Wireless Netw (2007) 13:679–690
DOI 10.1007/s11276-006-8149-y

Energy efficient routing with delay guarantee for sensor networks*

Sinem Coleri Ergen · Pravin Varaiya

Abstract The paper presents a routing algorithm that maximizes the lifetime of a sensor network in which all data packets are destined for a single collection node. Lifetime is maximized by adjusting the number of packets traversing each node. The adjustment is carried out by transmitting over alternative routes. The first part of the paper assumes that the worst case delay resulting from energy efficient routing is less than the maximum tolerable value. Ignoring the delay constraint of the network, the routes are selected as the solution to a linear programming (LP) problem in which the objective is to maximize the minimum lifetime of each node. The solution is implemented in a centralized algorithm, and then approximated by an iterative algorithm based on least cost path routing, in which each step is implemented efficiently in a distributed manner. The second part of the paper incorporates delay guarantee into energy efficient routing by constraining the length of the routing paths from each sensor node to the collection node. Simulations reveal that the lifetime of the network increases significantly by optimal routing, and including delay constraint in energy efficient routing improves the network performance since the delay of the network keeps increasing as the delay constraint is relaxed beyond the value at which the optimal lifetime is achieved.

Keywords Sensor networks · Routing · Energy efficiency · Delay guarantee · Linear programming

1. Introduction

A wireless sensor network consists of a group of nodes, each comprising one or more sensors, a processor, a radio and a battery. Such sensor networks are expected to find widespread use in such applications as traffic monitoring on freeways or urban street intersections, seismic and medical data-gathering because of their low cost, small size and wireless data transfer [1].

A key concern in wireless sensor networks is energy efficiency. The nodes in a sensor network may not be charged once their energy is drained so the lifetime of the network depends critically on the energy conservation mechanism. As shown in [2], most of the battery energy is consumed by the radio. A Time Division Multiple Access (TDMA) scheme at the medium access control (MAC) layer decreases this energy consumption by avoiding overhearing other nodes' packets, idle listening and collisions. However, when MAC protocol power savings are not combined with power-efficient routing, there will be high variations in energy spent in transmission and reception of packets at different nodes due to the different number of packets that each node needs to forward. This will cause some nodes to die earlier, which will decrease the efficiency of the network. Throughout the rest of the paper, we assume that the MAC protocol completely eliminates overhearing, idle listening and collision problems and we focus on adjusting the number of packets forwarded by each node through the choice of different paths in the network. It is straightforward to make changes in the formulations based on different MAC protocol capabilities.

A wireless sensor network is a special class of ad hoc networks, in which every packet is destined for the same collection node. Many routing protocols have been developed for general ad hoc wireless networks to provide correct and efficient route establishment between the nodes in the

---

*Research supported by National Science Foundation under Grant CMS-0408627 and California Department of Transportation.

S.C. Ergen (✉) · P. Varaiya
Department of Electrical Engineering and Computer Sciences,
University of California, Berkeley, CA 94720, USA
e-mail: {csinem, varaiya}@eecs.berkeley.edu

network so that messages can be delivered in a timely manner during mobility and changing topology [4]. Shortest hop routing is most common in table-driven protocols such as DSDV (Destination-Sequenced Distance-Vector) and WRP (Wireless Routing Protocol), and source-initiated protocols such as AODV (Ad Hoc On-Demand Distance Vector) and DSR (Dynamic Source Routing). Routing based on shortest hop, however, is not suitable for sensor networks where there are many flows towards one access point and the elimination of a node may disconnect a large number of nodes from the access point.

Various power-aware metrics are discussed in [5] to find the traffic distribution that balances the energy consumption optimally. These metrics include maximizing the time to network partition, minimizing the variance in node power levels, minimizing cost per packet and minimizing maximum node cost. Maximizing the time during which all nodes are alive is formulated as a Linear Programming (LP) problem in [6, 7]. None of these schemes, however, propose a routing protocol that uses this formulation. The effect of minimizing the cost per packet is investigated for different cost functions in [5, 6, 8, 9, 10]. The cost depends on the ratio of the initial battery energy to remaining battery energy in [5, 6, 10] or on the ratio of the rate of energy consumption to the remaining energy of the node in [8, 9]. The relation of maximizing the minimum lifetime of the nodes to minimizing the cost per packet was investigated in [6]. We take this relation one step further to provide a delay guarantee on the time the packets reach their destination, while maximizing the network lifetime.

The first part of this paper formulates the lifetime maximization problem as a linear programming (LP) problem, and proposes a routing protocol for centralized implementation of the LP solution. The centralized protocol is based on decomposing the LP solution into multiple routing trees. A distributed routing protocol is then proposed to implement this decomposition by a sequence of least cost path problems, in which the cost of a path is the sum or the maximum of the cost of the nodes on that path and the cost of each node is a function of its initial and remaining battery energy. This distributed implementation provides a platform for simultaneously achieving energy efficient routing and delay guarantee since the delay guarantee cannot be introduced into LP formulation before decomposition.

The second part of the paper extends the energy efficient routing to provide a guarantee on the maximum delay each packet experiences. Minimizing energy consumption subject to a deadline or delay constraint has been introduced in [11], which proposes an algorithm for scheduling packet transmissions over a wireless link based on the observation that in many channel coding schemes the energy required to transmit a packet can be significantly reduced by transmitting the packet over a longer period of time. In this paper, we generalize this problem to a multi-hop network. We assume that the transmission rate is fixed and limit the number of hops each packet experiences in the network to provide a guarantee on the worst case delay. We believe this is the first work that considers energy efficient routing with delay guarantee in multi-hop networks.

The rest of the paper is organized as follows: Section 2 presents the assumptions necessary for formulating the problem. Section 3 gives the LP formulation of maximizing the network lifetime and introduces a routing algorithm based on the LP solution. Section 4 describes a distributed algorithm based on least cost path routing that aims to achieve optimal lifetime. Section 5 proposes routing protocols that incorporate delay into the energy efficient routing with the goal of providing a delay guarantee. Section 6 analyzes the memory and CPU requirements at the sensor nodes for the implementation of these algorithms. Simulation results are in Section 7. Section 8 presents concluding remarks.

2. Assumptions

The following assumptions underly this study.

1. Consider a wireless ad hoc network that consists of one access point (AP) and several sensor nodes that generate data for transfer to the AP.
2. The transmission power of all the sensor nodes is the same across the network. Each node is supported by an omnidirectional antenna, so links are bidirectional: If two nodes i and j transmit at the same power, then if node i can hear node j, node j can also hear node i. Bidirectional links are needed for the proper functioning of some network protocols such as distributed Bellman-Ford algorithms [12].
3. The topology of the network is represented by a graph $G = (V, E)$, in which V is the set of nodes, including the AP as node 1. The edges $E \subset V \times V$ are such that $(i, j) \in E$ if nodes i and j can transmit to each other. N and $|V|$ are used interchangeably to denote the number of nodes in the network including AP.
4. The nodes in the network generate data at a specific rate, which may be different for each node. The LP formulation in Section 3 requires the estimation of these rates at the beginning whereas they can be learnt iteratively for the iterative routing algorithms in Section 4.
5. Data generated at a node is independent of the transmission structure. The data aggregation models where the amount of data supplied at nodes does depend on the routing paths [13] is a much harder problem and beyond the scope of this paper.
6. Power is consumed in the radio, sensor and microprocessor. We assume that the MAC protocol operates the radio in sleep mode when the node is not transmitting or receiving a packet.

7. The operational lifetime of the sensor network is defined to be the time until the first node dies since sensor network monitoring can be impaired by the early death of some nodes and the possible disconnectedness of some other nodes from AP as a result. The relation of this lifetime estimate to the time duration until different fractions of the nodes die is given in Section 7.

3. Centralized energy efficient routing

The objective of the routing algorithm is to determine the optimal path from each sensor node to the AP based on the topology of the network and the packet generation rates at the sensor nodes. Optimizing the routing paths will assign packet flow rate f_{ij} from node i to node j for $i, j \in V$. As long as these optimal flow rates are non-zero for at least two flows outgoing from a node in the network, choosing the same routing path all the time will not be optimal. In this case, the next hop for node $i \in V$ has to alternate among nodes $j \in V$ for which $f_{ij} \neq 0$. Notice that this is not multiple path routing since at any given time, there is still a single path from each sensor node to the AP, but this path may change over time.

Section 3.1 describes the LP formulation for determining optimal flow rates. Section 3.2 proposes a routing protocol based on the centralized calculation of these optimal values through the LP formulation.

3.1. Linear programming formulation

The optimization problem is given in Fig. 1. The variables of the problem are the packet flow rates f_{ij}, which is the average time spent for the transmission of the packets from node i to node j per unit time, and the network lifetime t. The goal is to maximize t.

The first constraint represents the non-negativity constraint of flows, whereas the second constraint eliminates the possibility of flows between nodes without a communication link. The third constraint requires that the net flow out of each node i be the time required to transmit the packets generated in that node $g_i tt$, where tt is the transmission time of one packet and g_i is the packet generation rate at node i, except AP. The fourth constraint requires that the total energy consumed by node i during the network lifetime, which is the product of the network lifetime t and the energy consumed per unit time, be less than its total energy e_i. The total energy consumed includes energy spent in transmission and reception of packets, listening to the channel and sensing. The energy spent in transmission and reception is $\Sigma_j p_{tx} f_{ij}$ and $\Sigma_j p_{rx} f_{ji}$ respectively, where p_{tx} is the energy spent in the transmission of a packet in unit time and p_{rx} is the energy spent in the reception of a packet in unit time. The energy spent in listening to the channel is then $(1 - \Sigma_j f_{ij} - \Sigma_j f_{ji})p_l$, where p_l is the energy spent per unit time by the radio in sleep mode since the radio is assumed to sleep by MAC protocol if there is no transmission or reception of packets. Finally, the energy spent in sensing is $p_s g_i$, where p_s is the energy spent in obtaining the samples in one packet.

Maximize t
Subject to: $f_{ij} >= 0$ for $i, j \in [1, N]$
$f_{ij} = 0$ for $(i, j) \notin E$
$\Sigma_j f_{ij} - \Sigma_j f_{ji} = g_i\ tt$ for $i \in [2, N]$
$t(\Sigma_j p_{tx} f_{ij} + \Sigma_j p_{rx} f_{ji} + p_s g_i + (1 - \Sigma_j f_{ij} - \Sigma_j f_{ji}) p_l) <= e_i$ for $i \in [2, N]$

Fig. 1 Linear Programming model for determining optimal flows along each link

This is an LP problem since all the constraints are linear in the variables when we replace variables f_{ij} by tf_{ij}.

In order to use this algorithm in a practical framework, a routing protocol is needed to decompose the network with optimal flow rates $f_{ij}, i, j \in V$ into multiple routing trees. Then the nodes in the network will be scheduled by a TDMA protocol at the MAC layer such that all packets reach the AP by a deadline. Since any routing tree may result from this decomposition, the deadline should be achieved for all possible such trees.

Let us assume that the time is divided into time slots. Each time slot is long enough to carry one packet. The TDMA scheduling algorithm guarantees the transmission of at least one packet in each time slot. The necessary and sufficient condition for the network to meet the delay constraint $delay_{max}$ for all possible routing trees is $tt(|V|\Sigma_{i \in V} g_i) \leq delay_{max}$. This is equivalent to sending all packets generated in the network on the path containing maximum possible number of hops, which is $|V|$. This assumption is not stated for the similar formulation described in [6] but it is implied by the LP formulation described above. Capacity constraints on the nodes such as $\Sigma_j f_{ij} + \Sigma_j f_{ji} \leq delay_{max}$ alone is not enough since it takes time for the flows to be transferred across the network. The problem of adjusting the energy efficient routing paths to provide a delay guarantee in the network is examined in Section 5.

3.2. Routing protocol

The LP approach provides optimal flow rates based on knowing the complete topology and packet generation rate at each node. We now provide a routing algorithm that generates a single path from any sensor node to the AP at each time and attains the optimal flow rates at each link at the end.

We first prove that a directed acyclic graph[1] attains the maximum lifetime while satisfying all constraints in the LP formulation in Fig. 1. We then give an algorithm to obtain such a directed acyclic graph solution and then to decompose it into directed-in-trees.[2] The optimal solution can then be attained by routing the packets over each of these directed-in-trees for the proportion of the battery lifetime of the network equal to its weight.

Theorem 1. *Consider the set of directed graphs consisting of arcs from node i to node j if and only if the optimal value of f_{ij} for LP problem in Fig. 1 is positive. The set includes a directed acyclic graph.*

Proof: We prove this theorem by providing an algorithm that takes a possibly cyclic graph and at each iteration eliminates one cycle in the graph by subtracting the minimum flow rate in the cycle from all the flow rates of the cycle without decreasing the lifetime of the network while satisfying the constraints. □

After obtaining the solution to the LP problem, the directed simple cycles in the graph can be found by using an extension of Depth-First Search (DFS) algorithm [14]. The cycles can be eliminated by the method given in the proof of Theorem 1 to obtain a directed acyclic graph solution. This solution can then be expressed as a superposition of multiple directed-in-trees due to its acyclic property. We now prove that we can find such directed-in-trees that satisfy the constraints of the problem in Fig. 1. We first need the following lemmas.

Lemma 1. *An undirected graph $G = (V, T)$ is a tree if it has $|V| - 1$ edges and is acyclic.*

Proof: See [14] for the proof. □

Lemma 2. *Let $G^v = (V, E^v)$ be a directed acyclic graph such that $outdeg(i) = 1$ for $i \in \{2, 3, ..., |V|\}$ and $outdeg(1) = 0$. Then G^v is a directed-in-tree rooted at node 1.*

Proof: Let $G_u^v = (V, E_u^v)$ be an undirected graph such that $(i, j) \in E_u^v$ only if $(i, j) \in E^v$. G_u^v does not contain any cycle since otherwise the outdegree of at least one node in the cycle must be 2 in G^v in order not to form a directed cycle. Besides, G_u^v has $|V| - 1$ edges. Thus, it is a tree by Lemma 1. If G^v is a directed-in-tree, the root must be node 1 since $outdeg(1) = 0$. Let us choose node 1 as the root of G_u^v. We can then prove that the head of each arc in G^v is closer to the node 1 than its tail by contradiction. □

For the directed acyclic graph solution to the LP problem shown in Fig. 1, let $f_{ij}, i, j \in V$ be the optimal flow rate from node i to node j, $E^+ = \{(i, j) | f_{ij} > 0\}$, $G = (V, E^+)$. Denote optimal lifetime by t_o. G is decomposed into multiple routing trees as follows:

1. Choose one outgoing link from each node $i \in V$ such that $f_{ij} > 0$ and use the single path from each sensor node to the AP on the resulting directed-in-tree rooted at the AP as the routing path.
2. If the packet flow up to now along at least one link $(i, j) \in E^+$ on the tree is greater than $f_{ij} t_o$, set $f_{ij} = 0$.
3. If $\Sigma_{i,j} f_{ij} = 0$, stop. Else, go back to step 1.

Choosing one outgoing link at each node in step 1 guarantees that the resulting graph is a directed-in-tree by Lemma 2.

The routing protocol is given as follows. Assume that the AP has enough processing energy whereas the other nodes are energy limited. At the beginning, the nodes determine their parent in the shortest path tree (SPT), i.e. tree that contains minimum-hop path from each node to the AP, in order to send their topology information to the AP. This can be performed by flooding a packet from the AP to the nodes, which then choose their parent to be the node sending the packet over least number of hops. During this flooding, the nodes also determine the identity of their neighbors, nodes with which direct communication can be established. They then send their neighbor information to the AP over SPT (see [2] for a detailed topology discovery protocol). AP runs the LP problem, and sends the optimal flow rates of the directed acyclic graph and the optimal lifetime to all the nodes in the network. A node i then chooses one of the outgoing links at each time to forward the packets in the network as long as the total packet flow at each link (i, j) is less than $f_{ij} t_o$.

4. Distributed energy efficient routing

In the previous section, we used an LP formulation in order to determine optimal routing paths from each node in the network to the AP to maximize the system lifetime. Solving the LP problem, however, requires knowing the whole network topology and packet generation rate at each node, and too many computations to be implemented at a sensor node. In this section, we provide an algorithm that approximates the LP solution through successive more efficiently solved minimum cost path problems. This approximation will also be useful in incorporating delay constraints into the routing protocol in Section 5.

---

[1] A directed acyclic graph is a directed graph where no path starts and ends at the same vertex.

[2] A directed-in-tree rooted at node s is a tree where there is a unique directed path from any node to the node s. Every node in the directed-in-tree (except node s) has outdegree 1.

The goal of the distributed energy efficient routing is to get as close as possible to the optimal LP lifetime while periodically executing a distributed routing algorithm based on the remaining and initial energy of each node. We present two distributed routing algorithms, namely *least sum-cost path algorithm* and *least max-cost path algorithm*, to be implemented for each period. These algorithms aim at minimizing the cost of the routing paths from each sensor node to the AP, where the cost of the path is defined as the sum or maximum of the costs of the links on that path. They are closely related to previous work on minimizing the cost per packet in [5, 6, 8, 9, 10].

Recall that the topology of the network is represented by the graph $G = (V, E)$ where $(i, j) \in E$ if nodes i and j can communicate with each other. Define the directed graph equivalent of G by $G_d = (V, E_d)$ where $\langle i, j\rangle \in E_d$ and $\langle j, i\rangle \in E_d$ for each $(i, j) \in E$. The iterative algorithm consists of obtaining a directed graph with a cost C_{ij} assigned to every link $\langle i, j\rangle \in E_d$ and then finding the shortest path tree from AP to all the nodes in that graph at the beginning of each time frame to be used for routing until the end of that frame.

Since the cost of the links is used in finding the shortest path from AP to the sensor nodes, the cost of directed link $\langle j, i\rangle$, C_{ji}, is equal to the cost of including node i on the path. A cost function for node i at p-th iteration is defined to be the ratio of the total energy consumed up to period p over the total battery energy:

$$C_i = \frac{\text{total consumed energy}}{\text{total battery energy}} \tag{1}$$

$$= p * \frac{\Sigma_j p_{tx} f_{ij} + \Sigma_j p_{rx} f_{ji} + p_s g_i + (1 - \Sigma_j f_{ij} - \Sigma_j f_{ji}) p_l}{e_i} \tag{2}$$

where f_{ij} is the average resulting flow rate on link $\langle i, j\rangle$. This node cost function increases from value 0 to value 1 as network evolves. The link cost function can be any monotonically increasing function d of C_i over interval (0,1) such that $C_{ji} = d(C_i)$. For instance, the cost function used in [6] is the ratio of total battery energy to the remaining energy, which corresponds to $d(x) = \frac{1}{1-x}$.

The next step is to calculate the shortest path from node 1 (AP) to all other nodes in the network where the cost of the path P from node i to node j is the sum and maximum of the costs of the links in P for least sum-cost path algorithm and least max-cost path algorithm respectively. The Bellman-Ford algorithm can be used to calculate this tree for both of these algorithms while requiring a slight change for least max-cost path algorithms, which is changing the action at each pulse from $\forall\langle i, j\rangle \in E_d$, if $ct(j) > ct(i) + C_{ij}$, $ct(j) = ct(i) + C_{ij}$ and $pred(j) = i$ to $\forall\langle i, j\rangle \in E_d$, if $ct(j) > max(ct(i), C_{ij})$, $ct(j) = max(ct(i), C_{ij})$ and $pred(j) = i$, in which $ct(i)$ is the cost of node i. The proof of the correctness of this algorithm is analogous to that of Bellman-Ford algorithm and the complexity remains the same. The stopping condition is the death of a node $i \in V - \{1\}$.

The routing protocol is as follows. Time is divided into time frames. At the beginning of a time frame, each node $i \in V$ calculates its *node cost* C_i as described in Eq. (1) to be used in the calculation of C_{ji}, and initializes its *cost* $ct(i)$ to ∞. (The cost of the AP $ct(1)$ is initialized to 0.) The AP floods the network with a tree construction packet. This packet contains the *node cost* and *cost* of the transmitting node in the routing tree. Upon receiving a tree construction packet, the node checks whether the transmitting node is the next hop on a path of smaller cost than previously learned paths. This is done by checking the condition $ct(j) > ct(i) + C_{ij}$ $(ct(j) > max(ct(i), C_{ij}))$ in least sum-cost (max-cost) path algorithm, in which j is the receiving node and i is the transmitting node. If the condition holds, the receiving node updates this *cost* by $ct(j) = ct(i) + C_{ij}$ $(ct(j) = max(ct(i), C_{ij}))$ in least sum-cost (max-cost) path algorithm, and rebroadcasts the packet. At the end of the flooding, each node chooses its parent node to be the next hop neighbor on the least cost path to the AP, and uses this next hop until the end of the time frame.

Simulation results in Section 7 show the effect of different cost functions for least sum-cost path algorithms and their comparison to least max-cost path algorithm and LP solution.

5. Energy efficient routing with delay guarantee

Energy efficient routing may choose paths that are much longer than the shortest paths to the AP while avoiding nodes with small residual energy. Longer paths may prevent the system from meeting the delay guarantee. This section aims to give a delay guarantee on the arrival of packets at the AP while generating energy efficient paths.

One way to introduce a delay constraint in the problem formulated in Fig. 1 is to upper bound the total flow rate by the delay constraint $delay_{max}$:

$$\sum_i \sum_j f_{ij} \leq delay_{max} \tag{3}$$

To understand whether the flow rates optimal for the optimization problem satisfy the delay constraint, we have to understand the routing algorithm in a practical framework. The network with optimal flow rates f_{ij} assigned to each link $(i, j) \in E$ needs to be decomposed into multiple routing trees. At each time, the network will use one of these trees, hence a single path from each sensor node to the AP. However, guaranteeing the average delay does not tell anything

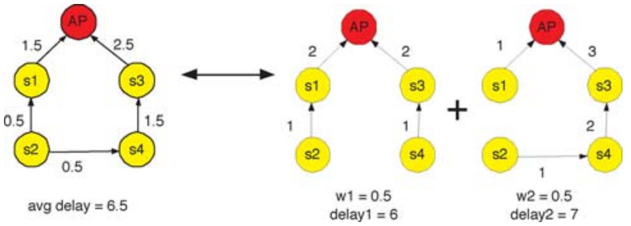

Fig. 2 Delay performance of the routing trees resulting from the decomposition of the optimal flow graph

about the delay at a specific time. Figure 2 illustrates this problem. The optimal flow rates are shown on the left whereas the decomposition is shown on the right. If all the nodes interfere with each other, then the delay is 7 time slots for one-half of the network lifetime, even though the average delay satisfies the delay constraint of 6.5 time slots.

We now introduce delay constraint on the decomposed network based on the distributed implementation described in Section 4. We assume that the transmission rate is fixed and limit the number of hops each packet experiences in the network to provide a guarantee on the worst case delay.

In order to bound the maximum delay, the MAC protocol should be based on TDMA such as PEDAMACS [2] instead of random access. In [3], it is shown that the problem of finding a schedule, which minimizes the maximum delay of the packets to reach the AP, is NP-complete. The problem of finding routes from each node to the AP such that the MAC layer minimizes the maximum length of the schedule is even harder. Therefore, we aim to guarantee the worst case delay while generating energy efficient routes.

We assume that the MAC protocol is TDMA and guarantees the transmission of at least one packet in each time slot. We first generate a distributed algorithm based on an upper bound on the worst case delay in Section 5.1. Then we show how to improve the performance in terms of energy efficiency and connectivity by using a centralized controller to help the nodes choose one of the paths available at each node based on the worst case delay in Section 5.2.

5.1. Level restricted energy efficient routing (LR-ENR)

The distributed Bellman-Ford algorithm is used to provide a guarantee on the delay by limiting the number of iterations. Bellman-Ford algorithm is known to find the minimum cost path of length at most i at iteration i [14]. The number of iterations is found by an upper bound on the worst case delay.

The worst case delay in the network is $tt\Sigma_{i\in V} l_i g_i$ where l_i is the length of the path from node i to the AP, g_i is the packet generation rate at node i and tt is the length of a time slot that includes the transmission time of a packet and a guard interval ($l_1 = 0$ since node 1 is AP). This delay is upper bounded by $d * g * tt$ where d is the maximum of the lengths of the routing paths from each node to the AP and g is the total number of packets generated in the network.

Given a delay requirement $delay_{max}$, we therefore calculate the maximum path length d_{max}:

$$d_{max} = argmax_d d * g * tt \leq delay_{max} \tag{4}$$

A modified version of Bellman-Ford algorithm can then be executed to find minimum cost paths of length at most d_{max} from each node to the AP. Instead of running each iteration of Bellman-Ford pulse for $|V|$ times and just keeping the parent in the routing tree, the modified version runs each pulse for d_{max} times and keeps the minimum cost path from each node to the AP.

The routing protocol is therefore the same as the distributed implementation described in Section 4 except for the inclusion of a counter and the complete routing path in the tree construction packet. The time is divided into frames. At the beginning of the frame, the AP floods the network with a tree construction packet. The tree construction packet keeps a *counter* c and the nodes on the routing path it followed starting at the AP in addition to the *node cost* and *cost* of the transmitting node in the routing tree. c is initialized to 0 at the AP and increased by 1 at each transmission. Upon reception of a tree construction packet, the node checks whether the transmitting node is the next hop on a path of smaller cost than previously learned paths only if this counter c is less than d_{max} and ignores the packet otherwise. If $c < d_{max}$ and a smaller cost path is found, the node updates the *cost* in the packet, increases the *counter* c by 1, adds its ID to the routing path the packet followed, and rebroadcasts the packet. At the end of the flooding, each node chooses the minimum cost routing path, and uses this path until the end of the time frame.

5.2. Hop restricted energy efficient routing (HR-ENR)

We can achieve higher lifetime and connectivity for a certain delay constraint by using a centralized controller based on the exact worst case delay instead of an upper bound on this delay as in Section 5.1. This is achieved by running the Bellman-Ford iterations $|V|$ times and then optimizing the path lengths based on the cost of the paths at each iteration.

If the vector containing the length of the routing paths, $l = [l_1, \ldots, l_{|V|}]$, was given, the minimum cost paths would be found by running Bellman-Ford algorithm for the number of times equal to l_j for node $j \in V$. In this problem, on the other hand, the delay requirement $delay_{max}$ is given. This restricts the lengths of the paths such that $tt\Sigma_{i\in V} l_i g_i \leq delay_{max}$. The problem is then to find the optimal vector $l = [l_1, \ldots, l_{|V|}]$ that minimizes the total cost while satisfying $tt\Sigma_{i\in V} l_i g_i \leq delay_{max}$.

Minimize $\Sigma_{i=2}^{N}\Sigma_{j=1}^{N}A_{ij}x_{ij}$

Subject to: $x_{ij} \geq 0$ for $i \in [2, N], j \in [1, N]$, x_{ij} integer

$\Sigma_{j=1}^{N}x_{ij} = 1$ for $i \in [2, N]$

$tt\Sigma_{i=2}^{N}\Sigma_{j=1}^{N}jg_ix_{ij} \leq delay_{max}$

Fig. 3 Integer Programming model of path length optimization

Figure 3 gives the IP formulation for choosing the optimal length of the paths based on the costs determined by Bellman-Ford algorithm. The variables of the problem are x_{ij}, which is 1 if the length of the path chosen by node i is j and is zero otherwise. The input to the problem is the cost of the paths A_{ij} and the packet generation rates g_i, in which A_{ij} represents the cost of the minimum cost path of length at most j from node i to the AP and g_i represents the packet generation rate at node $i \in V$.

The goal of the problem is to minimize the total cost of the paths. The first constraint represents the non-negativity constraint of x_{ij} whereas the second constraint represents the requirement that only one of the paths be chosen by each node $i \in V$. The third constraint rewrites the delay constraint requirement $tt\Sigma_{i\in V}l_ig_i \leq delay_{max}$ in terms of the variables x_{ij}.

The routing protocol is as follows. The time is divided into time frames. At the beginning of the frame, the AP floods the network with a tree construction packet, which contains the counter, its routing path, node cost and the cost of the transmitting node in the routing tree. The first part of the algorithm is similar to the one described in Section 5.1. The only difference is that each node $i \in V$ keeps the minimum cost path of length at most l for all $1 \leq l \leq |V|$. Upon reception of a tree construction packet, the node checks whether the cost of the path is smaller than that of previously learned paths of the same length. If so, it updates its minimum cost path for that length, and rebroadcasts the packet. At the end of the flooding, each node knows about the minimum cost path of each length. They then send only the cost of the paths corresponding to each length $1 \leq l \leq |V|$, not the paths themselves, to the AP. The AP finds the optimal path length for each node based on the formulation in Figure 3 and sends it back to the nodes in the network. The nodes then use the routing path of the optimal length until the end of the frame.

6. Complexity analysis

Since each individual wireless sensor node is usually a small device with limited memory space and computational capability, it is important to understand the memory and CPU requirements of the implementation of these algorithms at the sensor nodes.

For the centralized algorithm in Section 3.2, the per node memory requirement is $O(deg_{max})$ whereas the CPU requirement is $O(deg_{max}|V|)$, where deg_{max} is the maximum degree of the nodes in G. Each node has to keep the ID of its neighbors, the optimal rates of the links to these neighbors and the optimal lifetime in the memory, which is of complexity $O(deg_{max})$. At the beginning, the nodes send and receive packets to find the shortest path to the AP and discover their neighbors. The maximum number of iterations is $|V|$. In each iteration, each node sends at most one packet and receives at most deg_{max} packets. Each packet contains the ID and cost of the transmitting node. The nodes then send their own topology information and forward their children's topology information to the AP. The maximum number of packets that is forwarded is $|V|$ whereas each packet contains the ID of each node's neighbors. Finally, the optimal flow rates on the links to all neighbors of the nodes and optimal lifetime are sent back to the nodes. Each of these steps requires $O(deg_{max}|V|)$ packets to send and receive, resulting in $O(deg_{max}|V|)$ CPU requirement.

For the distributed algorithm described in Section 4, the per node memory requirement is $O(1)$ whereas the CPU requirement is $O(deg_{max}|V|)$. Each node just needs to keep the parent in the minimum cost path to the AP and its cost in the memory, resulting in complexity of $O(1)$. The nodes send and receive packets to find the shortest path to the AP. Each packet contains the ID and cost of the transmitting node. Therefore, the CPU complexity is $O(deg_{max}|V|)$.

For the level restricted energy efficient routing algorithm described in Section 5.1, the per node memory requirement is $O(|V|)$ whereas the CPU requirement is $O(deg_{max}|V|^2)$. Each node needs to keep the cost of the minimum cost path and all the nodes on that path. The maximum path length is $|V|$ so the memory complexity is $O(|V|)$. The nodes send and receive packets to find the shortest path to the AP. Each packet contains the ID and cost of the transmitting node, the nodes on the path followed by the packet and the counter. The maximum number of nodes on a path is d_{max}. Therefore, the CPU complexity is $O(deg_{max}d_{max}^2)$, which is at maximum $O(deg_{max}|V|^2)$.

For the hop restricted energy efficient routing algorithm described in Section 5.2, the per node memory requirement is $O(|V|^2)$ whereas the CPU requirement is $O(deg_{max}|V|^2)$. Each node needs to keep the minimum cost path corresponding to each length l for $1 \leq l \leq |V|$. The maximum path length is $|V|$, which results in memory complexity of $O(|V|^2)$. The nodes send and receive tree construction packets to find the shortest path to the AP. The complexity of this step is $O(deg_{max}|V|^2)$. The nodes then send their costs and forward their children's cost information to the AP. The maximum number of packets that is forwarded is $|V|$ whereas each packet contains the cost of $|V|$ paths, resulting in a

Table 1 Memory and CPU requirements of the sensor nodes for algorithm implementations

Algorithm	Memory	CPU
Centralized	$O(deg_{max})$	$O(deg_{max}\|V\|)$
Distributed	$O(1)$	$O(deg_{max}\|V\|)$
LR-ENR	$O(\|V\|)$	$O(deg_{max}\|V\|^2)$
HR-ENR	$O(\|V\|^2)$	$O(deg_{max}\|V\|^2)$

Table 2 Power consumption of basic operations in Berkeley mica nodes

Operation	Power consumption
Transmitting one packet	$0.92mJ$
Receiving one packet	$0.69mJ$
Listening to channel	$29.71mJ/sec$
Operating radio in sleep mode	$15\mu J/sec$
Sampling sensor	$1.5\mu J$/sample

complexity of $O(|V|^2)$. Finally, the optimal path lengths are sent back to the nodes, resulting in a complexity of $O(|V|)$.

The complexity results are summarized in Table 1.

7. Simulation

The purpose of the simulation is to observe the increase in network battery lifetime as a result of route optimization and to examine the effect of different link cost functions and delay constraints on the network lifetime.

In the simulations, the nodes are randomly distributed in a circular area of radius 100 units unless otherwise stated. The transmission range is chosen to be slightly larger than the threshold necessary for connectivity of the network [15]. The graph $G = (V, E)$ is obtained by placing edges between nodes closer than this transmission range. The results discussed below are averages of the performance of ten different random configurations.

The energy consumed in transmission and reception of packets, listening to the channel and sampling are derived from the power consumption figures for the Berkeley mica nodes, given in Table 2. The transmission rate is 50 kbps. The packet generation rate g_i at each node i is $1/30$ per second, which is a typical value for traffic applications [3]. The battery power level e_i is chosen to be that of a pair of AA batteries, which can supply 2200mAh at 3V, for all nodes $i, i \in V - \{1\}$. The sampling rate is 128Hz at each node.

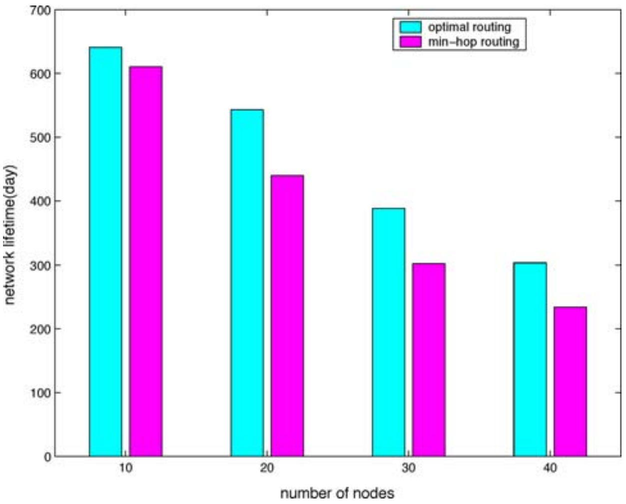

Fig. 4 Comparison of the lifetime for optimal and minimum hop routing

Figure 4 compares the lifetime of the network using the routing determined by the linear programming formulation in Fig. 1 with that using minimum hop routing. The optimization of the paths increases the network lifetime by 50-150 days. The energy spent for sampling the sensor is also included in lifetime calculations to get more realistic estimates. Ignoring [6] or decreasing the energy consumed in the sensor will further increase the effect of routing algorithm in the final lifetime estimates.

The goal of the iterative algorithm is to approximate the LP solution by dividing the complexity into multiple steps of minimum cost path problems. The parameters of the algorithm that affect the battery lifetime of the network are *the type of least cost path algorithm*, *step interval* and *cost function*. *Step interval* is defined to be the length of the time frame during which the same routing paths are used. *Cost function* is important in terms of emphasizing different battery levels at different intensities in least sum-cost path algorithm and is defined as $d(x)$ in Section 4.

Figure 5 gives the network lifetime for iterative algorithm. As the step interval decreases, the lifetime of the network increases due to the increasing number of iterations. For small enough step intervals, the lifetime is almost equal to the optimal lifetime for least max-cost path algorithm whereas it is very close to this optimal value for cost functions $d(x) = x^{50}$ and $d(x) = 1/(1-x)^{50}$ for least sum-cost path algorithm. This suggests that for cost function $d(x) = x^n$ in least sum-cost path algorithm, n should not be chosen too small since the cost function becomes almost linear and it may be hard to differentiate between a path containing one node with small residual energy and many nodes with high residual energy, from a path with many nodes of medium residual energy. Also, n should not be chosen too large because it is hard to differentiate between different $d(x) = x^n$ values unless x is very close to 1. On the other hand, as n gets larger in cost function $d(x) = 1/(1-x)^n$ for least sum-cost path algorithm, it gets closer to the optimal value since the cost of the path gets closer to taking the maximum of the costs on the path in that case.

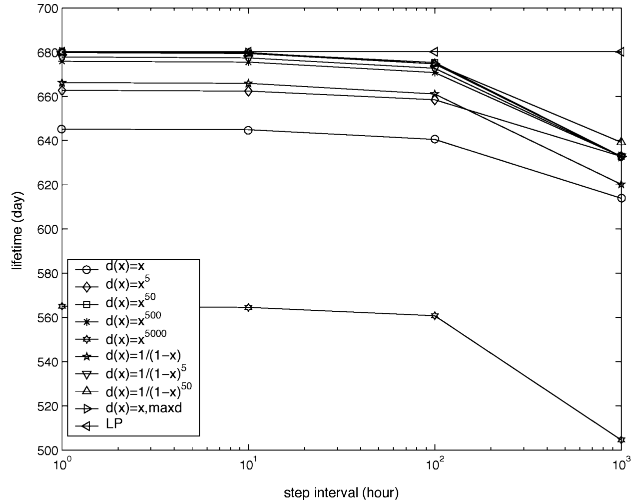

Fig. 5 Average battery lifetime of the random configurations of 20 nodes for different cost functions and step intervals

Figure 6 shows the effect of different cost functions on the number of nodes connected to the AP over time. The figure justifies estimating the network lifetime by the time duration until the first node dies since the time duration from the disconnectedness of 5% of the nodes to that of 50% of the nodes is very small. The least max-cost path algorithm and least sum-cost path algorithm with cost function $d(x) = 1/(1-x)^n$ for large n gives the best lifetime estimate in terms of the disconnectedness of a small percentage of the nodes. However, they perform worse than the least sum-cost path algorithms with cost function $d(x) = x^n$ for small n in terms of achieving a large lifetime until a higher percentage of the nodes become disconnected. The reason is that the latter case considers all the nodes on the path whereas the first case considers only the critical nodes with less remaining energy. In the long run, considering only the critical nodes creates cycles in the resulting flow rates causing more energy to be spent in non-critical nodes, which causes the energy of the remaining nodes to deplete earlier than other algorithms.

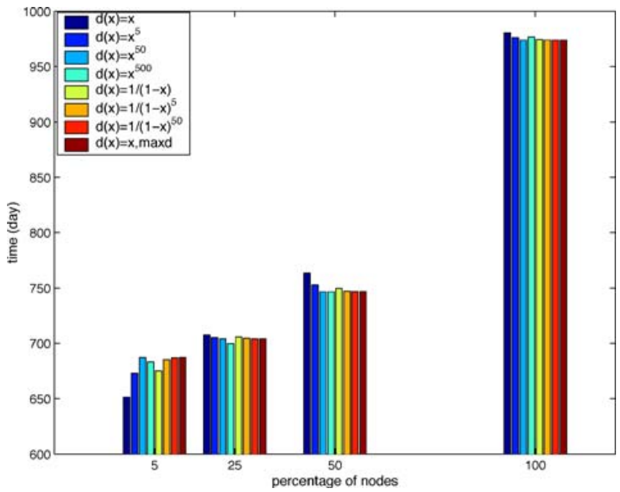

Fig. 6 The time at which a specific percentage of the nodes are either dead or disconnected from the AP

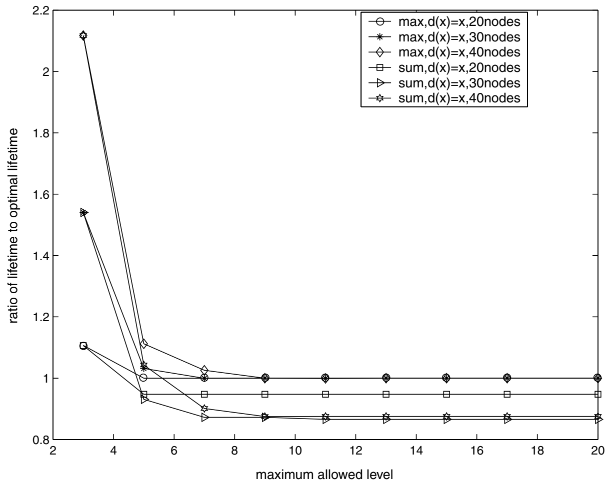

Fig. 7 Average battery lifetime of the random configurations of different number of nodes as a function of maximum allowed path length

The simulation for energy efficient routing with delay guarantee is performed to understand the tradeoff between providing delay guarantee and achieving a high lifetime. In all plots below, 'maximum allowed level' and 'maximum delay (slot)' denote d_{max} and $delay_{max}$ respectively, which are defined in Section 5.

Figure 7 shows that the increase in the maximum allowed length of the routing paths beyond the value required for connectivity does not affect the network lifetime for LR-ENR for both least-sum and least-max cost path algorithms for random configurations of different number of nodes. The first decrease in lifetime results from the increase in connectivity

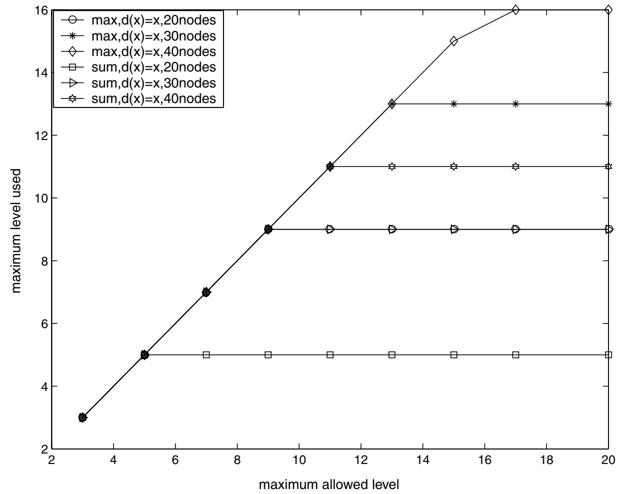

Fig. 8 Maximum level used in the routing of the random configurations of different number of nodes as a function of maximum allowed path length

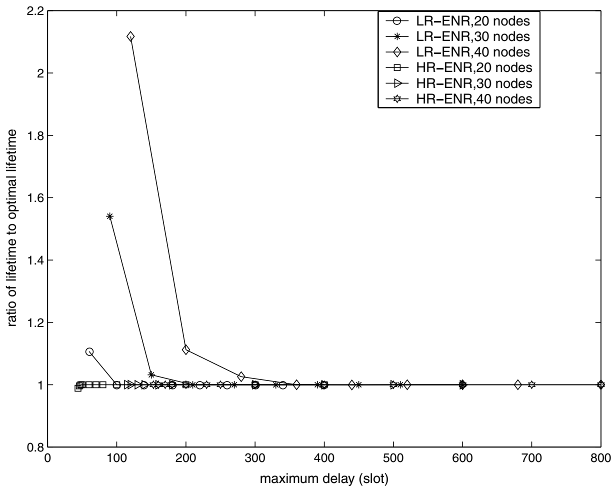

Fig. 9 Average battery lifetime of the random configurations of different number of nodes as a function of maximum allowed delay

of the network. The maximum level used in the network however keeps increasing as the maximum allowed length increases for the same lifetime as shown in Figure 8. This suggests that increasing the delay of the network does not necessarily increase the network lifetime.

Although the maximum level used in routing is higher in least-max cost path algorithms than least-sum cost path algorithms, least-max cost path algorithms achieve a higher lifetime in terms of the death of the first node. Therefore, the following simulations that compare the performance of LR-ENR and HR-ENR in terms of achieving optimal lifetime is performed for least-max cost path algorithms.

Figure 9 shows the lifetime of the network for LR-ENR and HR-ENR and random configurations of different number of nodes. For the maximum allowed delay where LR-ENR cannot provide connectivity, HR-ENR provides connectivity of all the nodes and achieves optimal lifetime. At the delay value where the connectivity of all the nodes is achieved, the lifetime is equal to optimal lifetime. This means that the uniform distribution of packets in a uniform topology causes the connectivity and optimal lifetime of the network to be achieved simultaneously.

Figure 10 shows the lifetime of the network for the grid configuration of 49 nodes for LR-ENR and HR-ENR. ‘same’ refers to the case where all the nodes generate packets at the same rate whereas ‘diff’ refers to the case where the packet generation rate at one side of the grid is twice that of the other side. For LR-ENR, once the connectivity of the nodes are achieved, the optimal lifetime is also achieved as in the uniform configuration of the nodes. For HR-ENR, although the lifetime is close to the optimal lifetime once all the nodes are connected for the ‘same’, the lifetime increases from 80% to 100% as the maximum allowed delay increases for the ‘diff’ case. This suggests that the non-uniform distribution of the packets in a uniform distribution of the nodes results in achieving connectivity before achieving the optimal lifetime.

Figure 11 shows the lifetime of the network for the ring configuration of 50 nodes for LR-ENR and HR-ENR. ‘same’ refers to the case where all the nodes generate packets at the

Fig. 10 Average battery lifetime of the grid configuration of 49 nodes as a function of maximum allowed delay

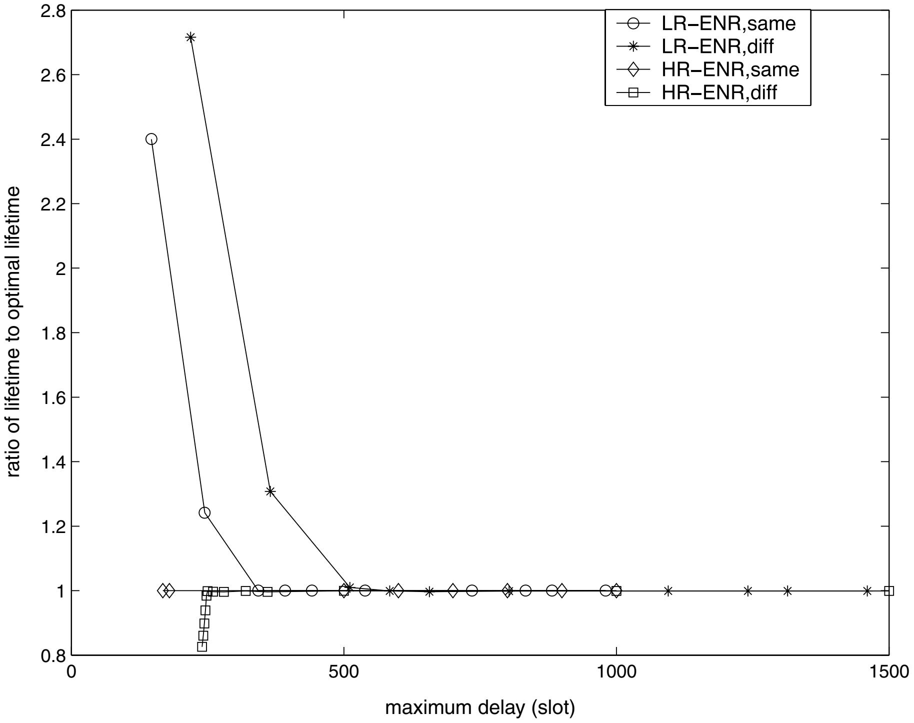

Table 3 Minimum and maximum path lengths and delays used for LR-ENR and HR-ENR respectively to achieve optimal lifetime and network connectivity

Configuration	LR-ENR min	LR-ENR max	HR-ENR min	HR-ENR max
Random, 20 nodes	5	9	48	68
Random, 30 nodes	9	13	115	152
Random, 40 nodes	9	16	153	264
Ring, 50 nodes - same	26	49	625	1177
Ring, 50 nodes - diff	32	49	1100	2054
Grid, 50 nodes - same	7	7	168	168
Grid, 50 nodes - diff	7	17	250	479

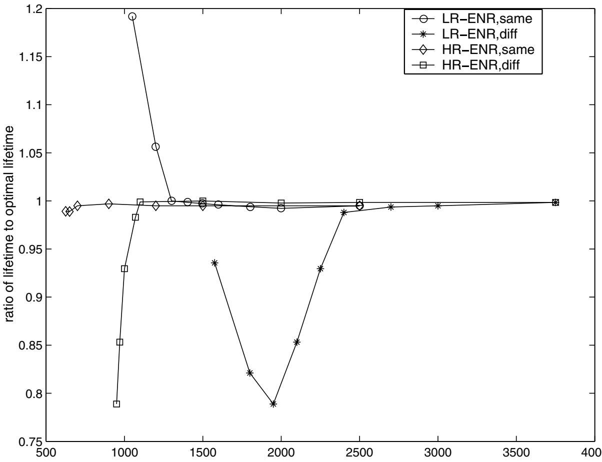

Fig. 11 Average battery lifetime of the ring configuration of 50 nodes as a function of maximum allowed delay

same rate whereas 'diff' refers to the case where the nodes on one side of the ring generate twice as many packets as those on the other side of the ring. In the 'diff' case, the lifetime increases from 77% to 100% after the connectivity of the nodes is achieved for both LR-ENR and HR-ENR. There is even a slight increase in the lifetime of 'HR-ENR, same' case after the connectivity is achieved. The main feature of ring network is that there are only two path options for each node, which may be of very different lengths for some nodes. The non-uniform distribution of the nodes therefore is another factor for the network lifetime to be below the optimal lifetime despite network connectivity.

When the maximum allowed delay increases, the network delay keeps increasing although the connectivity of all the nodes and optimal lifetime is achieved. To understand the decrease in network performance by not putting any constraint on the network delay, Table 3 compares the minimum and maximum value of the path length and delay values that achieve optimal lifetime and connectivity for LR-ENR and HR-ENR respectively. 'LR-ENR min' and 'LR-ENR max' correspond to minimum and maximum path length, whereas 'HR-ENR min' and 'HR-ENR max' correspond to minimum and maximum delay. For random configurations, the difference between them increases as the number of nodes increases for both LR-ENR and HR-ENR. For 40-50 nodes, for random, ring and grid configurations, the ratio of maximum delay to minimum delay increases up to 2. This ratio is expected to increase as the number of nodes increases.

8. Conclusion

In this paper, we present a routing protocol for sensor networks with the goal of maximizing the time duration until the first node dies. We show that the lifetime can be estimated by the minimum lifetime of the nodes over the network since the time from the disconnectedness of 5% to 50% of the nodes from AP is very small and the quality of data at the AP decreases as a result of the death of the nodes.

We first formulate energy efficient routing as a Linear Programming (LP) problem in which the objective is to maximize the lifetime of the network. We then give a centralized routing algorithm based on this LP solution. This optimal routing is shown to increase the network lifetime by 50-150 days over the minimum hop routing. We also describe a distributed algorithm based on iterative least cost path routing where the cost of each path is either the sum or the maximum of the cost of the nodes on that path. We show that the least cost path routing for the later achieves optimal lifetime when the cost of each node is given by the ratio of its total consumed energy to its initial energy.

We then modify the energy efficient routing to provide a delay guarantee by limiting the length of the routing paths from each node to the AP. The decrease in battery lifetime as a result of decreasing the maximum allowed delay is shown to be considerable for non-uniform distribution of the nodes and uneven packet generation patterns across the network. We also show that when the maximum allowed delay increases, the network delay keeps increasing even after the optimal lifetime is achieved. This favors limiting the maximum delay to the value at which the optimal lifetime is achieved.

References

1. J. Polastre, R. Szewczyk, C. Sharp and D. Culler, *The Mote Revolution: Low Power Wireless Sensor Network Devices*, Hot Chips 16: A Symposium on High Performance Chips (Aug. 2004).
2. S.C. Ergen and P. Varaiya *PEDAMACS: Power Efficient and Delay Aware Medium Access Protocol for Sensor Networks*, to appear in IEEE Transactions on Mobile Computing (2005).
3. S. Coleri, *PEDAMACS: Power Efficient and Delay Aware Medium Access Protocol for Sensor Networks*, Master Thesis, University of California, Berkeley (Dec. 2002).
4. E.M. Royer and C. Toh, *A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks*, IEEE Personal Communications (April 1999) pp. 46–55.

5. S. Singh, M. Woo and C. Raghavendra, *Power-aware Routing in Mobile Ad Hoc Networks*, MOBICOM (1998) pp. 181–190.
6. J.H. Chang and L. Tassiulas, *Maximum Lifetime Routing in Wireless Sensor Networks*, IEEE/ACM Transactions on Networking, Vol.12, issue. 4 (Aug. 2004).
7. M. Bhardwaj and A.P. Chandrakasan, *Bounding the Lifetime of Sensor Networks Via Optimal Role Assignments*, IEEE INFOCOM (2002) pp. 1587–1596.
8. F.J. Block and C.W. Baum, *Energy-Efficient Self-Organizing Communication Protocols for Wireless Sensor Networks*, IEEE MILCOM (2001).
9. F.J. Block and C.W. Baum, *An Energy-Efficient Routing Protocol for Wireless Sensor Networks with Battery Level Uncertainty*, IEEE MILCOM (2002).
10. C.K. Toh, *Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad hoc Networks*, IEEE Communications Magazine (June 2001) pp. 138–147.
11. E. Uysal-Biyikoglu, B. Prabhakar and A. El Gamal, *Energy-Efficient Packet Transmission over a Wireless Link*, IEEE/ACM Transactions on Networking, Vol. 10, No. 4 (Aug. 2002).
12. Swetha Narayanaswamy, Vikas Kawadia, R.S. Sreenivas and P.R. Kumar, *Power Control in Ad-Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW protocol*, Proceedings of European Wireless 2002 (Feb. 2002) Italy.
13. R. Cristescu, B. Beferull-Lozano and M. Vetterli, *On Network Correlated Data Gathering*, IEEE INFOCOM (2004).
14. R.K. Ahuja, T.L. Magnanti and J.B. Orlin, *Network Flows*, Prentice Hall (1993).
15. B. Krishnamachari, S.B. Wicker and B. Bejar, *Phase Transition Phenomena in Wireless Ad-Hoc Networks*, Symposium on Ad-Hoc Wireless Networks, GlobeCom 2001, San Antonio, Texas (Nov. 2001).

Sinem Coleri Ergen received the BS degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2000, and the M.S. and Ph.D. degrees in electrical engineering and computer sciences from University of California Berkeley (UCB), in 2002 and 2005. Since January 2006, she has been a postdoctoral researcher in electrical engineering at UCB. Her research interests are in wireless communications and networking with a current focus on energy efficient system design for sensor networks. She is a member of the Sensor Networks for Traffic Monitoring project at UCB. She received Regents Fellowship from University of California Berkeley in 2000.

Pravin Varaiya is Nortel Networks Distinguished Professor in the Department of Electrical Engineering and Computer Sciences at the University of California, Berkeley. From 1975 to 1992 he was also Professor of Economics at Berkeley. From 1994 to 1997 he was Director of the California PATH program, a multi-university research program dedicated to the solution of Californias transportation problems. His current research is concerned with communication networks, transportation, and hybrid systems. He has taught at MIT and the Federal University of Rio de Janeiro. Varaiya has held a Guggenheim Fellowship and a Miller Research Professorship. He received an Honorary Doctorate from LInstitut National Polytechnique de Toulouse, and the Field Medal of the IEEE Control Systems Society. He is a Fellow of IEEE and a member of the National Academy of Engineering. He is on the editorial board of several journals, including "Discrete Event Dynamical Systems" and "Transportation Research—C". He has co-authored three books and more than 250 technical papers. The second edition of "High-Performance Communication Networks" (with Jean Walrand) was published by Morgan-Kaufmann in 2000. "Structure and interpretation of signals and systems" (with Edward Lee) was published by Addison-Wesley in 2003. Varaiya is a member of the Board of Directors of Sensys Networks.