Joint Sensor Selection and Data Routing in Sensor Networks

Ozgur Ercetin, Ozgur Gurbuz, Kerem Bulbul and Aylin Aksu

Faculty of Engineering, Sabanci University, Istanbul 34956, Turkey.
{oercetin@, ogurbuz@, bulbul@, aylinaksu@su.}sabanciuniv.edu

Abstract. We propose a new joint sensor selection and routing algorithm, which selects a set of sensor nodes (sensing nodes) in a sensor network to take measurements, and determines a set of paths connecting the sensing nodes to the sink node. Our objective is to maximize the network lifetime, while satisfying the data precision required by the user. We first develop a multi-objective optimization model for this problem and design the near-optimal OPT-RE algorithm based on this model for network lifetime maximization. Next, we design a low complexity heuristic called SP-RE. SP-RE first labels the links between the nodes with a metric which trades off the residual energies of the transmitting and receiving nodes with the required transmission and reception energy. Then, SP-RE calculates the shortest paths from all nodes to the sink, and identifies the node which is closest to the sink as a *sensing* node. This process is repeated until the required data precision is satisfied. We demonstrate by simulations that SP-RE and OPT-RE can increase the network lifetime several orders of magnitude compared to naive approaches.

1   Introduction

Sensor networks consist of hundreds or thousands of low cost nodes that come with wireless communication and certain processing and storage capabilities in addition to sensing capabilities. One of the most important applications foreseen for sensor networks is environmental monitoring, where the sensor nodes are embedded into a physical environment to monitor and collect data about the ambient conditions. The nodes are deployed in the sensor field and can be thought of as distributed streaming data sources supporting various (monitoring) applications running at the base station. The data is collected from sensor nodes via wireless multi-hop communication. Sensor nodes usually have limited battery capacities and it is impractical to replace the batteries of thousands of sensor nodes after they are deployed in the field. Thus, a key challenge in ad-hoc, data-gathering wireless sensor networks is prolonging the lifetime of the network.

   In this paper, we aim to design algorithms which jointly select a set of *sensing* nodes in a sensor network, and the paths from each of these nodes to the sink in an energy-efficient way. The nodes in the sensor network cannot be selected randomly: First, the required data precision should be considered; and second, the residual energy of the nodes should be taken into account. As an illustrative

example, consider a simple application in which sensor nodes are used to monitor ambient conditions such as temperature, humidity, light, etc., in a physical area. While monitoring the conditions in this area, one extreme case is to use all the sensor nodes and get almost a complete picture of the conditions. This option leads to maximum possible precision; however, using all nodes in the network leads to very high energy consumption and a short network lifetime. Another extreme solution is to use a single sensor node to monitor the whole area regardless of the area size. This results in a very low resolution picture of the ambient conditions in the area, but it is clear that for this case the energy consumption would be very low. These two solutions are extreme cases and neither of them will be useful in real implementations. In a real implementation, a user or monitoring application will select a certain measurement precision level required by the application. This precision level determines how close the measurements should be taken over the field, e.g., every D square meters. Then, if the sensor network is densely deployed, an important question is to select a node within every such D square meter area for sensing and also to determine an appropriate route from this sensing node to the monitoring station so that the network lifetime is maximized. We investigate the answer to this question in the following sections.

Our work is related to the efforts on topology control and in-network query processing in wireless ad hoc and sensor networks [2], [3], [4], and [5]. Previously in the literature, topology control is investigated with the objective of providing end-to-end traffic, a connected and power/energy efficient path. Our work brings together application-specific requirements (in the form of measurement precision), topology control and routing. The closest model to ours is described in [3]; however, in [3] the objective is to detect the occurrence of an event in an energy-efficient manner rather than to continuously collect the ambient conditions of the area. More recently, Boulis et al. [1] and Kalpakis et al. [6], considered the energy-accuracy trade off in sensor networks when the network is continuously monitored. Boulis et al., investigated the periodic aggregation and estimation problem when the observed data is time correlated. Kalpakis et al., discussed a query-based operation and suggested storing summary data structures at the monitoring station rather than collecting similar data again for each new query. This stored aggregate value is used to answer queries at the monitoring station approximately to improve network lifetime. In our paper, we do not consider in-network processing, but we discuss the optimal selection of sensor nodes and routing of the sensor data given a required measurement precision.

The paper is organized as follows: In section 2, the system model is presented. In section 3, the lifetime maximization problem is formulated as a multi-objective mixed integer program, and a mathematical programming based heuristic for this problem is proposed. In section 4, a low complexity joint node and path selection algorithm is developed. In section 5, results of our performance analysis are presented that include a comparison of our two heuristics along with detailed simulations that benchmark our heuristics against traditional algorithms from the literature. Section 6 involves our conclusions and further research directions.

2   System Model

The sensor network is used to collect information from the sensors distributed in a field that is required for supporting various (monitoring) applications. Each node in the sensor network has the same sensing capability. The nodes can be deployed in this field randomly or according to a predefined topology. If the nodes are deployed randomly, we assume that the nodes can determine their respective locations by using methods such as those proposed in [7]. For ease of instruction, we discuss our model for the case when sensor nodes are deployed in 1-dimension, i.e., nodes are deployed on a line emanating from the monitoring station. It is easy to extend this model to 2-dimensions.

We consider a query-based operation: A user injects queries to the network, which includes a precision requirement in the form of "measurement every D meters". Upon receipt of this query, the network selects the set of sensing nodes among a number of candidate nodes (nodes that are within ϵ distance to the required measurement points). Note that if the total number of sensor nodes in the network and their locations are known, then defining the precision by the measurement interval is equivalent to defining the precision by the number of sensing nodes and how far off they can be located from each required measurement point. For example, if there are 100 nodes randomly distributed over a line of 100 meters, then asking for a measurement precision of every 10 meters is equivalent to selecting 10 sensing nodes that are as equidistant to each other as possible.

Once the sensing nodes are selected, the sensor data has to be forwarded to the monitoring station. In general, the message follows a multi-hop path from a sensing node to the destination. The path is defined by the set of relay nodes and their transmission powers. We assume that the transmission power of each node is uniquely determined by the distance between the transmitting node and the next node on the path receiving the message.

3   Lifetime Maximization Problem

The network lifetime is defined as the duration over which the network can respond to the measurement queries. In general, every query may require a different precision which we do not know in advance. In other words, the generation of queries over time is a random process, and we cannot maximize the network lifetime unless we know the distribution of this random process. Therefore, in order to prolong the network lifetime we propose a slightly different approach motivated by the following two observations: first, when data is routed from the sensing nodes to the monitoring station, it makes sense to avoid using nodes with small residual energy so that no node dies too early. Second, we would like to minimize the total energy consumption from sensing, transmission and receiving when responding to a query. Thus, by not depleting the energy of any individual node too quickly and minimizing the total energy consumption, we intend to increase the network lifetime. Below, we formulate these two goals in

a multi-objective optimization framework and then explain briefly how we solve this optimization model.

Assume that a given query Q requires that K sensing nodes are chosen out of a total of N nodes, and data from these sensing nodes are routed to the monitoring station. The parameter K is the required precision of query Q and is related to the interval between successive measurements as discussed in Section 2. Note that the nodes are deployed randomly in 1-dimension within a measurement interval of I units, and let x_i denote the location of node i, $i = 1, \ldots, N$. Without loss of generality, the monitoring station is denoted by the index $i = 0$ and located at $x_0 = 0$. The nodes are numbered in non-decreasing order of their respective distances to the monitoring station. The main sources of energy consumption are the sensing, transmission and receiving operations. Let E_i, $i = 1, \ldots, N$, be the initial energy level of node i before Q is processed, and let ξ and e_r be the constant amount of energy required for a single sensing and receiving operation by any node, respectively. The transmission energy depends on the distance between two nodes, and e_{ij}, $i = 1, \ldots N$, $j = 0, \ldots, N$, denotes the amount of energy consumed by node i when sending a unit size packet to node j. The messages generated by the K sensing nodes are routed to the monitoring station over multi-hop paths where the binary variable f_{ij}^k, $k = 1, \ldots, K$, $i = 1, \ldots, N$, $j = 0, \ldots, N$, denotes the number of packets generated by a sensing node k and transmitted from node i to node j.

Ideally, we would like to select K sensing nodes in the network such that they are approximately equidistant from each other while sensing data at the required precision of query Q. Note that if the nodes are randomly deployed, we may not always find a sensing node that is located exactly I/K units away from the previous sensing node. Moreover, as discussed in the previous section, it may not be optimal to choose the same sensing nodes for every query over time. Therefore, we allow the distance between two successive sensing nodes to change between $I/K * (1 - \epsilon)$ and $I/K * (1 + \epsilon)$. In addition, we require that the first and last sensing nodes are located within $I/K * (1 + \epsilon)$ units of the lower and upper bounds of the measurement interval, respectively. (In our computational experiments we set $\epsilon = 0.1$.) In order to impose these constraints on the relative locations of the sensing nodes, we construct the sets $S_L = \{i \, | \, x_i \leq \frac{I}{K} * (1 + \epsilon)\}$, $S_i = \{j \, | \, x_i + \frac{I}{K} * (1 - \epsilon) \leq x_j \leq x_i + \frac{I}{K} * (1 + \epsilon)\}$, $i = 1, \ldots, N$, and $S_U = \{i \, | \, x_i \geq I - \frac{I}{K} * (1 + \epsilon)\}$. We also define the binary variables s_i^k, $i = 1, \ldots, N$, $k = 1, \ldots, K$, where $s_i^k = 1$ if node i is the kth sensing node, and $s_i^k = 0$ otherwise. Finally, let L_i, $i = 1, \ldots N$, be a variable that denotes the residual energy level of node i after Q is processed and $W = \min_i L_i$ be the final, i.e., residual, energy of the node with the smallest final energy in the network. Then, we formulate the multi-objective residual energy optimization problem OPT-RE as given in (1)-(11) below.

$$\max\{z_1 = W, \; z_2 = \sum_{i=1}^{N} L_i\} \tag{1}$$

$$L_i \geq W \quad \forall i \tag{2}$$

$$E_i - \xi \sum_{k=1}^{K} s_i^k - \sum_{k=1}^{K}\sum_{j=0}^{N} e_{ij} f_{ij}^k - e_r \sum_{k=1}^{K}\sum_{j=1}^{N} f_{ji}^k - L_i = 0 \quad \forall i \tag{3}$$

$$\sum_{j=0}^{N} f_{ij}^k - \sum_{j=1}^{N} f_{ji}^k - s_i^k = 0 \quad \forall i \neq 0, \forall k \tag{4}$$

$$-\sum_{i=1}^{N} f_{i0}^k = -1 \quad \forall k \tag{5}$$

$$\sum_{i \in S_L} s_i^1 = 1 \tag{6}$$

$$s_i^k \leq \sum_{j \in S_i} s_j^{k+1} \quad \forall i, \; \forall k \neq K \tag{7}$$

$$\sum_{i \in S_U} s_i^K = 1 \tag{8}$$

$$L_i \geq 0 \quad \forall i \tag{9}$$

$$f_{ij}^k \in \{0, \; 1\} \quad \forall i, j, k \tag{10}$$

$$s_i^k \in \{0, \; 1\} \quad \forall i, k \tag{11}$$

The multi-objective function (1) reflects our two goals after the query Q is processed: maximize the residual energy of the node with the minimum residual energy (objective z_1) while also maximizing the total residual energy of the network (objective z_2). There exists a trade-off between these two objectives, and we will further elaborate on this issue below. The constraints (3) relate the initial energy levels E_i to the final energy levels L_i. The constraints (4) represent the flow conservation constraints for nodes $i = 1, \ldots, N$ assuming that a single packet is generated by a sensing node for each query. At a sensing node i, i.e., when there exists some k such that $s_i^k = 1$, the net flow out of the node is equal to 1 which corresponds to the packet generated at this node. Otherwise, when node i is a relay node, i.e., when $s_{ik} = 0 \; \forall k$, the net flow at node i is zero. The constraints (5) ensure that exactly one packet arrives at the monitoring station from each sensing node. Constraints (6)-(8) select K sensing nodes in the network such that they are approximately equidistant from each other while sensing data at the required precision of query Q.

In order to solve OPT-RE, we need to specify the exact relationship between the two objectives z_1 and z_2 in (1). The following case illustrates why we need a multi-objective model for this network lifetime maximization problem and proposes a way of solving OPT-RE. Assume that we need to process a given

query Q for m times and our only objective is to maximize the total residual energy after Q_m is processed. Clearly, if we determine the optimal sensing nodes and the optimal routes to the monitoring station such that the total energy consumption for processing the first query Q_1 is minimized, then the optimal sensing nodes and routes for processing the queries $Q_2, \ldots, Q_m$ are exactly the same unless a node dies while one of these queries is being processed. In other words, maximizing z_2 subject to the constraints (3)-(11) yields a solution that uses the same nodes over and over again until some nodes exhaust their total energy.

Obviously, this approach does not lead to an increased network lifetime, and we must make sure that we do not overuse a subset of the nodes in the network over time. Thus, we implement the following multi-objective approach: First, we maximize z_1 subject to the constraints (2)-(11) given a query Q_t and the initial energy levels E_i^t, $i = 1, \ldots, n$, and obtain the optimal minimum energy W^* in the network. Then, for the same query Q_t and the same initial energy levels E_i^t, $i = 1, \ldots, n$, we maximize z_2 subject to the constraints (2)-(11) except that we replace the right hand side of (2) with W^*. In other words, we maximize the total residual energy subject to the additional constraint that no node can have a residual energy less than W^* after Q_t is processed. The optimal solution of this second single-objective optimization problem yields us the optimal locations of the sensing nodes and the optimal routes to the monitoring station for query Q_t along with the final energy levels L_i^t of the nodes in the network. Then, we set $E_i^{t+1} = L_i^t$, $i = 1, \ldots, n$ and solve this multi-objective optimization problem for the next query Q_{t+1}.

The approach described above requires us two solve two mixed integer programs for each query processed, and thus is not practical. Furthermore, this is still a myopic approach for maximizing the network lifetime because when we select the sensing nodes and the optimal routes for a query Q_t, we completely ignore the queries to be processed in the future. Nevertheless, we expect this mathematical programming based heuristic to provide a good solution in general. In the next section, we propose another heuristic that is easier to implement and computationally much faster. In Section 5, we compare these two heuristics for practical size networks and demonstrate the effectiveness of both algorithms.

4   Joint Node Selection and Routing Algorithm

Due to the computational complexity of OPT-RE, we propose a second algorithm with a simpler heuristic, which determines the set of sensing nodes as well as the paths over which the sensor data is carried to the base station, while solving the lifetime maximization problem. Our new heuristic considers a new link cost function c_{ij} for link (i, j) with four parameters: the energy expenditure for transmitting a unit size packet over the link, e_{ij}; the initial energies of nodes i and j, E_i, and E_j; and the energy expenditure for the reception of a unit size packet, e_r. Note that the energy expenditure due to radio reception is usually considered as a constant. A good sensing node and path choice should consume

as little total amount of energy as possible and should avoid nodes with small residual energy. Therefore, the link cost function should be such that when the transmitting and receiving nodes have plenty of residual energy, the energy expenditure terms are emphasized, and if the residual energies of the transmitting and receiving nodes become small, the residual energy terms dominate the link cost. Thus, we propose a link cost of the form

$$c_{ij} = e_{ij}/E_i + e_r/E_j. \tag{12}$$

The path cost is computed by the summation of the link costs on the path, and the shortest path can be found by using any of the existing shortest path (SP) algorithms (e.g., Dijkstra's algorithm). For this reason, this new heuristic approach is named as SP-RE.

The general description of the node selection and routing algorithm for a given query is given in Figure 1.

Graph $G = (V, A)$ is given where V is the set of sensor nodes and $(l, m) \in A$ is the edge connecting nodes l and m.
1. Initially set $k = 1$, and select K sensing nodes. While $k \leq K$:
   - Label each link (l, m) in A, $c_{lm} = e_{lm}/E_l + e_r/E_m$.
   - Determine the shortest paths from each node in V to the sink node. Order the costs of the shortest paths in non-decreasing order, where $SP_i = \{\text{sink}, g_1^i, g_2^i, \ldots, g_{|SP_i|-2}^i, n_i\}$ is the ith lowest cost shortest path, and let n_i denote the sensing node corresponding to SP_i.
   - $i = 1$: while $s_{n_i}^k = 1$ does not satisfy constraints (6)-(8), $i = i + 1$.
   - Select n_i as the sensing node, set $s_{n_i}^k = 1$, and update the link costs of all nodes on the path SP_i with new residual energy values where $E'_{n_i} = E_{n_i} - \xi - e_{n_i, g_{|SP_i|-2}^i}$ for the sensing node and $E'_{g_t^i} = E_{g_t^i} - e_{g_t^i, g_{t-1}^i} - e_r$ for the relay nodes.
2. $k = k + 1$ and go to step 1.

Fig. 1. Pseudocode for SP-RE.

In the first iteration, the monitoring station determines the shortest cost paths to every node in the network. Then, among all nodes in the network the node with the shortest cost path to the destination is selected to sense and the sensor data flows over its shortest cost path to the destination. The link metrics are updated according to this decision. In the next iteration, the monitoring station again determines the shortest cost paths to every node in the network, but this time with the updated link metrics. If the location of the node with the shortest cost path to the monitoring station does not satisfy the constraints (6)-(8), then we examine the node with the next shortest path to the destination and continue until we determine the next sensing node that satisfies the conditions (6)-(8) on the relative locations of the sensing nodes. We continue to iterate until the requested number of sensor nodes are selected.

5   Performance Analysis

In this section, we first present the comparison of the performances of OPT-RE and SP-RE algorithms. Figure 2 shows the variation of minimum energy (E_{min}) and average energy (E_{avg}) versus time measured in a network consisting of 200 nodes randomly deployed over a line of 100 meters. Queries are sent every second and the measurements are taken every 50 meters, i.e., $K = 2$, with 2 sensing nodes. Note that the locations of the nodes are static, and the same network topology is used for both OPT-RE and SP-RE. The initial battery capacities of the nodes are randomly selected between 0.6 and 1.4 joules. Again, the same battery capacities are used for both algorithms. The node energies are decremented according to the Crossbow specifications [9]. Only the energies consumed due to the sensing and radio operations are included in the performance evaluations.

The solution for OPT-RE is obtained by using CPLEX and OPLStudio [10], which determines the sensing nodes, routes to the base station, the energy expenditures and the residual energy levels of all nodes. The solution is updated every 3600 queries (which corresponds to updating the sensor and path selections every hour), and E_{min} and E_{avg} are determined until the first node exhausts its energy. The heuristic SP-RE runs Dijkstra's algorithm with the link costs given in (12) for computing the shortest paths, and the solution is updated every hour.

In Figure 2, we observe that the network lifetime achieved by SP-RE is quite close to that attained by OPT-RE; the difference is less than 9%. Thus, SP-RE appears as a promising joint sensor selection and routing algorithm that performs similarly to more complicated optimization-based algorithms. When we examine Figure 2 more carefully, we observe certain characteristics of the two algorithms. With SP-RE, the minimum node energy drops quickly after several hours of operation in contrast to OPT-RE, where the algorithm keeps the minimum node energy constant for a long time. Meanwhile, the average node energy with OPT-RE drops below the average node energy attained by SP-RE. The reason is clear; unlike SP-RE, the objective of OPT-RE is to maximize the minimum residual node energy. Thus, OPT-RE tries to choose nodes with energies above the minimum energy level for sensing and routing before re-using nodes with minimum energy. In doing so, the average energy in the network drops quickly, since most of the nodes in the network tend to have similar residual energies.

Unfortunately, the required running time with OPT-RE prohibits us to perform the rest of the experiments with OPT-RE. Thus, in the following, we compare the network lifetime achieved by SP-RE with other non-optimization-based heuristic approaches; TinyDB and TinyDB-RE. We believe that this is acceptable considering Figure 2, where it is shown that SP-RE closely matches the network lifetime achieved by OPT-RE. The first method TinyDB [8] is the standard query and routing algorithm used in Crossbow motes. In TinyDB, each node transmits at a predetermined, fixed constant power level, and the nodes separately determine the shortest route to the base station. In this case, sensor selection is done only based on the physical proximity to the desired measurement point. The second algorithm, TinyDB-RE computes the routes to the base station in the same way as TinyDB; however, the node with the highest residual
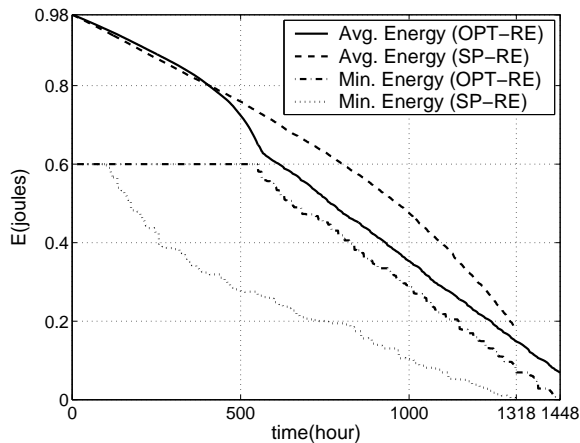
Fig. 2. Residual energy of nodes with OPT-RE and SP-RE.

energy within the required proximity to the measurement point is selected for sensing.

Figure 3 shows the variation of average and minimum energy levels measured over the lifetime of the network using SP-RE, TinyDB and TinyDB-RE. There are 100 nodes distributed over a line of 100 meters, and queries require a measurement every 50 meters. All nodes have the same initial energy of 1.4 joules. The query is repeated every 1 second, whereas the sensing node and path selection is updated every hour (every 3600 queries). As depicted in Figure 3, the lifetime of the network using SP-RE is around 330 hours. The simulations indicate that the other two methods perform poorly as compared to SP-RE, with an order of magnitude decrease in the lifetime. More specifically, when TinyDB is used, the lifetime of the network is measured as 7 hours. TinyDB-RE improves the lifetime to around 15 hours, with intelligent sensor selection. Nevertheless, since both algorithms apply constant transmission power, the algorithm SP-RE significantly enhances performance with smart selection of power levels and routing.

Next, we consider several scenarios with different node densities and query precisions, and compare the lifetimes achieved by three methods. First, we look into how the network lifetime is influenced by the increasing node density. Figure 4 depicts the measured lifetime with the three algorithms for values of N ranging between 10 and 600 where the nodes are placed randomly over a line of 100 meters. The sampling interval is set to 20 meters, i.e., $K = 5$. From Figure 4, we can infer that the lifetime with TinyDB or TinyDB-RE is not affected by the increasing node density, resulting in a lifetime of approximately 20 hours. Note that TinyDB performs the sensor node selections only once based on the physical proximity to the measurement points, and the routes of packets from those sensing nodes to the base station remain unchanged. When the algorithm TinyDB-RE is used, the sensing nodes can be changed according to the residual
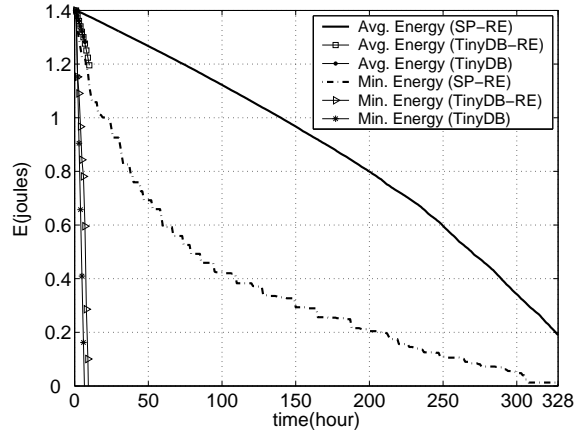
Fig. 3. Lifetime comparison of SP-RE with TinyDB and TinyDB-RE.

energy levels. However, the routing of sensor data is still not performed based on residual energies. Finally, when SP-RE is employed, both the sensing node and transmission power selections are performed dynamically, and routes to the base station are updated adaptively. Increasing the node density provides alternate sensing nodes and paths to the sink, and thus the lifetime is increased significantly. In our experiments, the measured lifetime attained by the algorithm SP-RE is about 2000 hours which is much higher those achieved by other methods.

In Figure 5, we evaluate the variation of lifetime with respect to the sampling interval for a network with 500 nodes uniformly distributed in an interval of 100 meters, and compare the behavior of the three algorithms. Our experiments show that the lifetime of the network remains the same when TinyDB is used, independent of the network and the sizes of the sampling intervals. The performance of TinyDB-RE is improved with the increased sampling interval to about the twice of the lifetime achieved by TinyDB. This is due to the fact that TinyDB-RE selects the sensor nodes based on residual energies, and increasing the sampling interval increases the choices for sensing nodes. Meanwhile, the lifetime attained by SP-RE increases significantly with increasing sampling interval. This behavior is especially apparent in dense networks such as the one considered in this experiment.

The computational experiments show that SP-RE is definitely a promising algorithm as compared to both optimization-based and naive methods for enhancing the lifetime of sensor networks, due to the incorporated features such as intelligent sensor selection, transmission power control and data routing. These experiments, although preliminary, provide us some insight about the extent of the advantages that SP-RE can provide. Our results show that these advan-
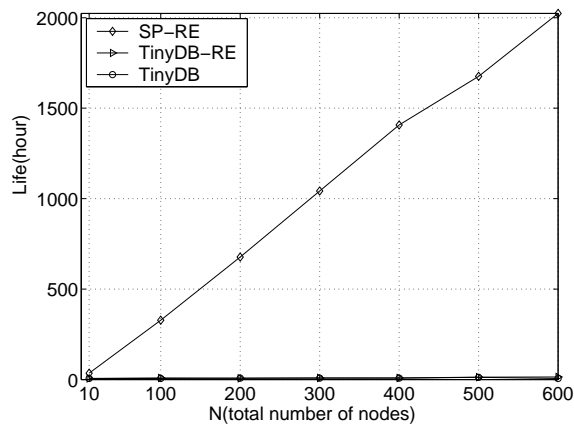
Fig. 4. Lifetime of SP-RE, TinyDB and TinyDB-RE with respect to varying node density (sampling interval = 20 meters).

tages become more significant as the network density is increased, which is quite valuable for environment monitoring applications.

6 Conclusions and Future Work

In this paper, the trade off between measurement precision and energy efficiency is explored through the lifetime of a sensor network. The goal is to maximize the network lifetime by determining the set of sensing nodes and the radio transmission powers for all participating nodes, i.e., network topology, while satisfying the required measurement precision. A mathematical programming based solution and a low complexity heuristic algorithm are proposed for the energy-precision optimization problem. Our results indicate that with only 10% of deviation from the requested exact measurement intervals, the improvement in network lifetime using our algorithms is significant compared to the algorithms currently available in the literature.

However, in this paper we solved only a part of the energy-efficient monitoring problem: We assumed that a user is interested in equally spaced measurements. In a more general setting, a user may be interested in non-uniform measurements to depict a closer approximation of the current conditions in the field. This problem has the promise to bring together signal processing and networking fields, since the current conditions in the field can be viewed as a 2-D image that is being encoded by the nodes in sensor network in an efficient way. One of our future goals is to further investigate this problem.

Another future extension of our work is to consider the energy consumption due to MAC; collisions and retransmissions in a realistic wireless network environment may cause increased consumption of energy.
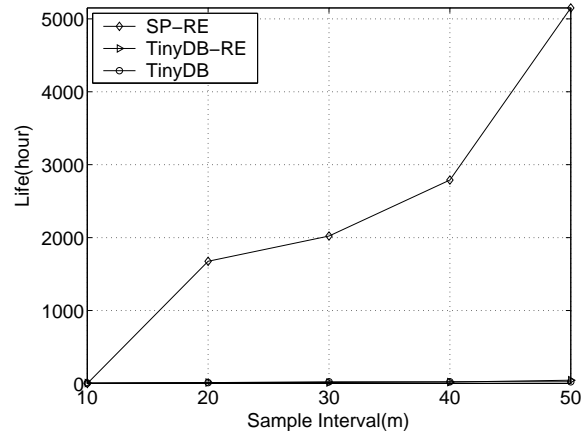
Fig. 5. Lifetime of SP-RE, TinyDB and TinyDB-RE with respect to varying sampling intervals.

References

1. A. Boulis, S. Ganeriwal, and M.B. Srivastava, "Aggregation in Sensor Networks: An Energy-Accuracy Trade-off. Sensor Network Protocols and Applications," SNPA 2003, 11 May 2003.
2. A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," *Proc. of ACM International Workshop on Wireless Sensor Networks and Applications, WSNA '02*, Atlanta, GA, September 2002.
3. S. N. Simic and S. Sastry, "Distributed Environmental Monitoring Using Random Sensor Networks," *Proc. of IPSN'03*, Palo Alto, CA.
4. N. Li and J. C. Hou, "Topology Control in Heterogeneous Wireless Networks: Problems and Solutions," Proc. IEEE INFOCOM 2004, March 2004
5. S. Narayanaswamy, V. Kawadia, R. S. Sreenivas, and P. R. Kumar,"Power Control in Ad-Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW Protocol," in Proceedings of European Wireless 2002.
6. K. Kalpakis, V. Puttagunta and Parag Namjoshi,"Accuracy Vs Lifetime: Linear Sketches for Appoximate Aggregate Range Queries in Sensor Networks," Department of Computer Science Technical Report, TR CS-04-04, University of Maryland, Baltimore County, 2004.
7. E. D. Kaplan, Ed., "Understanding GPS Principles and Applications.Norwood," MA: Artech House, 1996.
8. S. Madden, J. Hellerstein, W. Hong, "TinyDB: In-Network Query Processing in TinyOS," Sep 2003.
9. http://www.xbow.com, 2004.
10. http://www.ilog.com, 2004.