 $|\Lambda_B(s)| \leq \sum_{j=1}^{m(t)} |\Lambda_{B_j}(s)| \quad \forall\ s \geq 0$

Proof. Let $c_{ij}(s)$ be the number of active tokens starting in A_i and coalescing with active tokens starting in A_j. Then,

$$|\Lambda_B(s)| = \sum_{j=1}^{m(t)} |B_j| - \sum_{i=1}^{m(t)} \sum_{j=1}^{m(t)} c_{ij}(s) \leq \sum_{j=1}^{m(t)} (|B_j| - c_{jj}(s)) = \sum_{j=1}^{m(t)} |\Lambda_{B_j}(s)| \tag{29}$$

□

Now using the Markov property we can upperbound the number of active tokens at any time as follows. In the beginning all the nodes of the graph $G = (V, E)$ are active. Hence we need to analyze $N(t) = E\left[|\Lambda_V(t)|\right]$. Suppose $O_i \subset V$ be an arbitrary subset of V. Since V is finite the collection of all subsets, $\{O_i,\ i \in \mathcal{I}\}$, can be indexed by a finite index set $\mathcal{I}$. Denote $O_{ij} = A_i \cap O_j$. It follows that,

$$\begin{aligned}
N(t+s) &= E\left[|\Lambda_V(t+s)|\right] \qquad (30)\\
&= E\left[E\left[|\Lambda_V(t+s)| \mid \Lambda_V(t)\right]\right] = \sum_{j \in \mathcal{I}} Prob(\Lambda_V(t) = O_j) E\left[|\Lambda_V(t+s)| \mid \Lambda_V(t) = O_j\right]\\
&\overset{(a)}{=} \sum_{j \in \mathcal{I}} Prob(\Lambda_V(t) = O_j) E\left[|\Lambda_{O_j}(s)|\right]\\
&\overset{(b)}{\leq} \sum_{j \in \mathcal{I}} Prob(\Lambda_V(t) = O_j) \left(\sum_{i=1}^{m(t)} E\left[|\Lambda_{O_j \cap A_i}(s)|\right]\right) = \sum_{j \in \mathcal{I}} Prob(\Lambda_V(t) = O_j) \left(\sum_{i=1}^{m(t)} E\left[|\Lambda_{O_{ij}}(s)|\right]\right)
\end{aligned}$$

where, (a) follows from Markovianity, (b) follows from Lemma 7.2. Next, since for all i, j we have $O_{ij} \subset A_i \subset V$ we can apply Lemma 7.1 and obtain:

$$\begin{aligned}\sum_{i=1}^{m(t)} E[|\Lambda_{O_{ij}}(s)|] &\leq \sum_{i=1}^{m(t)} (|O_{ij}| - (|O_{ij}|-1)\alpha_s(\mathcal{A}_t)) \\ &= (1-\alpha_s(\mathcal{A}_t)) \sum_{i=1}^{m(t)} |O_{ij}| + m(t)\alpha_s(\mathcal{A}_t) = (1-\alpha_s(\mathcal{A}_t))|O_j| + m(t)\alpha_s(\mathcal{A}_t)\end{aligned}$$

Substituting this result in the inequality (30) we obtain

$$\begin{aligned}N(t+s) = E[|\Lambda_V(t+s)|] &\leq (1-\alpha_s(\mathcal{A}_t)) \sum_{j\in\mathcal{I}} Prob(\Lambda_V(t) = O_j)|O_j| + m(t)\alpha_s(\mathcal{A}_t) \\ &= (1-\alpha_s(\mathcal{A}_t))E[|\Lambda_V(t)|] + m(t)\alpha_s(\mathcal{A}_t) \\ &\overset{(a)}{\leq} \left(1 - \frac{\alpha_s(\mathcal{A}_t)}{2}\right) E[|\Lambda_V(t)|] \leq \exp\left(-\frac{\alpha_s(\mathcal{A}_t)}{2}\right) N(t)\end{aligned} \tag{31}$$

where (a) follows from the choice of the number of partitions that it is one half of the number of active tokens at time t. To prove Equation 13 we note from Equation 31 that for any r and s such that $t \leq r \leq r+s \leq 2t$ we have,

$$\begin{aligned}N(r+s) &\leq (1-\alpha_s(\mathcal{A}_t))N(r) + m(t)\alpha_s(\mathcal{A}_t) \leq (1-\alpha_s(\mathcal{A}_t))N(r) + \frac{\alpha_s(\mathcal{A}_t)}{2}N(2t) \\ &\leq (1-\alpha_s(\mathcal{A}_t))N(r) + \frac{\alpha_s(\mathcal{A}_t)}{2}N(r) \leq \left(1 - \frac{\alpha_s(\mathcal{A}_t)}{2}\right) N(r) \leq \exp\left(-\frac{\alpha_s(\mathcal{A}_t)}{2}\right) N(r)\end{aligned}$$

where the third inequality follows from the fact that since $r \leq 2t$ we have $N(2t) \leq N(r)$. Now iterating over s $\left\lfloor \frac{t}{s} \right\rfloor$ Equation 13 follows.

8 Proof of Theorem 6.3

We first consider the case where

$$2 \leq \frac{N(t)}{4} \leq \frac{N(2t)}{2} \tag{32}$$

This implies that $N(t) \geq 8$ and $2N(2t) \geq N(t)$. If these assumptions are violated then we are in the case where either $N(t) \leq 8$ or

$$N(2t) \leq \frac{1}{2}N(t)$$

First, consider the situation when Equation 32 is satisfied. We will choose the collection $\mathcal{A}_t$ and the time step s so that assumptions underlying Equation 13 are satisfied. Specifically, we let $\mathcal{A}_t$ be the collection of balls $B(u, R_t)$ of radius R_t for suitable vertices $u \in V$ to cover the graph G. We select radius, R_t as follows:

$$R_t = \sqrt{\frac{8n}{C_0 N(t)}},\ s = s_t = R_t^2$$

where C_0 is the constant satisfying Equation 19. We can assume that, $s \leq t/2$. This is because if this condition is violated then, we have

$$N(t) \leq \frac{128n}{C_0 t} \tag{33}$$

which satisfies the condition of the Theorem and there is nothing to prove.

So we suppose $s \leq t/2$. The number of partitions,

$$m(t) \leq \left\lfloor \frac{n}{C_0 R_t^2} \right\rfloor + 1 \leq \left\lfloor \frac{N(t)}{8} \right\rfloor + 1 \leq \left\lfloor \frac{N(t)}{4} \right\rfloor.$$

This ensures that assumptions underlying Equation 13 are satisfied. Consequently, we get

$$N(2t) \leq N(t) \exp\left(- \left(\frac{t}{2R_t^2} \right) \frac{C_2}{\log(R_t)} \right)$$

Denoting

$$f_t = \frac{N(t)}{n} \frac{t}{\log(t)},\ t \geq 2$$

and substituting for

$$R_t = \sqrt{\frac{8n}{C_0 N(t)}} = \sqrt{\frac{8t}{C_0 f_t \log(t)}}$$

we get,

$$\begin{aligned} f_{2t} &\leq f_t \frac{2\log(t)}{\log(2t)} \exp\left(- \left(\frac{t}{2R_t^2} \right) \frac{C_2}{\log(R_t)} \right) \leq f_t \exp\left(\log(2) - \left(\frac{t}{2R_t^2} \right) \frac{C_2}{\log(R_t)} \right) \\ &\leq f_t \exp\left(\log(2) - \left(\frac{C_0 f_t \log(t)}{16} \right) \frac{C_2}{0.5 \log(\frac{8t}{C_0 f_t \log(t)})} \right) \\ &\leq f_t \exp\left(\log(2) - \frac{C_0 C_2}{8} f_t \frac{\log(t)}{\log(\frac{8}{C_0}) - \log(f_t) + \log(\frac{t}{\log(t)})} \right) \end{aligned}$$

where the second inequality follows from the fact that $\log(t)/\log(2t) \leq 1$. Now we note that $f_t \leq t/\log(t)$. Consequently, if $\frac{8}{C_0} < f_t$ then

$$\frac{\log(t)}{\log(\frac{8}{C_0}) - \log(f_t) + \log(\frac{t}{\log(t)})} \geq 1.$$

Also simultaneously if $f_t \geq \frac{8\log(2)}{C_0 C_2}$ we get $f_{2t} \leq f_t$. On the other hand if any of these conditions are violated we get

$$f_{2t} \leq 2 \max(\frac{8}{C_0}, \frac{8\log(2)}{C_0 C_2}) \triangleq \tilde{C}.$$

This implies that,

$$f_{2t} \leq \max\left(\tilde{C}, f_t\right) \implies N(2t) \leq \max\left(\tilde{C} \frac{n \log(2t)}{2t}, \frac{N(t)}{2}(1 + 1/\log(t)) \right)$$

Finally, we have two cases to consider: (1) if Equation 32 itself is violated we have, $N(2t) \leq \frac{1}{2} N(t)$; (2) Equation 33 holds. In all of these cases Eq. 21 is satisfied and the result follows.

8.1 Proof of Lemma 6.1

Proof. The proof follows directly along the lines of Pettarin et al [18] (Lemma 9). We provide a brief sketch of their proof here for the sake of completion. First, let $P_t(x, y)$ denote the probability that a walk starting at node x at time zero is at node y at time t. Now consider two walks, one starting at node u and another starting at node w at time zero. Note that the two walks are independent and they have their own corresponding transition probabilities. Let $N(u, w, T_0)$ be the mean number of times the two walks starting at u and w meet in the time interval $[0, T_0]$. Then noting that the two walks could meet at any node v at any time $t \in [0,\, T_0]$ we obtain,

$$N(u, w, T_0) = \sum_{t=0}^{T_0} \sum_{v} P_t(u, v) P_t(w, v)$$

This is because $P_t(u, v)P_t(w, v)$ is the probability that both walks starting at u and w respectively are at the same node v at time t. Summing over the different possibilities leads to the above result. Using this fact Pettarin et al establish that,

$$\alpha_{R^2}(B(u, R)) \geq \frac{N(u, w, R^2)}{\max_{v \in B(u,R)} N(v, v, R^2)},\ w \in B(u, R)$$

Here, $N(v, v, R^2)$ is the number of times two walks starting at the same node v meet again in the time interval $[0, R^2]$. The problem now boils down to lower bounding $N(u, w, R^2)$ and upper bounding $N(v, v, R^2)$. We are now ready to substitute the Gaussian t-step bounds to establish the result. Specifically, let

$$D = \{v \in V \mid d(v, u) \leq 2R,\, d(v, w) \leq 2R\}$$

We also note that since $B(u, R) \subset D$ and the graph satisfies the geometric neighborhood property we have $|D| \geq C_0 R^2$. So

$$\begin{aligned} N(u, w, R^2) &= \sum_{t=0}^{R^2} \sum_{v} P_t(u, v) P_t(w, v) \geq \sum_{t=R^2/2+1}^{R^2} \sum_{v \in D} P_t(u, v) P_t(w, v) \\ &\geq \sum_{t=R^2/2+1}^{R^2} \sum_{v \in D} \left(\frac{C_3}{t}\right)^2 \exp\left(-\frac{d^2(v, u) + d^2(v, w)}{C_4 t}\right) \end{aligned}$$

By bounding $d^2(v, u)$ and $d^2(v, w)$ with $4R^2$ we obtain $N(u, w, R^2) = \Omega(1)$. Next we use the fact that there are no more than $C_1 k\Delta$ nodes in any annulus of size Δ at distance k to obtain an upper bound for $N(v, v, R^2)$. Specifically, by taking an annulus of size one, our geometric condition implies that there are no more than $C_1 R$ nodes at distance R. So,

$$\begin{aligned} N(v, v, T) &= \sum_{t=0}^{T} \sum_{x} P_t(v, x) P_t(v, x) \leq 1 + \sum_{t=1}^{T} \sum_{k=1}^{t} \sum_{d(v,x)=k} P_t(v, x) P_t(v, x) \\ &\leq 1 + \sum_{t=1}^{T} \sum_{k=1}^{t} C_1 k \left(\frac{C_3}{t}\right)^2 \exp\left(\frac{-2k^2}{t}\right) \end{aligned}$$

The computation of the above sum follows along the same lines as in Pettarin et al (Lemma 9). It follows that $N(v, v, T) = O(\log(T))$ which is $O(\log(R))$ for $T = R$. Consequently, there is a constant C_2 such that,

$$\alpha_{R^2}(B(u,R)) \geq \frac{N(u,w,R^2)}{\max_{v \in B(u,R)} N(v,v,R^2)} = \frac{C_2}{\log(R)},\ w \in B(u,R)$$

□

References

[1] M. Alanyali, S. Venkatesh, O. Savas, and S. Aeron. Distributed bayesian hypothesis testing in sensor networks. *American Control Conference*, Boston, MA, July 2004.

[2] D. Aldous and J. Fill. Reversible Markov Chains and Random Walks on Graphs *manuscript in preparation.*

[3] R. Aleliunas, R. M. Karp, R. J. Lipton, L. Lovasz and C. Rackoff, Random walks, universal traversal sequences, and the complexity of maze problems, Proceedings of the 20th IEEE Symposium on Foundations of Computer Science, 1979.

[4] C. Avin and G. Ercal, On the Cover Time of Random Geometric Graphs, ICALP. 2005,

[5] D.P. Bertsekas and J.N. Tsitsiklis, Parallel and distributed computation: numerical methods. Prentice-Hall, 1989.

[6] D.P. Bertsekas and R. Gallager, Data Networks. Prentice-Hall, 1992.

[7] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah. Randomized gossip algorithms. IEEE Transactions on Information Theory. 2006.

[8] J.T. Cox. Coalescing random walks and voter model consensus times on the torus. *The Annals of Probability,* 17(4):1333-1366, 1989.

[9] A. K. Chandra, P. Raghavan, W. L. Ruzzo, R. Smolensky and P. Tiwari, The Electrical Resistance Of A Graph Captures Its Commute And Cover Times, 1989

[10] F. R. K. Chung, Spectral Graph Theory, AMS 1997

[11] A.G. Dimakis, A.D. Sarwate, and M.J. Wainright, Geographic gossip: efficient aggregation for sensor networks. IPSN 2006.

[12] S. N. Ethier and T. G. Kurtz, Markov Processes, John Wiley and Sons Inc., New York, 1986.

[13] A. Grigor'yan and A. Telcs, Harnack inequalities and sub-Gaussian estimates for random walks, Math. Annalen, 2002

[14] W. R. Heinzelman, J. Kulik and H. Balakrishnan, Adaptive Protocols for Information Dissemination in Wireless Sensor Networks, Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, 1999

[15] D. Kempe, A. Dobra and J. Gherke, Gossip-based computation of aggregate information. In Proc. IEEE Conference of Foundations of Computer Science (FOCS), 2003.

[16] W. Li and H. Dai, LADA algorithms: Performance lower bounds and cluster-based variant, Allerton Conference on Communication, Control and Computing, Monticello, IL, Sept. 2007.

[17] A. B. Merkov, Second-order elliptic equations on graphs, Math. USSR, 1986.

[18] A. Pettarin, A. Pietracaprina, G. Pucci and E. Upfal, Infectious Random Walks, ArXiv e-prints 1007.1604, 2010

[19] R. Rahman, M. Alanyali, and V. Saligrama, Distributed Tracking in Multi-hop Sensor Networks with Communication Delays, IEEE Transactions on Signal Processing. Vol. 55, no.9, pp. 4656-4668. September 2007.

[20] V. Saligrama, M. Alanyali and O. Savas. Distributed detection in sensor networks with packet losses and finite capacity links. *IEEE Transactions on Signal Processing*, to appear, 2005.

[21] O. Savas, M. Alanyali, and V. Saligrama, Randomized Sequential Algorithms for Data Aggregation in Sensor Networks," CISS 2006, Princeton, NJ.

[22] D.S. Scherber and H.C. Papadopoulos. Distributed computation of averages over ad hoc networks. *IEEE Journal on Selected Areas in Communications,* vol. 23(4), 2005.

[23] S. Tavaré. Line-of-descent and genealogical processes, and their applications in population genetics models. *Theoret. Population Biol.,* vol. 26, pp. 119-164, 1984.

[24] D. Zuckerman, A technique for lower bounding the cover time, *SIAM Journal on Discrete Mathematics,* no. 5, pp. 81-87, 1992.