Connectivity-aware Routing in Sensor Networks

Praveen Kumar, Joy Kuri, and Pavan Nuggehalli
Centre for Electronics Design and Technology
Indian Institute of Science, Bangalore, India

Mario Strasser, Martin May, and Bernhard Plattner
Computer Engineering and Networks Laboratory
ETH Zurich, Switzerland

***Abstract*—Sensor network applications such as environmental monitoring demand that the data collection process be carried out for the longest possible time. Our paper addresses this problem by presenting a routing scheme that ensures that the monitoring network remains connected and hence the live sensor nodes deliver data for a longer duration. We analyze the role of relay nodes (neighbours of the base-station) in maintaining network connectivity and present a routing strategy that, for a particular class of networks, approaches the optimal as the set of relay nodes becomes larger. We then use these findings to develop an appropriate distributed routing protocol using potential-based routing. The basic idea of potential-based routing is to define a (scalar) potential value at each node in the network and forward data to the neighbor with the highest potential. We propose a potential function and evaluate its performance through simulations. The results show that our approach performs better than the well known lifetime maximization policy proposed by Chang and Tassiulas [1], as well as *AODV* [2].**

I. Introduction

Sensor nodes are typically battery-powered devices with limited computing and memory resources that can only communicate with their direct neighbourhood. A wireless sensor *network* is formed by a collection of sensor nodes that collaborate to carry out a particular sensing task. In this paper, we examine the problem of routing in sensor networks.

This problem has recently attracted a lot of attention (see Section VI). Different sensor network applications impose different requirements on routing. Consequently, the routes chosen in these scenarios may considerably vary from each other. Many routing algorithms have been suggested with particular objectives in mind. However, these objectives are relevant in *particular* cases only, and not in others. As an example: the algorithm in [1] seeks to maximize the time until the *first* node runs out of energy—this is referred to as "network lifetime maximization." Implicit in this phrase is the idea that the network serves a useful purpose only as long as *all* the nodes in the network are alive. This may be true in certain application scenarios, but there are others in which this definition of network lifetime is too stringent. In applications like environmental monitoring, for instance, the network continues to operate even after the first node failure. Hence routing schemes that maximize the time until the first node failure may not be useful. In fact, it may be necessary that all live sensor nodes are able to communicate with the sink. In this case, the lifetime of the sensor network is the time for which sensor nodes remain connected to the base-station. Therefore, routing schemes should maximize this lifetime, i.e, they should ensure that the network stays connected.

A network is disconnected or partitioned when one or more sensor nodes are unable to the reach the base-station even when they have enough energy. This happens when a group of sensor nodes runs out of energy. In a network, the burden of data transmission is more on those neighbouring nodes of the base-station that have to send their own data to the base-station as well as forward data from other sensor nodes (we refer to these nodes as relay nodes hereafter). These nodes run out of energy faster than others and are likely to disconnect several sensor nodes from the base-station. Hence, we consider the problem of routing data so that the time when a network gets partitioned because of relay node failure is maximized.

We first analyze simple networks and use the thereby obtained findings to develop a distributed routing scheme using potential-based routing paradigm. The basic idea of potential based routing is to define a *potential* at each node. Once these potentials are defined, the actual forwarding of traffic is achieved by simply choosing the neighbor with the highest potential. One of the advantages of this approach is its flexibility and adaptability. Thus, there is no unique defining function that yields the potential at a node in every case but it depends on the performance metric of interest in different scenarios. Among the several factors that may be considered while assigning potentials are: residual energy at the node, traffic load on the node, number of hops from the node to the base-station along the shortest path, etc.

In this paper, we propose a potential function definition based on our analytical study of simple networks and evaluate its performance by means of simulations. As a baseline for comparison, we extend the well-known lifetime maximization policy in [1] to our context. We also compare our protocol with *AODV* routing protocol [2]. Simulation results show that our approach can lead to considerable advantages.

II. Definitions and Assumptions

We consider a set of stationary sensor nodes that are randomly strewn in a given area. Each sensor samples the environment periodically and produces data that needs to be communicated to a single central base-station at a constant rate. All sensor nodes transmit at a fixed constant power and two nodes can communicate with each other if and only if they are separated by a distance less than a parameter D, the radio range of the sensor nodes. A point p in the given area is said to be *covered* if there exists at least one sensor node at a distance $< C$ from the point p. Equivalently, each sensor covers a circle of radius C centered around itself. Note that sensors can have overlapping coverage areas. We define coverage to be the area covered by nodes in the system. In our analysis, we only focus on the energy spent in transmitting and receiving packets. We ignore the energy cost of sensing and computation.

A. Network Model

We model the sensor network by a graph $G = (V, E)$, where V denotes the set of sensor nodes and E denotes the set communication links between sensor nodes. All communication links are considered to be bidirectional and a link $\{u, v\} \in E$ exists between two sensor nodes u and v if and only if they are within the transmission range of each other. We provide some definitions related to graphs and formally state the problem.

B. Definitions

1) A network graph $G = (V, E)$ is said to be connected if all vertices have a path to the base-station.
2) Let $S \subseteq V$ be a set of vertices. We denote by $G \setminus S$ the graph which is obtained by removing all vertices (and their incident edges) in S from G. More precisely, $G \setminus S := (V \setminus S, \{\{x, y\} : \{x, y\} \in E \wedge x, y \notin S\})$.
3) Given a graph $G = (V, E)$, the separating set $S \subseteq V$ is a set of vertices such that the graph $G \setminus S$ is disconnected.
4) The connectivity of a graph is the size of the minimum sized separating set, that is $\min\{|S| : S \subseteq V \wedge S \text{ is a separating set}\}$. This number represents the minimum number of vertices that have to be removed from G to make the network disconnected.

C. Problem Statement

Consider a network graph G with connectivity n. Let us denote by S_{min} a minimum sized separating set; by definition of connectivity, $|S_{min}| = n$. The network gets disconnected when sensor nodes in S_{min} run out of energy. Therefore, to ensure that the network stays connected for a longer duration, it is necessary to prolong the time when all nodes in S_{min} run out of energy. Thus, the problem is to find a routing scheme so that sensor nodes in S_{min} stay alive for a longer duration. In a graph, there can be several separating sets and several minimum sized separating sets. In what follows, we restrict ourselves to the separating set S^* formed by relays (i.e nodes which are neighbours of the base-station and can forward any non-neighbour's data). If N is the set of neighbours of the base-station u_0 (i.e., $\forall v \in N : \{u_0, v\} \in E$), then S^* is given by $S^* := \{v : v \in N \wedge \exists u \in V \setminus N, u \neq u_0 : \{u, v\} \in E\}$. Clearly, the set of relays forms a separating set, but it may or may not be minimum sized. Our aim is now to maximize the time when the network gets disconnected as relay nodes in this separating set runs out of energy.

III. Analysis

We consider two ways by which a network can get partitioned because of relay node failure alone. In the first case, the network gets partitioned only when all relays run out of energy and in the second case, the network gets partitioned when a subset of relays nodes run out of energy. We first analyze case 1 by considering a simple network and then extend the idea to case 2.

A. Analysis for Case 1

We consider the simple network shown in Figure 1(a). In this network, D is the base-station and other nodes are sensor nodes. All sensor nodes except A generate data at a constant rate of r. A generates data at rate $\gamma r, \gamma > 0$. The connectivity of this network is n as the n neighbors (relays) of the base-station build the only separating set of this network. Here, node A gets disconnected from the base-station only when all n relays die. Hence, atleast one relay should stay alive to ensure that node A has a route to the base-station.

Relays $1, 2, \ldots, n$ send their data directly to the base-station whereas node A sends at rates $r_1, r_2, \ldots, r_n$ via relays $1, 2, \ldots, n$ such that $\sum_{i=1}^{n} r_i = \gamma r$. The problem is now to find $r_1, r_2, \ldots, r_n$ so that the time when all n relays run out of energy is maximized. For that purpose, we consider the simple routing scheme where node A sends all of its data via one of the relays until it runs out of energy and then repeats the same process by switching to another active relay. Furthermore, we denote by

E_0	the initial battery energy of sensor nodes in joules,
E_t	the Transmit energy per bit in joules,
E_r	the receive energy per bit in joules,
T_i	the time at which relay i runs out of energy, and
T_L	the time at which all n relays run out of energy.

The following simple analysis gives T_L for the above scheme.

Let us assume that A picks relay 1 to forward all its data. The total rate of flow out of relay 1 is then $(\gamma + 1)r$. Hence relay 1 runs out of energy first at time

$$T_1 = \frac{E_0}{\gamma r E_r + (\gamma + 1) r E_t}.$$

The remaining energy in relays 2 to n when relay 1 dies is

$$E_0 - T_1 \times r \times E_t = \frac{\gamma E_0 (E_r + E_t)}{\gamma E_r + (\gamma + 1) E_t}.$$

Next, node A picks relay 2 in order to forward its data. The time at which node 2 runs out of energy is,

$$\begin{aligned} T_2 &= T_1 + \frac{\gamma E_0 (E_r + E_t)}{\gamma E_r + (\gamma + 1) E_t} \times \frac{1}{r(\gamma E_r + (\gamma + 1) E_t)} \\ &= \frac{E_0}{\gamma r E_r + (\gamma + 1) r E_t} + \frac{\gamma E_0 (E_r + E_t)}{r(\gamma E_r + (\gamma + 1) E_t)^2} \end{aligned}$$

and the remaining energy in relays 3 to n is

$$E_0 \left(\frac{\gamma (E_r + E_t)}{\gamma E_r + (\gamma + 1) E_t} \right)^2.$$

In general relay i dies at

$$T_i = \sum_{j=1}^{i} \frac{E_0 (\gamma (E_r + E_t))^{j-1}}{r (\gamma E_r + (\gamma + 1) E_t)^j}.$$

Therefore, the time at which all n relays run out of energy is

$$T_L = \sum_{j=1}^{n} \frac{E_0 (\gamma (E_r + E_t))^{j-1}}{r (\gamma E_r + (\gamma + 1) E_t)^j} = \left(1 - \left(\frac{\gamma E_r + \gamma E_t}{\gamma E_r + (\gamma + 1) E_t} \right)^n \right) \frac{E_0}{r E_t}.$$

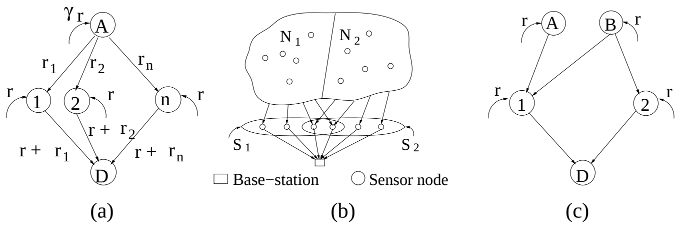

Fig. 1. (a) A simple n relay network (b) A case-2 network (c) An example network for case 2

Let T_L^* be the optimal time at which all n relays die. Now, by definition $T_L^* \geq T_L$. Moreover $T_L^* < \frac{E}{rE_t}$, since any relay will run out of energy by time $\frac{E_0}{rE_t}$. Thus,

$$\left(1 - \left(\frac{\gamma E_r + \gamma E_t}{\gamma E_r + (\gamma+1)E_t}\right)^n\right) \times \frac{E_0}{rE_t} \leq T_L^* < \frac{E_0}{rE_t}.$$

Since $\frac{\gamma E_r + \gamma E_t}{\gamma E_r + (\gamma+1)E_t} < 1$, the bounds get closer as n increases and therefore T_L approaches T_L^*. Thus, the time when the network gets disconnected with the above routing scheme is closer to the optimal value for higher number of relays.

In the above analysis, we considered a simple network and showed that by choosing relays one by one, the time to network partition can be maximized. Next, we consider a network where node A is replaced by a set of sensor nodes of which all are able to reach the base-station through any relay. In addition, let the size of this set be γ so that it can be represented by a single node generating data at a rate γr. The time when this node gets disconnected will be maximized if it chooses relays one by one, as shown in previous analysis. Thus, all sensor nodes in this set should send their data through a single relay until it runs out of energy and then switch together to another relay.

B. Extension to Case 2

Next, we analyze cases where the network gets disconnected when a subset of relays die. Therefore, consider the network shown in Figure 1(b). Let R denote the set of relays and $N := V \backslash R$ be the set of nodes that are not neighbours of the base-station.

We denote by $N_1 \subseteq N$ the set of sensor nodes that get disconnected when S_1 runs out of energy and by $N_2 := N \backslash N_1$ the set of nodes that get disconnected when a separating set say S_2 runs out of energy (see Figure 1(b)). Let $S_{12} := S_1 \cap S_2$ be the set of relays that has neighbors in N_1 as well as in N_2. The network gets disconnected when either of these separating sets runs out of energy.

In order to maximize the time when N_1 gets disconnected, nodes in N_1 should use the relays in S_1 one by one and the load on S_1 from nodes in N_2 should be minimal. Due to symmetry, a similar method has to be followed to maximize the time when N_2 gets disconnected. If a node in N_1 starts by using relays in S_{12} and then moves on to other relays in $S_1 \backslash S_{12}$, it increases the load on S_2, since $S_{12} \subset S_2$. As a result, N_2 can get disconnected. The same applies to a situation where N_2 starts by using relays in S_{12}. Hence, both N_1 and N_2 should avoid using relays in S_{12}, i.e, they should first use relays in $S_1 \backslash S_{12}$ and $S_2 \backslash S_{12}$, respectively.

As an example, consider a simple network shown in Figure 1(c). Here, $S_1 = \{1\}$, $N_1 = \{A\}$, $S_2 = \{1, 2\}$, $N_2 = \{B\}$ and $S_{12} = \{1\}$. In this network, if both A and B send their data via relay 1, it dies at,

$$T_1 = \frac{E_0}{(2rE_r + 3rE_t)}.$$

Since, node A gets disconnected when relay 1 dies, the time at which the network gets partitioned in this case is same as T_1. Alternately, if node A uses relay 1 and node B uses relay 2, relays 1 and 2 will die at the same time given by,

$$T_{12} = \frac{E_0}{(rE_r + 2rE_t)}.$$

Here, the network gets disconnected at T_{12}. Note that $T_{12} > T_1$. Thus, when node B avoids using a relay in S_{12}, disconnection occurs at a later time.

In general, there can be several separating subsets of the relay set and in all those cases relays that lie in the intersection of several separating sets should be used last. It can be observed that the relays in S_{12} connect more sensor nodes to the base-station than the relays in $S_1 \backslash S_{12}$ or $S_2 \backslash S_{12}$. Sensor nodes should thus avoid using such relays while routing their data. In the following section, we use this idea to obtain a distributed routing algorithm using potential based routing.

IV. Potential-based Routing

In potential-based routing [3], each node is assigned a potential value and all nodes keep track of the potential values of their neighboring nodes. The base-station or sink node has its potential set to infinity. While forwarding, a node sends packets to the neighbour with the highest potential. Consequently, a node will not forward any packet if its potential value exceeds the potential values of all its neighbors. Hence, for potential-based routing to work, there should be no local maxima at a node other than the base-station. In the remainder of this section, we introduce our potential function and show that this potential function ensures that the above condition is met.

A. Potential Function: Definition

In potential-based routing schemes the potential function must be designed to get the desired performance. In our case the potential function must be set so that the network stays connected as long as possible. We design a generic potential function and apply it to our case. This potential function (hereafter referred to as *PBR*) is described as follows:

Consider a network with N nodes and let $w_1, w_2, \ldots, w_N$ be some positive weights assigned to these nodes. Now, consider a node (say A) that has L different routes to the base-station. Let $d_1, d_2, \ldots, d_L$ be the hop-lengths of these L distinct routes and define W_{min}^i, $1 \leq i \leq L$, as the minimum weight of all nodes on route i. The potential at node A, is then defined as

$$\phi_A = \max_{1 \leq i \leq L} \frac{W_{min}^i}{d_i^k}$$

where k is some positive parameter.

The potential value at a node represents the quality of the routes from the node to the base-station; the higher the potential the better the quality. In case of our potential function, the potential value at a node is higher if the node has a route to the base-station with a higher minimum weight. However, if this route is of longer hop length, the potential drops. The decrease-rate of the potential is controlled by k. For a smaller value of k, the potential value decreases slowly with the hop length. Hence a route with a higher minimum weight gives a higher potential even if its hop length is high. For higher values of k, however, the potential value decrease quickly with an increasing distance. In this case routes with shorter hop length result in a higher potential at a node even if their minimum weight is lower.

We use this potential function to prolong the connectivity of the network. To this end, we assign positive weights to neighbours of the base-station and infinite weights to all other nodes. In this case, W^i_{min} for a route i is equal to the minimum of all weights of the relay nodes that lie on route i. Thus, for small ks, if a node A can reach the base-station via many relays, the route that passes through the relay with the highest weight determines the potential at A. As a result, node A sends its data via a single relay, namely the relay with the highest weight in the set of relays R that connect A to the base-station.

The assignment of weights to relays is based on our analysis. We saw in Section III that if a relay node connects many sensor nodes to the base-station it must be used later. In order to meet this condition, the weight of a relay node p is calculated as follows:

We find the number of sensor nodes that stay connected to the base-station when all neighbours of the base-station other than p are removed from the network. These nodes must reach the base-station only through p. Now, the weight of p is made inversely proportional to this number. When this number is high (i.e., when p connects many sensor nodes to the base-station), the weight of p decreases, indicating that it should be used later. Thus, if a node A can reach the base-station via a set R of relay nodes, the relay with the least weight in R would be used last. This is similar to the routing scheme discussed in section III.

As already mentioned, a network gets partitioned when a group of nodes runs out of energy. When this group consists only of relays, *PBR* performs well since it routes data through appropriate relays. However, in general a network can get partitioned even when any other group runs out of energy. Hence, we evaluate *PBR* for all cases in section V. Since the network topology changes when nodes run out of energy, we propose that potentials be recalculated whenever a node dies. It remains to show that our potential function generates no local maxima at a node other than the base-station.

Theorem 1: Given an arbitrary assignment of weights to nodes, the potential based routing scheme PBR does not allow local maxima to develop.

Proof: Consider an arbitrary node A. Let its potential be ϕ_A. Let us denote by l the route from node A that maximizes the ratio $\frac{W^i_{min}}{d^k}$ for all routes originating from node A. Let node B be the neighbour of node A that lies is route l and j is the route from node B that maximizes the ratio $\frac{W^i_{min}}{d^k}$ for all routes originating from node B. By assumption, route l contains B. Therefore, route l also provides us a route from B to the base-station. Let us denote this route by i. Let W^i_{min}, W^j_{min} and W^l_{min} be the weights of routes i, j, and l. Then $\phi_A = \frac{W^l_{min}}{d^k_l}$ and $\phi_B = \frac{W^j_{min}}{d^k_j}$. By definition, $\phi_B \geq \frac{W^i_{min}}{d^k_i}$. Since $d_l = d_i + 1$ and $W^l_{min} \leq W^i_{min}$, we have

$$\phi_A = \frac{W^l_{min}}{d^k_l} < \frac{W^i_{min}}{d^k_i} \leq \frac{W^j_{min}}{d^k_j} = \phi_B$$

■

Though non-relay nodes have infinite weight with our weight assignment scheme, the weight of a route i (W^i_{min}) is finite since every route contains a relay (with finite weight). Thus, the theorem also holds in our case.

B. Practical Implementation

In this section, we describe the implementation of our potential-based routing scheme. Sensor nodes compute potentials through flooding induced by the base-station. That is, the base-station broadcasts an update message that propagates throughout the network. The update message contains the following fields:

1) **Weight field (W):** This field carries the weight (W^i_{min}) of a route. The base-station sets this value to infinity. Nodes update this field by setting it to the minimum of their weight and the value present in the field.
2) **Hop Count field (d):** The base-station sets this value to zero and it is incremented by every node.

Upon reception of this message, a node computes the path metric given by $\frac{\min(W, w_u)}{(d+1)^k}$. If this path metric is greater than the current potential of the node, the current value is updated and the message is broadcasted again. Otherwise the message is dropped. Nodes exchange their potential values through local broadcasts. Link and node failures are detected by absence of (several) ACK packets. In either case the neighbor's potential is removed from the table. A node that detects a node failure sends an update-request message to the base-station which then broadcasts a new potential update message. In section IV-A, weights were computed using knowledge about the entire network. Since relay nodes do not have complete information, an approximate weight computation is done as follows: Each relay node first finds out the number of one hop neighbours it has using the local topology knowledge. It then finds out the number of two hop neighbours by getting information from its one hop neighbours. The total number of one hop and two hop neighbours of a relay node gives an indication of the number of sensor nodes that can reach the base-station through it. The assigned weights are made inversely proportional to this number. All other sensor nodes set their weights to infinity.

V. Evaluation

We evaluate the performance of the presented potential-based routing scheme through simulations. The metrics that we use to measure the performance are:

1) **Node lifetimes:** A sensor node is said to be dead either when it runs out of energy or does not have a route to the base-station; the lifetime of a node is the duration for which a node lives.
2) **Time to partition:** This is the time at which the network gets partitioned.
3) **Coverage process:** This process shows how the fraction of area covered by active sensor nodes decreases as sensor nodes die.

The rest of this section is organized as follows: We compare, in section V-A, the performance of *PBR* with a lifetime maximization policy proposed in [1]. In section V-B, we evaluate *PBR* using Glomosim and compare it with *AODV*.

A. Model Evaluation

In this section, we evaluate the performance of the presented potential-based routing scheme and the lifetime maximization policy proposed in [1] using Matlab and compare their performance. The lifetime maximization policy in [1] uses a linear programming based approach for routing. According to [1], the routes that maximize the time to first node failure can be obtained by solving a linear program. However, in our scenario, the network continues to function even after the first node runs out of energy. Therefore, we adapt this approach to our context by repeatedly solving the linear program in [1] to get new routes for every new topology that arises as time progresses. Hereafter we refer to this approach as *LP* based approach. In case of *PBR*, weights and potentials are computed in a centralized manner using information about the entire network topology and are then assigned to nodes.

Simulation Scenario: We consider a 30 node static sensor network of which one node is the base-station and the remaining nodes are sensor nodes. The deployment region is a square of size 200×200 meter and the communication range is 80 meter. We assume that nodes transmit at a constant power level. All nodes start with the same amount of energy and energy is consumed only when nodes transmit and receive data. Also, we assume that the transmission of a packet consumes three times more energy than its reception. All sensor nodes generate data at a constant rate of 1 packet per minute; the data packet length is 1000 bits.

Simulation Results: Simulations were run for several randomly generated networks. We study the performance of *PBR* in networks where disconnection occurred due to relay node failure only (we call them special networks). i.e, When the network gets disconnected, a non-relay node should not have run out of energy. We also consider general networks where disconnection occurred because of non-relay node as well as relay node failures. We only obtain the average time to partition for comparison with LP.

In section III-B, we showed that our scheme maximizes the time to partition for special networks. Simulation results given in Table I confirm it. Moreover, the scheme also performs well in general networks. The time to partition is about 8% and 6% higher in *PBR* for special and general networks respectively.

Network	"Special"- Partition due to relay node failure only	"General" - Partition due to relay node failure or any other node failure
LP	265	260
PBR	287	276

TABLE I
TIME TO PARTITION (IN HOURS) FOR *PBR* AND *LP*

B. Protocol Evaluation

Next, we evaluate the performance of *PBR* with Glomosim and compare it with *AODV*. We implement *PBR* using the flooding technique described in section IV-B.

Simulation Scenario: We consider a 30 node static sensor network. We assume a MAC with negligible idle energy consumption so that energy is only consumed when packets are transmitted (including retransmissions) or received. This evaluation gives us more practical results as we no longer assume perfect knowledge at all nodes and the evaluation accounts for lower layer effects as interference, collisions, retransmissions, as well as routing protocol overhead.

Simulation Results: We study separately the performance of *PBR* and *AODV* in special and general networks. The results are shown in Figures 2(a) to 2(d).

The node lifetime curves in Figures 2(a) and 2(c) show the average time at which events E_i, $1 \leq i \leq N$ occur. Here, event E_i denotes the death of the i^{th} node in the network. In special networks, we see from Figure 2(a) that the first few nodes die at an earlier time in *PBR*, while a significant number die later as nodes stay connected for a longer duration. The same behavior can be observed for general networks (see Figure 2(c)). The time to partition is about 7.5% and 4.2% higher in *PBR* for special and general networks respectively.

The coverage process for *PBR* and *AODV* is shown in Figures 2(b) and 2(d). This curve shows the average fraction of covered area after every node death event (E_i). In case of *PBR*, the lifetimes of several nodes are higher than in *AODV*. Hence, the coverage process for *PBR* is better than that for *AODV* for both special as well as general networks.

For some networks, the improvement can be quite considerable. In Figure 3, we show an example 30-node network in which the time-to-partition is about 28% longer with *PBR*. The reason for this can be seen with reference to Figure 1(b). The *PBR* scheme seeks to reduce the load on the nodes in the set $S_1 \cap S_2$, while *AODV* does not. This turns out to be a crucial factor, and leads to substantial improvement.

VI. RELATED WORK

A. Potential-based Routing

The idea of *PBR* was first proposed in [4] for unicast routing in the Internet. Here, *PBR* is used to route packets around congested nodes. In [3], the authors use *PBR* as a unified model to study density and proximity-based anycast routing for mobile networks. In this paper, we explored the idea of *PBR* in a simple monitoring network to ensure that sensor nodes stay connected to the base-station for a long duration.

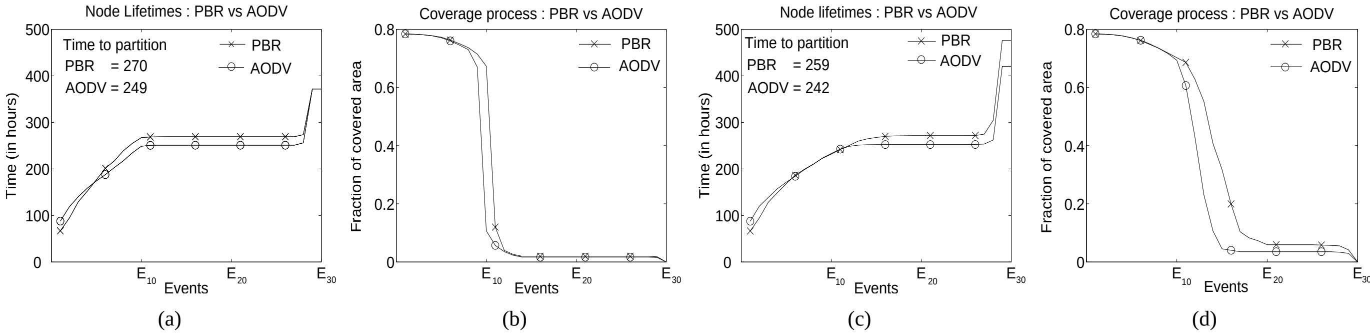

Fig. 2. Performance comparison between *PBR* and *AODV*. (a), (b) Node lifetimes and coverage for special networks. (c), (d) Node lifetimes and coverage for general networks.

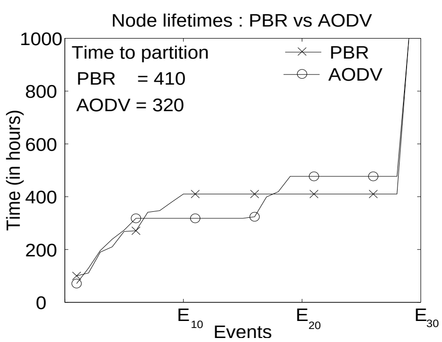

Fig. 3. Node lifetimes for a network that gave considerable improvement

B. Energy-aware Routing

In monitoring applications, a deployed sensor network is expected to last months, if not years. Therefore, efficient energy management is a critical issue in such applications. Noting that low-energy operation is critical to long lifetimes, some of the early work in sensor network routing focused finding routes where the energy consumed is minimized. [5] and [6] are prominent examples of algorithms in this category.

Even though minimizing the consumed energy is a natural strategy, it was realized soon that this does not necessarily lead to longer lifetimes. In much of the published literature, network lifetime is defined to be the interval until the *first* node runs out of energy. If the energy expenditure can be distributed among multiple nodes evenly, network lifetime would improve. This is essentially a load balancing technique. [1], [7] and [8] note that there is an inherent trade-off between minimum energy routes and load balancing. [9] considers a generalization of the network lifetime definition mentioned before and gives an *LP*-based algorithm for computing the maximal node-life curve (number of live nodes against time).

Recently, several authors have approached network lifetime from the perspective of *coverage* and *connectivity*. In [10] and [11], and references therein, the authors consider dense deployment of sensors, where, energy can be saved if many sensors are switched off, while keeping just enough on so as to cover the area completely. In [12], authors try to maintain full coverage by proposing a density based route selection protocol which avoids depleting nodes that result in loss of covered area. Here, authors define a metric called potential energy to identify such nodes. This is different from the potential values used in our *PBR* scheme. In this paper, we consider a small sized monitoring network, where it is critical that sensor nodes stay connected to the base-station and deliver more data.

VII. Conclusions

In this paper, we proposed a routing scheme for data collection in monitoring network and explored the idea of potential-based routing in such applications. We showed using simulations and analysis that our scheme performs better than *LP* and *AODV* routing approaches. In future, we plan to give distributed schemes for weight computation and also address network partition due to failure of non-relay nodes.

References

[1] J.H.Chang and L.Tassiulas. Maximum lifetime routing in wireless sensor networks. *IEEE/ACM Transaction on Networking*, 12(4):609–619, Aug 2004.

[2] Charles Perkins. Adhoc on demand distance vector routing. In *Internet Draft, draft-ietf-manet-aodv-00.txt*, 1997.

[3] V. Lenders, M. May, and B. Plattner. Density-based vs. proximity-based anycast routing for mobile networks. In *IEEE Infocom*, 2006.

[4] A. Basu, A. Lin, and S. Ramanathan. Routing using potentials: A dynamic traffic-aware routing algorithm. In *ACM SIGCOMM*, 2003.

[5] V. Rodoplu and T. H. Meng. Minimum energy mobile wireless networks. *IEEE Journal on Selected Areas in Communication*, 17(8):1333–1344, 1999.

[6] L. Li and Y. Halpern. Minimum energy mobile wireless networks revisited. In *IEEE International Conference on Communications (ICC)*, pages 278–283, 2001.

[7] S. Singh, M. Woo, and C.S. Raghavendra. Power-aware routing in mobile adhoc networks. In *ACM/IEEE International Conference on Mobile Computing and Networking*, 1998.

[8] R. Madan, S. Cui, S. Lall, and A. Goldsmith. Cross-layer design for lifetime maximization in interference-limited wireless sensor networks. *IEEE/ACM Transaction on Networking*, 10(2):208–216, Apr 2004.

[9] T. X. Brown, H. N. Gabow, and Q. Zhang. Maximum flow-life curve for a wireless ad-hoc network. In *ACM Mobihoc*, 2001.

[10] H. Zhang and J. Hou. On deriving the upper bound on α-lifetime for large sensor networks. In *ACM Mobihoc*, 2004.

[11] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. SPAN: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *ACM Mobicom*, 2001.

[12] Ren-Song and Chih-Chung Lai. Density-based routing mechanism for sensor networks. In *ISCC 2005*, 2003.