Work-in-Progress: A New Random Walk for Data Collection in Sensor Networks*

Constantinos Marios Angelopoulos*, Sotiris Nikoletseas *, Dimitra Patroumpa *, Christoforos Raptopoulos *
*University of Patras, Greece and
Research Academic Computer Technology Institute (RACTI), Greece

Emails: aggeloko@ceid.upatras.gr, nikole@cti.gr, patroump@ceid.upatras.gr, raptopox@ceid.upatras.gr

***Abstract*—Motivated by the problem of efficient sensor network data collection via a mobile sink, we present undergoing research in accelerated random walks on Random Geometric Graphs. We first propose a new type of random walk, called the α-*stretched random walk*, and compare it to three known random walks. We also define a novel performance metric called *Proximity Cover Time* which, along with other metrics such us visit overlap statistics and proximity variation, we use to evaluate the performance properties and features of the various walks. Finally, we present future plans on investigating a relevant combinatorial property of Random Geometric Graphs that may lead to new, faster random walks and metrics.**

I. Introduction

Multihop data propagation protocols in sensor networks, where the sink is static, tend to have increased implementation complexity causing the sensor devices to consume significant amounts of energy in operations other than sensing (i.e. inter-node communication for synchronisation purposes, exploratory messages, etc). Towards a more balanced and energy efficient method of data collection, sink mobility can be used. The main idea is that the sink has significant and easily replenishable energy reserves and can move inside the network region, each time being in close proximity to a (usually small) subset of the sensor devices to collect their data.

Random walks in wireless sensor networks can serve as fully local, very simple strategies for sink motion that significantly reduce energy dissipation a lot but increase the latency of data collection. To achieve satisfactory energy-latency trade-offs the sink walks can be made adaptive or biased; however, this increases the memory requirements and strengthens the network model assumed . Towards a better balance of memory/performance, we propose a new type of random walk, the α-stretched random walk, that we compare to already existing walks: the Blind Random Walk, the Random Walk with Inertia and the Random Walk with Memory 1. We plan to thoroughly evaluate the performance of these walks on Random Geometric Graphs (from now on referred to as RGG) using a series of relevant, novel metrics.

II. The network model

Sensor networks comprise of a vast number of ultra-small homogeneous sensor devices (which we also refer to as sensors), whose purpose is to monitor local environmental conditions. Each sensor is a fully-autonomous computing and communication device, characterized mainly by its available power supply (battery), its transmission range r, the energy cost of data transmission and the (limited) processing and memory capabilities. Sensors (in our model here) do not move. The positions of sensors within the network area are random and in the general case follow a uniform distribution. We focus on data collection methods, so we assume that initially all sensors have some data to deliver to the sink. For clarity, we also assume that no data is generated during the network traversal. That is, a node is called "visited" when it has no data to send to the sink.

There is a special node within the network region, which we call the sink *S*, that represents a control center where data should be collected. Here, we assume that the sink is *mobile*. The sink is not resource constrained i.e. it is assumed to be powerful in terms of computing, memory and energy supplies.

We consider that the random uniform placement of the sensors inside the network area is abstracted by a *Random Geometric Graph*. Random Geometric Graphs are formed by n vertices that are placed uniformly at random in the $[0,1]^2$ square. An edge (u, v) exists iff the Euclidean distance of vertices u and v is at most R. Clearly, R corresponds to the wireless communication radius r of the sensors. This holds assuming a disc radio model; two sensors can communicate with each other iff each one lies inside the communication range of the other. Random Geometric Graphs also have an important nice property: unlike other random graphs, like $G_{n,p}$, edges are not statistically independent of each other. That is, the existence of an edge (u, v) is not independent of the existence of edges (u, w) and (w, v). This property makes RGG a quite realistic model for wireless sensor networks that captures to a great extent the communication structure of real wireless sensor networks (at least its spatial aspects).

III. The α-stretched Random Walk

We propose a novel random walk that we believe will accelerate the data collection process while keeping the memory requirements restricted. The basic idea is that each time the next node the walk visits is chosen among the "more distant" neighbours of the current node.

* This work was partially supported by EU/FIRE HOBNET project - STREP ICT-257466

Let $G(\mathcal{X}_n; r)$ be a random instance of the random geometric graphs model with vertex set $V = \mathcal{X}_n$, where $|V| = n$. We will say that a vertex u is an *α-distant neighbour of vertex v* if $\|u-v\| \geq \alpha r$, where $\|u-v\|$ is the Euclidean distance between u, v and $0 < \alpha < 1$. Clearly, if u is an α-distant neighbour of v, then v is an α-distant neighbour of u. Furthermore, for any $0 \leq \alpha' \leq \alpha$, if u is an α-distant neighbour of v, then it is also an α'-distant neighbour of v. For any vertex $v \in \mathcal{X}_n$, we will denote the set of α-distant neighbours of v by $N^{(\alpha)}(v)$, i.e. $N^{(\alpha)}(v) = \{u \in N(v) : \|u-v\| \leq \alpha r\}$.

Consider a particle moving on the vertices of the graph $G(\mathcal{X}_n; r)$. Given that it occupies a specific vertex $v \in V$ at time $t \geq 0$, it decides where to move at time $t+1$ by choosing a vertex uniformly at random among the vertices in $N^{(\alpha)}(v)$. The Markov chain describing the above process will be called *α-stretched random walk on $G(\mathcal{X}_n; r)$* and will be denoted by $\mathcal{W}^{(\alpha)}_{G(\mathcal{X}_n;r)}$. Furthermore, we will denote by $\mathcal{W}^{\alpha}_v(t)$ the state of the walk that begins at v at time t. More formally, the state space of the α-stretched random walk is the set of vertices $\mathcal{X}_n$ of the graph. Furthermore, the transition probability matrix for any two states (i.e. vertices) $v, u \in \mathcal{X}_n$ is given by

$$P_{v,u} = \begin{cases} \frac{1}{|N^{(\alpha)}(v)|} & \text{if } u \in N^{(\alpha)}(v) \\ 0 & \text{otherwise.} \end{cases}$$

IV. Known Random Walks

We plan to compare the α-stretched random walk to three already known random walks.

1) Blind Random Walk : The Blind Random Walk on a RGG is the simplest of all possible sink mobility patterns, since the next move of the sink is stochastically independent to the previous one. If the mobile sink is currently on vertex u of degree deg_u, then the probability of moving to any neighbouring vertex v is $p_{v,u} = \frac{1}{deg_u}$.

This method is very robust, since it probabilistically guarantees that eventually all network regions and nodes will be visited and thus all data will be collected given that the network is connected. However, in some network structures it may become inefficient, mostly with respect to latency, since the sink uses no memory of the past movements in order to select the next one and thus overlaps (i.e. visits to already visited vertices) occur.

2) Random Walk with Memory: The performance of the Blind Random Walk can be improved using some memory of past visits. In Random Walk with Memory the sink maintains a first-in-first-out (FIFO) list M which contains the last K nodes visited during the random walk, i.e $M = \{c_1, c_2, ..., c_K\}$. The next hop is chosen uniformly among the neighbours of the node that are not in the memory list M.

The use of memory eliminates loops in random walks, but it may lead to deadlock situations. Without memory, i.e. K = 0, the random walks become blind and can have loops but no deadlocks. For complete memory, the random walks can only have deadlocks and no loops. When the size of M is $0 < K \leq n-1$, the random walks can have both loops and deadlocks. In this study we will evaluate the performance of Random Walk with Memory 1 on Random Geometric Graphs.

3) Random Walk with Inertia: In this type of walk [2] the sink performs a biased random walk by trying to maintain its current movement. More formally assume that the sink moves from vertex v to vertex u. Let C be the set of the neighbouring vertices of u. Then to each vertex $c_i \in C$, corresponds an angle $\widehat{c_i} = \widehat{vuc_i}$. The sink chooses to move towards c_i with probability $p_i = \frac{\widehat{c_i}}{\sum_{j=1}^{|C|} \widehat{c_j}}$. This walk requires a small, constant sized memory since the next step of the walk depends solely on the previous one. The sink makes long paths and traverses many different sub-regions of the network area very quickly, thus avoiding early overlaps. However, after most of the network area has been covered, there exist small unvisited sub-regions that are hard to find. The fact that many different sub-regions are visited very soon makes this walk suitable for time critical applications, such as reactive event detection.

V. Experimental Evaluation

In order to evaluate the performance of the random walks we are going to implement them in Matlab R2008b and evaluate them on a Random Geometric Graph of 400 nodes in the $[0,1]^2$ square. The evaluation will take place in rounds. Because RGG are probability distributions of network instances, in order to measure the mean values of the metrics, each round will consist of 10 iterations over 10 instances of a RGG of 400 nodes. Furthermore, we will examine the impact of several parameters like the communication range r of the sensors (i.e. sparse and dense RGG) and the α value of the α-stretched Random Walk.

A. Metrics

In our evaluation, we are going to use several informative metrics, that we describe below.

1) Proximity Cover Time: We propose a new metric called Proximity Cover Time. Denote by $D(v, \rho)$ the disc of radius $\rho > 0$, centered at v. Consider the (blind) random walk $\mathcal{W}_v$ that begins on vertex v of a random instance of the random geometric graphs model $G(\mathcal{X}_n; r)$. We will say that $\mathcal{W}_v$ is *within distance* ρ to vertex u at time t if $W_v(t) \in D(u, \rho)$, i.e. the random walk started at v occupies a vertex that lies in $D(v, \rho)$ at time t. Let $T_v(\rho) \stackrel{def}{=} \inf\{t | \forall u \in V, \exists t' \leq t : W_v(t') \in D(u, \rho)\}$ be the time needed until $\mathcal{W}_v$ has come within distance ρ to all vertices of the graph. We define the proximity cover time as follows:

Let $G(n, r) = (V, E)$ be a random instance of the random geometric graphs model $G(\mathcal{X}_n; r)$. The ρ-proximity cover time of $G(n, r)$, denoted by $C(\rho)$, is defined as follows

$$C(\rho) = \max_{v \in V} \mathbb{E}[T_v(\rho)],$$

where $\mathbb{E}$ above is the expectation of the time $T_v(\rho)$ (this time is a random variable).

A similar but different metric is presented in [1] where each time the random walk visits an already visited vertex, then u.a.r. picks a single unvisited neighbour (if there exists) and marks it as visited.

2) Proximity Variation: We also plan to evaluate *the mean value (over all cells) of the smallest distance from the sink for any node* introduced in [2]. More strictly, let $dist(x)$ be the function that returns the distance of node x from the sink. Then Proximity Variation (PV), equals:

$$PV = \frac{\sum_{i=1}^{n} min(dist(i))}{n}$$

where n is the total number of nodes and minimum of $dist(i)$ is taken over time.

The rationale behind this metric is the following: there is a different way behind each walk traverses the network, regardless of the total number of hops needed. For example, while the Walk with Inertia tends to create long straight lines by keeping the same direction as long as it discovers new nodes, the Blind Random Walk makes a Brownian-like motion with small dislocation, thus making many overlaps while visiting neighbouring sub-regions. As the Walk with Inertia traverses many different sub-regions early in the network traversal process, the sum of the smaller distance each cell has had from the sink decreases significantly faster compared to the Blind Random Walk.

In a real network this corresponds to the way of covering the network area. If the PV metric converges to zero quickly, this means that the sink gets close to all sensors quite soon and data collection progresses fast; this is especially relevant in case when the role of sensors is not completely passive but includes some limited multihop propagation of data to accelerate data propagation at a reasonable energy cost. On the contrary, when the PV converges slowly to zero, it means that the network traversal is performed in a way that some areas may stay unvisited for long time.

3) Visit overlap statistics: Other metrics we plan to use include counting, during each walk's evolution, how many times each vertex of the graph is visited. Also, what is the distribution of the overlaps inside the network area and what is the inter-arrival time interval (time period between two successive visits of the same vertex). These overlap statistics will provide an insight on how exactly each walk traverses the graph and how different components of the graph are visited and in what rate.

4) Cover times: Finally, we also plan to measure the cover time and the partial cover time of each walk, i.e. the expected time needed for the random walk to visit all nodes (respectively, the largest part of them). These metrics are related to latency, as they capture the time the sink needs to collect the sensory data from the entire network. The partial cover time is the expected number of hops needed to cover a specific percentage of the entire network area (usually 90% - 95%). This metric is of great interest as the majority of overlaps occur while the sink tries to locate the last few unvisited sub-regions that are scattered in the network area; however, in most sensor network applications it is sufficient to collect a vast percentage of the total sensory data, so this metric is relevant and informative.

B. Expected Performance Results

The Random Walk with Inertia and the α-stretched Random Walk are expected to perform much better than the Blind Random Walk in terms of Proximity Cover Time and Proximity Variation, while the performance of the Random Walk with Memory 1 will be lying somewhere in between. This is due to the tendency of the first two walks to quickly change subregions, thus getting close enough to every node in the network early in their evolution. Due to the same reason, these two walks also avoid early overlaps, in contrast to the Blind Random Walk, leading us to expect significant differentiation also in terms of the overlap statistics. In terms of cover time and partial cover time we expect the Random Walk with Memory 1 and the Random Walk with Inertia to perform slighlty better than the newly proposed α-stretched Random Walk as they use more memory. However, we do not expect to observe great variations in these metrics as all walks face difficulties in locating the last few unvisited vertices towards the end of their evolution.

VI. Future Work

In [3] the authors prove a very interesting combinatorial property for $G_{n,p}$ random graphs. Consider a random walk on $G_{n,p}$ and let $\bar{G}_t$ be the induced subgraph of $G_{n,p}$ on the set of vertices not visited until time t. There is a time t^* with the following property: for $t \leq (1-\epsilon)t^*$ the graph $\bar{G}_t$ has a unique giant component, plus components of size $O(\log n)$, while for $t \geq (1-\epsilon)t^*$ all components of $\bar{G}_t$ are of size $O(\log n)$. We plan to experimentally investigate this property on RGG and if findings are encouraging to analytically compute the value of t^* for RGG. If such a property exists, we plan to exploit it by defining a new random walk that will take advantage of it for improving the cover time. We finally plan to formally define and study a novel metric, called "compactness" that will keep track of the evolution of the size of the largest unvisited component during the evolution of a random walk. High compactness could eventually lead to heuristics for accelerating the cover time of random walks.

References

[1] Berenbrink P., Cooper C., Elsasser R., Radzik T., Sauerwald T., *"Speeding up random walks with neighbourhood exploration"*, SODA '10 Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms.

[2] C. M. Angelopoulos, S. Nikoletseas and D. Patroumpa, *"Coverage - Adaptive Random Walks for Fast Sensory Data Collection*, in the Proceedings of The 9th International Conference on Ad Hoc Networks and Wireless (ADHOC-NOW), Edmonton, Canada, 2010.

[3] C. Cooper and A.M. Frieze, *"Component structure of the vacant set induced by a random walk on a random graph."*, in Random structures and Algorithms, 32, pp.401-439, (2008).