Hierarchical neighbor graphs: A fully distributed topology for data collection in wireless sensor networks

Amitabha Bagchi, Adit Madan and Achal Premi
Indian Institute of Technology, Hauz Khas, New Delhi 110016
bagchi@cse.iitd.ernet.in, madanadit@gmail.com, achalpremi@gmail.com

Abstract

We introduce hierarchical neighbor graphs, a new topology control mechanism for wireless sensor networks. This mechanism is a randomized one that takes a single parameter, $0 < p < 1$, and uses it to build a structure that has the flavor of hierarchical clustering and is fully distributed in the sense that it requires only local knowledge at each node to be formed and repaired, and moreover requires minimal computation in this process. Hierarchical neighbor graphs naturally account for differences in the battery power of nodes and are able to use energy efficiently by reorganizing dynamically–without any global coordination or communication–when the battery power of heavily utilized nodes decreases. In this paper we study the lifetime and delay of hierarchical neighbor graphs, giving analytical characterizations of both. We perform simulations to demonstrate the sensitivity of the lifetime and delay to the number of nodes in the network and the parameter p, also studying the tradeoff between these two quantities and the effects of application-specific data aggregation policies. Through extensive simulations we compare hierarchical neighbor graphs against other leading proposals for data collection in wireless sensor networks and demonstrate that in general our structure provides better lifetime values than most other structures, and, importantly, is able to deal with heterogeneous distributions of initial battery power much better than previous proposals.

I. Introduction

For sensors placed in a remote unattended environment the ability to form a multi hop network that relays sensed data back to a base station that may lie outside the sensed region is crucial. This problem is known as the topology control problem for wireless sensor networks (WSNs) and has been extensively studied in the literature [9]. In this paper we present a novel architecture for the topology control problem, the *hierarchical neighbor graph* (HNG). The key idea of our construction is that each node is assigned a level that is determined partly by the battery power of the node and partly by a geometric random variable with parameter p. Each node chooses as its parent the node nearest to it whose level is strictly greater than its own and connects to this parent, also connecting to all other nodes of its own level that are closer than its parent. Our structure is distinguished from the several other structures proposed for topology control by the property that it is fully distributed: both topology construction and maintenance can be achieved using local actions at individual nodes without the need for any global information.

The HNG has all the benefits of a bounded degree hierarchical structure. Additionally the structure also seamlessly handles changes in battery power and heterogeneity in the initial battery power of the nodes in the network, thereby making it a more pragmatic solution than other proposals. In this paper we study the properties of hierarchical neighbor graphs in light of the constraints and requirements of the problem setting and demonstrate analytically and through simulation that our architecture compares favorably with the leading solutions to this problem proposed in the literature.

The basic issue of topology control for wireless networks, not necessarily in the context of WSNs, is that of choosing the right topology for wireless nodes that transmit under the constraint of interference. Keeping the *degree* of the network low is critical in this case, while ensuring *connectivity* i.e. that every node has a path to the base station. For a detailed treatment of this more general problem see [15]. In the WSN situation where all communication is between the nodes and a single base station the primary measure of the quality of such a network is the *throughput* between the nodes and the base station. But that is not the only criterion. Specific to wireless *sensor* networks is also the problem of energy conservation. Sensor nodes are small devices with limited battery power. In remote environments we cannot assume that the batteries can be recharged easily or replaced. So, research in this area operates under the assumption that each node of the network lives till it has the battery power to perform its function and hence *network lifetime* is a critical parameter of any proposal. Additionally any asymmetry in the energy expended in the process of data collation means that the roles of nodes have to change as their battery power changes. Hence the *cost of restructuring* has to be accounted for as it eventually affects network lifetime. The primary advantage of HNG is that restructuring is based on locally available information at every node and is oblivious to what happens in other parts of the network. Hence, fewer messages have to be sent and our restructuring cost is low.

Application-specific issues also arise. In applications like target tracking or radiation level monitoring, the *delay* between sensing the data and its arrival at the base station has to be as low as possible. In other applications, the data sensed is often redundant and highly correlated and the energy required for transmitting data is often much more than the energy required to aggregate the data. Data fusion at some nodes can reduce the number of messages transmitted through the network and hence improve its energy efficiency (see e.g. the survey by Rajagopalan and Varshney [14]). We demonstrate that hierarchical neighbor graphs are well suited to taking advantage of data aggregation to improve efficiency. We study cases where all the data has to be fused into a single value (e.g. finding the maximum of a set of sensed values which could be the quantity of interest in radiation level monitoring in a nuclear plant [14]) and also cases where there is a bounded amount of compression that can be achieved without losing information (e.g. temperature monitoring where only neighboring sensors may have values close to each other.)

Main contributions.: (i) We describe hierarchical neighbor graphs, a new structure for data collection in wireless sensor networks, detailing the algorithms for topology construction and maintenance (Section II). (ii) We describe the operation of a network built on hierarchical neighbor graphs and provide analytical characterizations of the network lifetime and delay (Section III. (iii) Through extensive simulation studies we investigate the sensitivity of our structure to various parameters and also compare the lifetime and delay of our mechanism to that of leading proposals for this problem (Section IV).

A. Related Work

Several architectures for collecting data in sensor networks have been proposed in the literature. We do not attempt a comprehensive survey of the literature here, referring the reader to [9] and [1] instead. In this section we mention some of the prominent proposals reported in the literature and discuss their properties in relation to HNGs.

Hierarchical neighbor graphs can be classified along with a set of proposals that are cluster-based in structure. These include LEACH [6], TEEN [12], APTEEN [13], HEED [21], [19]. One major advantage the HNG has over these schemes is that cluster heads choose themselves randomly and no network-wide coordination is required. Schemes based on chains are also reported in the literature e.g. PEGASIS [11]. Chain based protocols are more energy efficient than cluster based protocols but, as expected experience very large delay, making them unsuitable for time-critical applications. The delay is linear in the size of the network. In HNG, by contrast, because of the hierarchical nature the delay is logarithmic in the size of the network.

Another class of architectures are based on methods of building spanning structures like connected dominating sets or spanning trees out of the sensor network. Among these are [8], PEDAP [18] and trees built on voronoi tessellation of points of independent homogeneous poisson point processes [2]. The advantage of HNGs over these methods is that they require complicated algorithms to be run, often in a centralized fashion. Maintaining and repairing such structures is a non-trivial task, unlike HNGs which can be maintained very easily. This is an advantage HNGs have over other classes of architectures as well. In LEACH, for example, nodes must know the number of nodes in the network and the cluster heads of the network and an estimate of energy remaining in the network at the end of each round. Both PEGASIS and H-PEGASIS require chain building using greedy approach for which they assume that nodes have global knowledge of the network.

Like our hierarchical neighbor graphs that build a connected structure by sampling repeatedly many of the schemes reported in the literature also have hierarchical versions e.g. H-PEGASIS [10], the concentric clustering scheme for PEGASIS [16] and COSEN [17]. As expected these do better than PEGASIS in terms of delay since their hierarchical nature helps in simultaneous multiple chain buildings and hence reduced delays. The tradeoff is that constructing hierarchical versions of these schemes has great computational overhead while HNGs are organically hierarchical by their very nature.

Finally we note that HNGs belong to a class of topologies that are based on selecting connections from among the neighbors of a node. The model of choosing a fixed number of neighbors has been used to construct connected topologies [5]. The fundamental problem with this approach is that the number of neighbors required to achieve a connected network, and so the degree of the network, scales up as the logarithm of the size of the network when the nodes are placed randomly [20], [4] unlike HNGs in which the expected degree remains constant even as the size of the network scales up. Additionally HNGs have better delay because of their hierarchical nature.

II. Hierarchical Neighbor Graphs

In this section we define hierarchical neighbor graphs and explain how to construct them and how to adapt to decrease in battery power and death of individual sensor nodes. We postpone to Section III a discussion of how the network operates.

A. The structure

Consider a set of points $V \subset \mathbb{R}^2$. We are given a function $w : V \to \mathbb{R}_+$ such that each node $u \in V$ has a battery power $w(u)/c$ associated with it, where c is a constant determined by the minimum battery power a node needs to operate. Taking a parameter p such that $0 < p < 1$, we form the p-hierarchical neighbor graph on V with weight function w, denoted $\mathsf{HN}_p^w(V)$ as follows:

1) We create a sequence $\{S_n : n \geq 0\}$ of subsets of V such that $S_0 = V$. S_i is populated in two ways, one deterministic and one randomized.
   - Deterministically, all $u \in S_{i-1}$ with $\lfloor \log_{\frac{1}{p}} w(u) \rfloor \geq i$ are put into S_i.
   - The remaining points of S_{i-1} are placed in S_i with probability p independently of the choice of all other points.

2) After obtaining the sequence of sets, we say that the *level* $lev_p(u) = i$ such that $u \in S_i$ and $u \notin S_{i+1}$.
3) Each point $u \in V$ grows a circle around it which stops growing the first time a point of v with $lev_p(v) > lev_p(u)$ is encountered. u makes connections to all nodes w with $lev_p(w) = lev_p(u)$ that lie within this circle and to the node(s) of $S_{lev_p(u)+1}$ that lie on the circumference of the circle.

As a convention we assume that if n is the largest integer such that S_n is non-empty then the nodes of S_n are fully connected among themselves. Further we assume that all the nodes in S_n are connected to the base station.

Before moving on to discuss the issues involved in constructing this topology we make two observations. Firstly, note that if V is a finite set of nodes then it is obvious that $\mathsf{HN}_p^w(V)$ is connected. Secondly, we note that the way the structure is defined, it may be that nodes make connections with other nodes which are arbitrarily far away which is unrealistic given that wireless sensors have a limited transmission radius. However, it is easy to prove that if the nodes of V are uniformly distributed in a bounded region then the probability of having long connections decreases with the density of the nodes in the region. Hence by raising the density of the nodes to a suitably high value we can ensure with high probability that no node needs to connect beyond its transmission radius. We omit a formal proof in this extended abstract, referring the reader to [3] where we also define a radius-bounded version of HNGs.

B. Topology construction

Prior to deployment each node's level is determined as described above based on the initial battery power and random promotions. The random promotion can be hardwired into the sensor node. The deterministic promotion is determined by the battery power. When the nodes are deployed in the field, the process of network formation commences. It proceeds in three phases: *Phase 1: Advertise and listen.* (i) Each node sends out a message containing it's ID and level. (ii) Every node also receives advertisement messages from other nodes. (iii) The node disregards all messages from nodes with lower levels than its own, notes the messages from the nodes of its own level and level greater than its own. (iv) At the end of the phase it identifies as its parent the node with level greater than its own whose signal is the strongest. It also identifies the nodes of its own level whose signal is stronger than its parent. *Phase 2: Request connection.* Each node sends messages requesting connection to the nodes it has identified in the previous phase. A node may receive messages from nodes of lower level which have identified it as parent. *Phase 3: Make connection.* The nodes requested acknowledge the request and a connection is made.

Clearly the number of messages sent out in the *Advertise and Listen* phase is one per node i.e. $|V|$. Since every connection requested is made, the number of messages is twice the degree of the network. If we assume that $w(v) = 1$ for every $v \in V$, we can show the following theorem whose proof is given in the full version of this paper [3]:

Theorem 2.1: Given a point set V and a weight function $\mathbf{1}$ such that $\forall v \in V : \mathbf{1}(v) = 1$, the expected degree of any point $v \in V$ in $\mathsf{HN}_p^{\mathbf{1}}(V)$ constructed with parameter p, $0 < p < 1$ is at most $\frac{1}{p} + \frac{6}{p(1-p)}$. Moreover, if V is a Poisson point process with density $\lambda > 0$ then the expected degree is at most $\frac{7}{p}$.

Hence the expected number of messages per node sent in Phases 2 and 3 is independent of the size of V and of the density λ i.e. the total number of messages sent in constructing $\mathsf{HN}_p^{\mathbf{1}}(V)$ is $O(|V|)$.

C. Topology maintenance

Nodes at higher levels in the hierarchy get depleted of their energy quicker. This leads to heterogeneity in the network even if all sensor nodes had the same residual energy to begin with. We periodically restructure the network to distribute the energy load according to the residual energy of each node. In order for the network to correspond to the definition of an HNG, restructuring should be triggered every time the residual energy of a node goes down by a factor of p because this causes the level of the node to decrease by 1. In practice we need to schedule repair windows where a node interrupts its regular operation to check its energy level and repair itself. The details of how these repair windows are scheduled are postponed to Section III. Here we discuss how to repair nodes once the repair window has been entered and the node finds that it needs repair.

The repair process proceeds, like the construction process in three phases: *Phase 1: Demotion advertise and listen.* At the beginning of round i of the operation of the network if the energy of a node has decreased to the point that $\lfloor \log_{\frac{1}{p}} w(u) \rfloor$ has decreased from what it was at the beginning of round i, the node initiates a repair procedure by advertising its new lower level to all the nodes which requested it to be their parent. In this phase a node may receive a message from its parent with the parents new level. If that level falls to or below the nodes own level, the node goes into parent rediscovery mode. If the node's parent's new level is strictly lower than the node's level, the node initiates a disconnection procedure. *Phase 2: Parent rediscovery.* The node whose parent has fallen sends out a message seeking a new parent. This message contains its level. A node that receives a parent rediscovery message with a level lower than its own responds by sending out its own level and id. *Phase 3: Parent request.* The node seeking a parent sends a parent request message to the node of level strictly greater than its own that has the strongest signal. *Phase 4: Make connection.* The parent-to-be receives the parent request and a connection is established.

We note that when the node loses power it does not need to find a new parent for itself. It may need a new parent only if its own parent's level decreases. To prevent nodes from having to repair multiple times a round, we assume that nodes have a predefined order, based on their id, according to which they initiate repair at the beginning of the phase. As in the case of construction, the number of messages can be estimated by the sum of the degrees of the nodes that had the demoted node as their parent node before repair. As before, this can be bounded in expectation by $O(1/p^2)$ which is be independent of the size of the point set. Also, it is possible that the battery power falls to the point where the sensor node becomes non-functional, in this case the node simply exits the network.

III. NETWORK OPERATION, LIFETIME AND DELAY

In Section III-A we describe in detail the operation of a WSN organized as a hierarchical neighbor graph. In order to contextualize and analyze the operation of the network we characterize two important parameters of the network in Section III-B: the energy spent per unit transmission at each level of the network, and the expected number of children of a node. In Section III-C we analytically characterize the lifetime of the network in terms of the parameter p, demonstrating that the lifetime increases with decreasing p while the delay increases. Hence there is a tradeoff between these two criteria which we will explore further through simulations in Section IV.

A. Network operation

We assume synchronous operation of the network, assuming that in each time slot each node senses the environment and generates a packet of length k bits. We assume that each packet generated by a sensing action comes with a time stamp or sequence number. In order to relay this data to the BS, the nodes of $\mathsf{HN}_p^w(V)$ relay the node to their parents along with the time stamp. When the data reaches the nodes of level $ht_p^w(V) = \max_{v \in V} lev_p(v)$, they relay the data directly to the BS. At each level in the hierarchy nodes receive data from their children and aggregate according to the application before relaying it to their own parent (or the BS). In order to avoid interference we use TDMA and CDMA. All nodes use CDMA to communicate with their parents, the code being communicated to them at the time of making connection (Phase 3 of topology construction.) This ensures that the communication from a child to one parent doesn't interfere with the communication from another node to its parent. The children of a given parent use TDMA amongst themselves. The operation of the network proceeds in phases.

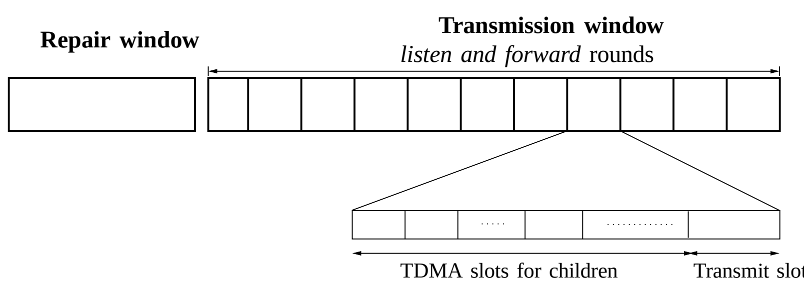


Fig. 1. One round in the operation of a node of S_i.

Each phase consists of a *transmission window* and a *repair window* (see Figure 1). The transmission window consists of *listen and forward rounds*, the number of which varies from node to node. The *listen and forward* round differs depending on the nature of data aggregation allowed by the application:

- *Unlimited aggregation.* A node collects all the packets corresponding to a single timestamp from its children, aggregates them into a single packet and forwards its up to its parent along with the timestamp.
- *Limited aggregation.* If the application allows a compression factor of $c < 1$, each node collects $\lfloor 1/c \rfloor$ packets corresponding to a single timestamp from its children, aggregates it and forwards its up to its parent along with the timestamp.

Note that since children use TDMA to communicate with their parent in both cases a node may have to wait for its children to send sufficient data before it can create a packet send it up to its parent. At the end of every *listen and forward* round the node checks its battery level. If the level has fallen by a factor of p since the last repair window the node initiates a new repair window in which repair is performed as described in Section II-C.

B. Two network parameters

a) Energy dissipated in transmission: The energy cost per unit data transmitted over a distance d by a node is given by $E_{elec} + \epsilon_{fs} d^2$ where E_{elec} is the radio electronics energy that depends on the coding and spreading of the signal, and ϵ_{fs} is the amplifier constant that depends on the acceptable bit-error rate. Since in a WSN organized as an HNG all transmissions take place from a node to its parent, the quantity of interest is the expectation of the square of the distance $d(u, \textit{parent}_p(u))$. For a homogeneous network, i.e. every node in $v \in V$ has $w(u) = 1$ we claim the following:

Claim 3.1: Given a Poisson point process V with parameter $\lambda > 0$ and $\mathsf{HN}_p^{\mathbf{1}}(V)$ with $0 < p < 1$, for any $u \in V$,

$$\mathrm{E}[(d(u, \textit{parent}_p(u))^2 \mid \textit{lev}_p(u)] = i = \frac{1}{\lambda \pi p^{i+1}}.$$

Proof: We know that $d(u, \textit{parent}_p(u)) = l$ if there exists no node of $S_j, j \geq i+1$ inside the circle of radius l with center at u and a point of $S_j, j \geq i+1$ on the circumference. Hence, the expected distance to parent for a node u of level i is given by

$$\mathrm{E}[(d(u, \textit{parent}(u))^2 \mid lev_p(u) = i] = \int_0^\infty l^2 e^{-\lambda p^{i+1} \pi l^2} \lambda p^{i+1} 2\pi l dl = \frac{1}{\lambda \pi p^{i+1}}.$$

■

Hence the expected energy dissipated per unit of data sent by a node of S_i to its parent is $E_{elec} + \frac{\epsilon_{fs}}{\lambda \pi p^{i+1}}$. We observe that in the case of general values of the weight function there will be a larger number of nodes at each level and so the expectation of the square of the distance will, in general, be upper bounded by this quantity.

We now have a handle on how long a node can operate before its battery power falls by a factor of p, necessitating its demotion by 1 level. A simple calculation reveals that a node u with weight $w(u)$ and $\textit{lev}_p(u) = i$ can transmit

$$\ell_i = \frac{\lambda \pi w(u)(1-p)p^{i+1}}{\lambda \pi p^{i+1} E_{elec} + \epsilon_{fs}} \tag{1}$$

units of data in expectation before its weight falls to $w(u) \cdot p$. The quantity ℓ_i will be crucial in determining the transmission schedules of the nodes of the network to be discussed in Section III-A.

b) Number of children of a node: Each node $u \in S_i$ in $\mathsf{HN}_p^w(V)$ has a number of children from S_{i-1}. This number is a random variable depending on the placement of the nodes and the random processes by which the sets S_i are formed. Since the network operation involves nodes collecting data from their children, we will characterize the number of nodes that count a single node as their parent. In [3], as part of the proof of Theorem 2.1 we prove that the expected number of children of a node u with $\textit{lev}_p(u) = i$ is at most $6i(1-p)/p$.

C. Lifetime versus Delay

In this section we present analyses of the lifetime of the network and the delay it experiences, and demonstrate that, as expected, there is a tradeoff between these two criteria. This tradeoff can be measured using the parameter p.

c) Lifetime analysis: For wireless sensor networks deployed in remote locations recharging of sensor batteries is often impossible. A critical parameter in the analysis of topology control mechanisms used to collect data from is such settings is the network lifetime i.e. the time before the node's battery power falls to a level at which the node is no longer functional. In this paper we measure the lifetime not in actual units of time but in terms of rounds of data sent. We assume that in every round one packet of data is generated by each sensor and count the number of such rounds which are communicated all the way to the base before the first node's battery power falls below a given threshold. In this section we provide an analytical view of the network lifetime, postponing a simulation study to Section IV.

In order to analyze the lifetime, we proceed as follows. We consider a node u with $\textit{lev}_p(u) = i$ and battery power $w(u)$, such that $\lfloor \log_{\frac{1}{p}} w(u)/c \rfloor \geq 1$ i.e. a node that is operational because it still has c/p units of battery power left. We then compute the number of *listen and forward* rounds it performs before going in for repair i.e. the number of *listen and forward* rounds it takes to lose a factor p of its battery life. Since the calculations of Section III-B show that the transmission cost is exponential in $1/p$ with the level in the exponent while the cost of receiving is polynomial in $1/p$ and the level of a node (proportional to the number of children), we focus here on the energy spent in transmission and neglect the energy spent in receiving packets. We only present the cases where unlimited aggregation is allowed. The limited aggregation case can be approached similarly but we omit it due to space considerations.

Let us consider a node u with initial battery power $w(u)$. This node lives through $t(u) = \log_{\frac{1}{p}} \lfloor \frac{w(u)}{c} \rfloor$ cycles before it goes defunct. Since we are in the unlimited aggregation case, u has to send only one packet per round. Hence, when u has level j, by (1), it can transmit data for ℓ_j rounds. Hence, conditioning on the initial level of u being i, we get that the lifetime of u, denoted $\textit{lifetime}_p^w(u)$, i.e. total number of rounds of that u can send before dying is $\sum_{j=i-t(u)}^{i} \ell_j$. To simplify calculations, we bound the denominator of ℓ_j above by $\lambda \pi p E_{elec} + \epsilon_{fs}$ and below by ϵ_{fs} and sum to get

$$\boxed{\frac{\lambda \pi w(u) p^{i-t(u)}}{\lambda \pi p^{i-t(u)} E_{elec} + \epsilon_{fs}} \cdot \left(1 - \frac{cp}{w(u)}\right) \leq \mathrm{E}(\textit{lifetime}_p^w(u) \mid \textit{lev}_p(u) = i) \leq \frac{\lambda \pi w(u) p^{i-t(u)}}{\lambda \pi p^i E_{elec} + \epsilon_{fs}} \cdot \left(1 - \frac{c}{w(u)}\right).} \tag{2}$$

Note that $i - t(u) > 0$ for all nodes since each node get an initial level of $t(u)$ before the probabilistic part of the promotion takes place. Hence, we see that the time the node dies is a generally increasing function of p, but, if the lower bound is tighter than the upper bound, may decrease after a certain value of p. And, in fact, it is this observation that is borne out by our simulations in Section IV.

d) Delay: The delay incurred in data gathering using hierarchical neighbor graphs is a function of the parameter p and the number of nodes in the network $|V|$. As transmission accounts for most of the delay we assume a unit delay cost for a single transmission. According to the protocol detailed in Section III-A, a TDMA schedule is created by a node and communicated to the children. In order to transmit, a node has to wait for its allocated time slot. This causes a delay equal to the number of children. Using DSSS, nodes which have a different parent are allowed to transmit in parallel, thereby reducing the delay. In order to calculate the expected delay cost we use the upper bound on the expected number of children from Theorem 2.1 i.e. $6i(1-p)/p$.

When a packet is transmitted, a delay is incurred at each level of the hierarchy. Assuming N nodes in the network and observing that the probability that the network has more than $2\log_{\frac{1}{p}} N$ levels is very low ($O(1/N)$), we sum over the levels to get an expected delay cost of at most

$$\boxed{\sum_{i=1}^{2\log_{\frac{1}{p}}(N)} 6i \cdot \frac{1-p}{p} + 1 = \frac{1-p}{p} \cdot \alpha \log^2_{\frac{1}{p}}(N),} \tag{3}$$

for an appropriately chosen constant α.

The tradeoff : Note also that the delay cost for hierarchical neighbor graphs increases with the parameter p. Hence, since we want to minimize delay and maximize lifetime and they both increase with the parameter p, there is a tradeoff and so the parameter p has to be fixed at an optimal point. We attempt to determine this optimal point through simulations in the next section.

IV. SIMULATION STUDY

In this section we perform extensive simulation studies of hierarchical neighbor graphs. Network lifetime is the main metric we use to evaluate our proposal and we begin by investigating the sensitivity of this metric to various parameters, especially the parameter p, and study the tradeoff between lifetime and delay. We demonstrate that it is possible to identify an optimal value of p where minimizes the ratio of delay and lifetime. We also compare HNGs against LEACH, the energy aware version of LEACH, PEGASIS and H-PEGASIS. We show through simulation that the lifetime of HNG is better than these proposals and has the special property that it can handle heterogeneity in initial battery power much better than these other mechanisms. We also show that HNGs fare well when compared against these proposals on the product of delay and energy spent per unit data transmitted during the network's life, a metric that also seeks to optimize both lifetime and delay simultaneously [10].

A. Simulation setup and parameters

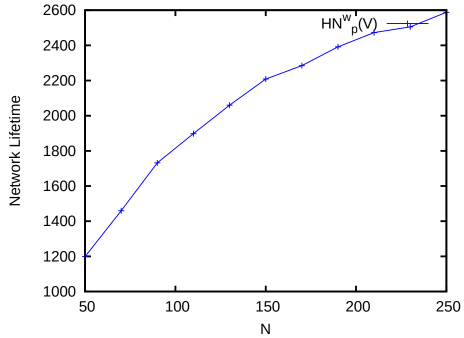
We initialized a network with a set V with $|V| = N = 100$ sensor nodes spread uniformly over a square region of side 100 units. The minimum energy required for a node to be operational was taken to be $0.1J$. The BS was located outside the square region at $(50, 300)$ and packet size was taken to be 2000 bits. These settings are taken from [7] to facilitate comparison with LEACH and the other architectures we consider. For radio transmission we assumed that to transmit l bits of data over a distance d, the energy dissipated is $E_{Tx}(l,d) = lE_{elec} + l\epsilon_{fs}d^2$ where the radio electronics energy $E_{elec} = 50\ nJ/bit$, depends on the coding and spreading of the signal and the amplifier constant $\epsilon_{fs} = 100\ pJ/bit/m^2$ depends on the acceptable bit-error rate. We assumed the energy consumed in receiving an l-bit message is $E_{Rx}(l) = lE_{elec}$. Additionally the energy consumed for data aggregation was taken to be $E_{DA} = 5\ nJ/bit/signal$.

Our model for delay is simple. We assuming a unit delay for each transmission. Under this assumption, as shown in Section III-C the delay of the network depends only on p and the number of nodes N. We consider also two different data aggregation models corresponding to different application. *(a) Limited Aggregation:* Only data signals from nodes located close to each other are highly correlated and can be aggregated into a single signal. Since nodes which share a common parent in $\mathsf{HN}_p^w(V)$ are located close to each other, we assume their data signals are correlated and can be fused into a single signal. *(b) Unlimited Aggregation:* All data signals, irrespective of location can be fused to get a single signal. This model is valid for applications in which we are interested in quantities like the average, min or max of a set of values. In this case, all data signals at a relaying node are fused into a single signal. Throughout this section the unlimited aggregation model is used unless stated otherwise. Also, unless otherwise stated we assume that the initial battery power of all nodes is the same.

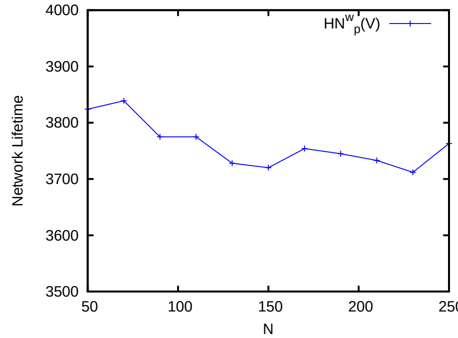
B. Sensitivity analysis for HNGs

Quantities like the density of the nodes, the initial battery power of the nodes and the values of the radio parameters are parameters of the system that are given to us. We first investigate the sensitivity of HNGs to these parameters and then move on to finding optimal settings of p which is in our control.

Network lifetime versus density: In order to examine change in network lifetime with network density we vary the number of nodes between 50 and 250 in steps of 20 for a fixed area. The constant p is fixed at 0.5 and the initial battery for each node equal to $1J$. Figure 2(a) shows that the network lifetime increases almost linearly with the network density. This is because the energy consumed in transmission decreases linearly with node density whereas the number of connections remains the same. In order to validate the bounds given in (2) we also simulated the network's performance under the assumption that the amplifier constant ϵ_{fs} was 0 and, as expected, we observed a more or less constant behavior (see Figure 2(b)).

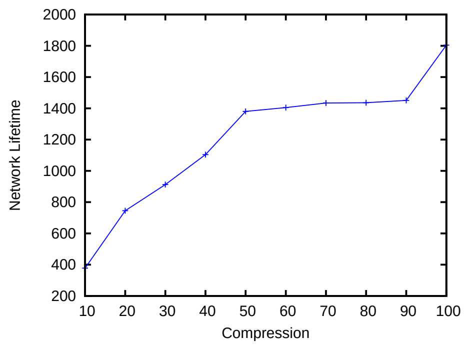


(a) $\epsilon_{fs} = 100pj/bit/m^2$.

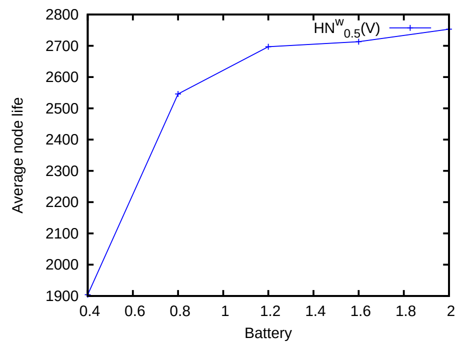


(b) $\epsilon_{fs} = 0$.

Fig. 2. Network lifetime vs. network density.



(a) Network lifetime vs. extent of aggregation.



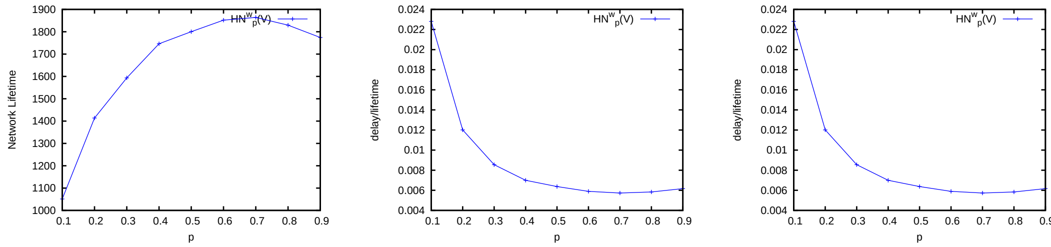
(b) Average node life vs. the initial residual energy.

Fig. 3.

Network lifetime versus aggregation: Since different applications allow different level of data aggregation we tested the performance of HNGs by varying the compression ratios from $10:1$ to $100:1$. Figure 3(a) shows the lifetime for $\mathsf{HN}^w_{0.5}(V)$. We see that the network lifetime increases steadily with the compression ratio, validating the intuition that greater compression allows for more efficient transmission, an intuition that should hold for any architecture..

Life of a node versus initial battery: From the lifetime inequalities given in (2) it is clear that the initial battery power is a crucial factor in determining how long an individual node can function. In order to study this we simulated a heterogeneous network in which each node was randomly assigned an initial energy from 5 levels between between $0J$ and $2J$ to see the dependence of the life of a node on its initial residual energy. In Figure 3(b) we see that the plot is concave i.e. higher battery powered nodes do not live much longer than lower battery powered nodes. This validates our claim that in HNGs nodes with higher battery power are made to do more work per round than nodes with lower battery power and hence the network uses its energy efficiently.

Network lifetime versus p: The variation of network lifetime–the number of rounds before the first node dies–with p was studied by simulating $\mathsf{HN}^w_p(V)$ for the set described above and varying p in steps of 0.1 between 0.1 and 0.9 (Figure 4(a)). We observed that the lifetime increases initially and then decreases, with a maxima at around 0.7 which makes us believe that the lower bound described in (2) is perhaps tighter than the upper bound. This point of maxima is, of course, a function of the radio parameters. We simulated $\mathsf{HN}^W_p(V)$ over the same range of values of p with three different values of the amplifier constant ϵ_{fs}, keeping the radio electronics energy E_{elec} fixed at $50nJ/bit$. We observed that the shape of the curve remains the same except the point of maxima shifts slightly and the maximum value is different. Those plots are omitted here due to space constraints.



(a) Network lifetime (in rounds) vs. p.

(b) delay versus p.

(c) delay/lifetime vs. p.

Fig. 4. Lifetime and delay vs. p.

Delay/lifetime versus p: In order to demonstrate that delay and lifetime can be simultaneously optimize for hierarchical neighbor graphs we computed the delay of $\mathsf{HN}_p^w(V)$ by running an algorithm that computed the maximum number of time slots it takes for a packet to get to the base station (see Figure 4(b)). The shape of the curve obtained validates the bound reported in (3), and we observe that there is a minima which happens to be around 0.7 here. We plotted delay/lifetime as well and found that it too has a definite minima, also seen around 0.7 (see Figure 4(c)). Note that there is nothing special about the value 0.7. The optimal point for delay/lifetime shifts around based on the relative value of the radio parameters. The key observation is that for all our simulations there was an optimal point for the delay/lifetime function.

C. Comparison with competing proposals

As mentioned earlier there are several proposals for topology control mechanisms for collecting data in wireless sensor networks. Here we compare through simulation the performance of hierarchical neighbor graphs with the performance of two leading proposals, LEACH [6] and PEGASIS [11], and some of their variants.

Comparing lifetimes: We simulated HNGs, LEACH and PEGASIS and observed the number of nodes alive over time. Our simulation results for a network where all nodes start with the same energy value of **VALUE NEEDED.** are presented in Figure 5. HNGs were simulated with two values of the constant $p = 0.5, 0.7$, the latter value being the optimal value of p obtained for the given choice of radio parameters that we discussed in Section IV-B. We simulated two versions of LEACH, one being the baseline proposal that selects cluster heads uniformly and the other version, denoted LEACH$_2$ here, being the energy aware version that considers residual energy in choosing cluster heads. We observed that $\mathsf{HN}_{0.7}^w$ outperformed LEACH by 200% in terms of network lifetime, i.e. it has a network lifetime three times that of LEACH. For PEGASIS we observed that although its last node dies later that $\mathsf{HN}_{0.7}^w(V)$, it's first node dies significantly earlier. In fact by the time the first node dies for $\mathsf{HN}_{0.7}^w$ more than 10% of PEGASIS's nodes are dead.

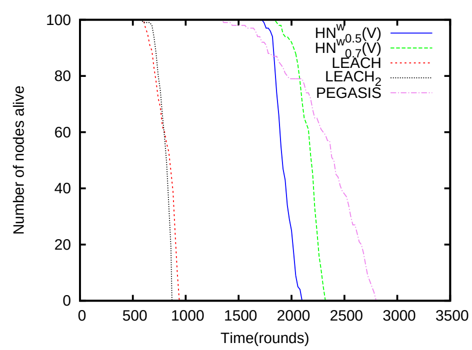


Fig. 5. Number of nodes alive over time for different architectures.

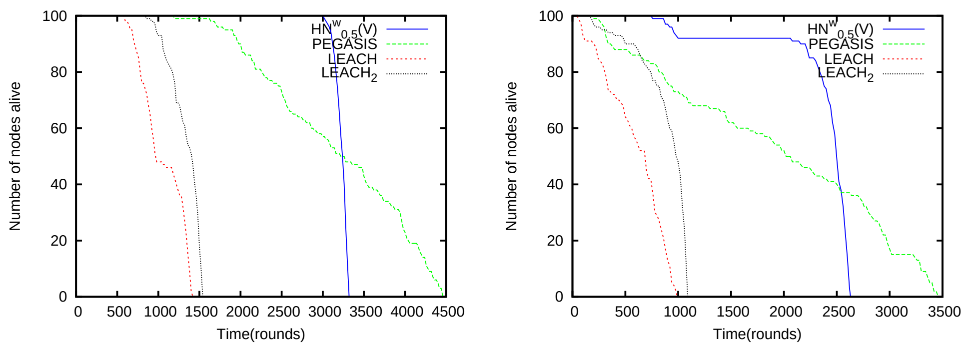
Lifetime and delay: Following the analysis of [10] we compare the performance of HNGs to that of LEACH and PEGASIS when delay and lifetime are considered simultaneously. This is done in [10] by using a metric they call $energy * delay$ where $energy$ is the average energy spent per round until the first node dies. We summarize the results in Table I including data from [10] for LEACH, PEGASIS and its hierarchical forms. It is clear that HNGs performs well in comparison with the other protocols. $\mathsf{HN}_{0.7}(V)$ is outperformed marginally by binary PEGASIS according to this metric because its delay is higher although it is more energy efficient.

Heterogeneous battery power: Hierarchical neighbor graphs are particularly well suited to situations where different nodes of the network begin with different battery power and consequently perform even better than other architectures than they do

Protocol	$energy$	$delay$	$energy * delay$
$\mathsf{HN}^{w}_{0.7}(V)$	0.048283	11	0.5311
LEACH	0.204786	27	5.5292
PEGASIS	0.036107	100	3.6107
Binary PEGASIS	0.055898	8	0.4516
3 level PEGASIS	0.058287	15	0.8743

TABLE I
$Energy$ PER ROUND AND $Delay$ FOR VARIOUS PROTOCOLS.

in the uniform initial battery power case. We simulates HNGs, LEACH, and PEGASIS for two heterogeneous networks having low and high variation in initial energy. In the first scenario a node is randomly assigned energy $1J$ or $2J$ with equal probability (*Heterogeneous Network 1*). In the second scenario a node is randomly assigned an initial energy between $0.1J$ and $2J$ from 10 levels (*Heterogeneous Network 2*). Figure 6 shows that the margin by which HNGs outperform other protocols increases with the heterogeneity in the network. For *Heterogeneous Network 1* the improvement over LEACH_2 was up to 250% from just under 200% for a homogeneous network. Note that this energy aware version of LEACH requires an estimate of the residual energy in the entire network and hence requires greater set-up energy cost which is not accounted for. In Figure 6(b) we notice that when approximately 90% of the nodes are alive in $\mathsf{HN}^{w}_{0.5}$, more than half of the nodes have died in all other protocols.



(a) Heterogeneous Network 1.

(b) Heterogeneous Network 2.

Fig. 6. Number of nodes alive over time using HNGs, LEACH, PEGASIS.

REFERENCES

[1] K. Akkaya and M. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3:325–349, 2005.

[2] F. Baccelli and S. Zuyev. Poisson-voronoi spanning trees - with applications to the optimization of communication networks. *Operations Research*, 47:619–631, 1997.

[3] A. Bagchi, A. Madan, and A. Premi. Hierarchical neighbor graphs: An energy-efficient bounded degree connected structure for wireless sensor networks. arXiv:0903.0742v3 [cs.NI], July 2009.

[4] P. Ballister, B. Bollobás, A. Sarkar, and M. Walters. Connectivity of random k-nearest-neighbour graphs. *Adv. Appl. Prob. (SGSA)*, 37:1–24, 2005.

[5] D. M. Blough, M. Leoncini, G. Resta, and P. Santi. The k-neigh protocol for symmetric topology control in ad hoc networks. In *Proc. 4th Intl. Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003)*, pages 141–152, 2003.

[6] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans. Wireless Commun.*, 1(4):660–670, 2002.

[7] W. B. Heinzelmann, A. P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans. Wireless Commun.*, 1(4):660–670, 2002.

[8] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Trans. Netw.*, 11(1):2–16, 2003.

[9] M. A. Labrador and P. M. Wightman. *Topology control in wireless sensor networks*. Springer, 2009.

[10] S. Lindsey, C. Raghavendra, and K. Sivalingam. Data gathering in sensor networks using the energy*delay metric. In *In Proc. of IPDPS Workshop on Issues in Wireless Networks and Mobile Computing*, pages 924–935, 2001.

[11] S. Lindsey and C. S. Raghavendra. Pegasis: Power-efficient gathering in sensor information systems. In *Aerospace Conference Proceedings, 2002. IEEE*, volume 3, pages 3–1125–3–1130 vol.3, 2002.

[12] A. Manjeshwar and D. P. Agrawal. Teen: a routing protocol for enhanced efficiency in wireless sensor networks. In *Parallel and Distributed Processing Symposium., Proceedings 15th International*, pages 2009–2015, 2001.

[13] A. Manjeshwar and D. P. Agrawal. Apteen: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In *IPDPS '02: Proceedings of the 16th International Parallel and Distributed Processing Symposium*, 2002.

[14] R. Rajagopalan and P. Varshney. Data-aggregation techniques in sensor networks: A survey. *IEEE Commun. Surveys Tuts.*, 8(4):48–63, 2006.

[15] P. Santi. *Topology Control in Wireless Ad Hoc and Sensor Networks*. Wiley, 2005.

[16] Y.-J. H. Sung-Min Jung and T.-M. Chung. The concentric clustering scheme for efficient energy consumption in the pegasis. In *Proc. 9th Intl Conf. on Advanced Communication Technology,*, volume 1, pages 260–265, 2007.
[17] N. Tabassum, Q. E. K. Mamun, and Y. Urano. COSEN : A chain oriented sensor network for efficient data collection. In *Proc. 3rd Intl. Conf. on Information Technology : New Generations (ITNG '06)*, pages 262–267, 2006.
[18] H. O. Tan and I. Körpeoğlu. Power efficient data gathering and aggregation in wireless sensor networks. *SIGMOD Rec.*, 32(4):66–71, 2003.
[19] M. Xiao, G. Chen, and D. Xiao. A policy-based energy efficient clustering scheme for wireless sensor networks. In *Proc. 8th ACIS Conf. on Software Engineering, AI, Networking and Parallel/Distributed Computing*, pages 689–694, 2007.
[20] F. Xue and P. R. Kumar. The number of neighbors needed for the connectivity of wireless networks. *Wirel. Netw.*, 10:169–181, 2004.
[21] O. Younis and S. Fahmy. HEED: A hybrid energy-efficient distributed clustering approach for ad hoc sensor networks. *IEEE Trans. Mob. Comput.*, 3(4):366–379, 2004.