Energy Efficient Routing in Wireless Sensor Networks

Mehdi Kalantari and Mark Shayman
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
{mehkalan, shayman}@eng.umd.edu

Abstract

In this paper we introduce a new scheme for the purpose of routing in the wireless sensor networks. Our proposed approach is for the case in which many sensors need to collect data and send it to a central node. We will show that in order to find the routes that give energy efficiency, we can solve a set of partial differential equations similar to the Maxwell's equations in the electrostatic theory. These partial differential equations give the geographical paths from each sensor to the destination. In order to find the actual routes, we approximate the found paths by a sequence of wireless links each between a pair of sensors. Our simulation results show considerable improvement in the life of the network compared to the traditional shortest path approach.

I. Introduction

The wireless sensor networks have been studied extensively in the recent years. Such networks are made of several hundred to several thousand of sensors propagated in a geographical area. There are many different applications for such networks including military, environment monitoring, agriculture, transportation control, disaster, fire fighting and protection, and home applications. Sensors are very simple identical electronic devices equipped with a processor and small storage memory and a communication channel. The sensors can communicate to each other through wireless links, and most of the times they use radio frequency channels for the purpose of communication.

In many applications the sensors perform measurements of specific metrics like temperature, pressure, movements or other physical values in a periodic or non-periodic way. Most of the times it is desired to collect the data of all sensors in a specific station for processing, archiving and other purposes. This station is a data sink, and it has enough processing power, storage space, and capability of communicating to the sensors. We will call this station the *central node* in the rest of the paper. For the purpose of communication to the central node, the sensors relay the packets of each other in a multi-hop way.

Since the sensors operate on the battery power, it is very important to make efficient use of energy of sensors to increase the lifetime of the network. Most of the energy of a sensors is spent for transmission of the data packets generated by that sensor or relaying the packets of the other sensors, so finding optimal transmission paths from each sensor to the destination is a very important task.

The routing problem in the sensor networks has been studied by many researchers. Sequential Assignment Routing (SAR) is proposed in [5], and it takes into account

This work was partially supported by AFOSR under contract F496200210217.

the energy constraints by making a tree rooted in the central node. The tree starts to grow toward the sensors on the paths with enough residual energy. The routing from each sensor to the central node is based on the structure of the tree. Minimum Cost Forwarding Algorithm for Large Sensor Networks is proposed in [11]. In this approach, each sensor maintains the least cost from it to the central node. For the transmission of a packet, it is broadcasted by a sensor, and after receiving a packet, a sensor checks of it is on the least cost path of the source sensor to the central node. If it is so, it retransmits that packet. Similar routing schemes can be found in [8], [10] and [9]. Good surveys of the sensor networks have been given in [7] and [6].

In this paper we propose an energy efficient routing scheme for the sensor networks. Our energy efficiency is based on matching the routes to energy constraints in order to increase the network life. When the energy of the sensors in some area of the network is low due to heavy communication activity in the past, we will increase the cost of routing through this area to protect the sensors from early energy depletion. We will introduce a mathematical machinery based on partial differential equations very similar to the Maxwell's equations in the electrostatic theory. The routes are found based on the solution of this partial differential equations. In our formulation, the sensors are sources of information, and they are similar to the positive charges in the electromagnetic, the central node is the sink of information and it is similar to a negative charge, and the network is like a non-homogeneous dielectric media with variable dielectric constant (or permittivity coefficient). Our routing scheme is based on changing the permittivity factor to a higher value in the places in the network where we have a high residual energy of the nodes, and set it to a low value for the places of the network that the nodes do not have much energy left. Our simulations show that our method gives a significant increase in the network life compared to the shortest path scheme.

In the next section, we will introduce the mathematical formulation of our method. In Section III we will use the introduced formulation to develop our energy efficient routing scheme. Section IV shows the results of our simulation experiments.

II. Mathematical Formulation

Consider a network of N wireless sensors that can communicate with each other through radio links. The sensors are placed in a region A in the plane, and they are intended to collect information about the events in the area of the network. Each sensor is responsible for the events happening in its neighborhood. All messages are desired to be collected in a central node. When an event happens at some place in the network, the closest sensor to the place of the event generates a message. All messages should be sent to the central node, which is assumed to have enough storage, energy and processing power. Furthermore, we make the following assumptions:

A1: Each sensor has a limited amount of energy, and the residual energy of the sensors is known at every time.

A2: The events in the geographical area of the network happen with a known spatial density rate denoted by $r(z) \geq 0$ for the place z. This quantity means that for the area $a \subseteq A$ the rate of events that happens inside a is:

$$w(s) = \int_a r(z)\,\mathrm{ds} \tag{1}$$

in which integration is over area a, and ds is a differential area containing z.

A3: The sensors are not mobile and their locations are known.

A4: The sensor placement in the network is such that the cental node can be reached from every sensor in the network by a sequence of multi-hop transmissions.

A5: For the purpose of routing, we keep a direction for every point z of the network. A sensor placed at the point z uses the direction defined for this point, and it forwards its traffic to sensors that are closest to this direction as the next hop. We assume that this direction is a continues function of z. In other words, the very close sensors either use the same next hops, or they use the next hops that are very close to each other.

The above set of assumptions is not restrictive from a practical point of view; the assumption of the knowledge of residual energy of each sensor can be justified by the fact that if we know the initial energy of a sensor at the start of a time interval and its communication activity in that interval, the residual energy of that sensor can be found at the end of that interval. Note that both of these information ore observable to the central node. The assumption of knowing the locations does not require GPS devices. We will see that we need the location of the nodes relative to the central node, and since the sensors are not mobile, the location of each node can be set at the installation of the network. The value of $r(z)$ may be unknown at the start of the network, but it can be estimated by the central node based on the statistics of the received messages from the sensors a different locations.

Based on Assumption **A2**, the rate of messages generated by each sensor can be found in terms of the area in which that sensor collects events. Let $w_i > 0$ denotes the average rate of the messages that are generated by sensor i; if t_i denotes the area in which sensor i is responsible for collecting events, then w_i can be found by using equation (1), and plugging t_i for a. Note that we have assumed that every event is picked up by only one sensor. If many sensors generate messages corresponding to an event, we can assume that only one of the sensors takes the responsibility of reporting that event; this can be achieved in different ways: for example, among all the sensors that sense an event, the sensor with the lowest ID can take the responsibility of sending the event, or the sensors wait a small random time before sending the event and after hearing the transmission of the message of an event, the other sensors in the neighborhood avoid sending the message of that event.

We also assign a weight to the central node denoted by w_0 and define it in the following way:

$$w_0 = -\sum_{i=}^{N} w_i \tag{2}$$

The purpose of defining this weight for the central node is that we want to assign a positive weight to the all sources, and a negative weight to the destination. And since all the traffic of the sources is collected at the destination, we make the above definition for w_0.

Assume for each sensor a path in the plane is chosen to send the messages of that sensor to the central node. Mathematically, we define a path as a directed curved line that starts at a sensor and ends at the central node. Therefore for the N sensors of the network, we have N paths. Let p_i denotes the path for the sensor i. The amount of load on each path is the rate of generated messages at the sensor located at the start of the path (i.e., w_i). So we define the weight of p_i to be w_i.

It should be noted that the chosen paths are not constrained by the location of intermediate nodes. Instead, the paths are 'abstract' paths in the plane that represent desired paths for the transmission of messages. For communication to occur, we need to define the routes in terms of the paths. For this purpose, each abstract path is approximated by an actual path route consisting of a piecewise linear multi-hop path connecting the source and destination through a sequence of intermediate sensors. We assume the sensors in the network are densely distributed, such that we can always make the above approximation.

Given a set of (abstract) paths for each sensor to the destination, we define a vector field on A which we refer to as the *load density* vector field and denote by $\vec{D}$. This vector field represents the flux density of the paths to the destination. Given a point $z \in A$, we choose a small area element at z. For each path that intersects S, we take the tangent vector to the path and scale it so it has magnitude equal to the weight of the path. Adding up these scaled tangent vectors, dividing by the area of S, and letting the area element go to zero gives the value of $\vec{D}(z)$. In the other words:

$$\vec{D}(z) = \lim_{|S| \to 0} \frac{1}{|S|} \sum_{p_i \cap S \neq \emptyset} w_i \hat{l}_i \tag{3}$$

in which S is a connected area in the network, and $\hat{l}_i$ is a unit vector tangent to p_i at S, and pointing toward the direction of p_i that goes to the destination. The above definition has been illustrated in Figure 1. It should be noted that when $|S| \to 0$, all paths that pass through S will have the same $\hat{l}_i$ based on Assumption **A5**; hence the vectors that are summed up in equation (3) have the same direction; in other words, $|\vec{D}(z)|$ will be the sum of the weights of the paths that pass through S. So $|\vec{D}(z)|$ represents the actual amount of communication activity at point z.

Mathematically, if the number of sensors is finite, the values of $\vec{D}$ defined by equation (3) will be zero except for a set of measure 0. In practice we do not need to have a large number of sensors; we can define a small enough lower bound on the value of S depending on the required accuracy of defining $\vec{D}$. For example, the network terrain can be divided into small rectangles via vertical and horizontal grids, and S can be defined as any of these rectangles. In this example the accuracy of $\vec{D}$ depends on the size of rectangles, and the value of this variable will be constant on each rectangle, so we deal with a discrete version of equations and operators. For instance the partial derivative in x direction will be written in terms of the difference of the value of $\vec{D}$ on adjacent horizontal rectangles and the distance between the rectangles. For the sake of simplicity, in the rest of this paper we assume that the size of S is small enough so that we can deal with $\vec{D}$ as a continuous variable.

The definition of $\vec{D}$ given by equation (3) satisfies the following equation:

$$\oint_C \vec{D} \cdot \vec{dn} = w \tag{4}$$

in which the integral is over a closed contour C, $\vec{dn}$ is a differential vector normal to the contour at each point of its boundary, dot represents the inner product of vectors in two dimensional space, and w is the algebraic sum of the weights of the nodes

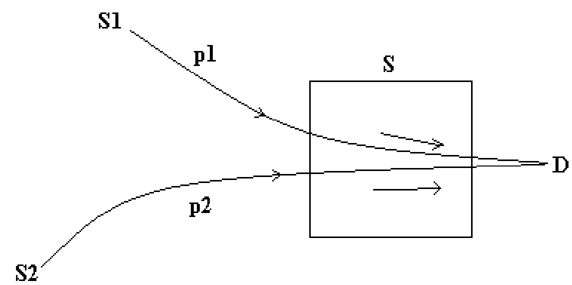

Fig. 1. The illustration of defining the load density vector field based on paths.

inside the closed contour. In calculating w we count the weights of the sources with a positive sign and the weight of the destination with a negative sign (if it is inside the contour). Equation (4) is analogous to Gauss's law in the electrostatic theory.

We say a message enters a contour if it is forwarded from a node outside the contour to a node inside it, and similarly, we say a message exits a counter if the reverse happens. With these definitions, equation (4) has a very simple explanation: the rate at which the messages exit a contour is the algebraic sum of the weights of the nodes that are inside that contour (i.e., the net amount of the sources inside the contour).

Now we define another function ρ that represents the spatial density of rate on which the messages enter the network. This quantity is a function of location, and obviously $\rho(z) = r(z)$ for $z \neq z_0$ in which z_0 is the place of the central node. But since all messages end at the destination, this means that the density of the rate has a Dirac delta form at the place of the central node. In other words,

$$\rho(z) = r(z) + w_0\delta(z - z_0). \tag{5}$$

With the above definition of ρ equation (4) can be expressed in partial differential equation form:

$$\vec{\nabla} \cdot \vec{D}(z) = \rho(z) \tag{6}$$

where $\vec{\nabla}$ is defined as:

$$\vec{\nabla} = \frac{\partial}{\partial x}\hat{i} + \frac{\partial}{\partial y}\hat{j} \tag{7}$$

in which x and y represent the variables in the Cartesian coordinate frames, and $\hat{i}$ and $\hat{j}$ represent the unit vectors along x and y axes respectively.

Depending on how we select the set of paths, the value of $\vec{D}$ is different, but independent of the selection of paths, $\vec{D}$ satisfies the following equations:

$$\begin{cases} \vec{\nabla} \cdot \vec{D} = \rho \\ D_n(z) = 0 \quad for\ z \in Boundary\ of\ A \end{cases} \tag{8}$$

in which A denotes the geographical set that contains the network and $D_n(z)$ denotes the normal component of $\vec{D}$ on the boundary of A. The first equation in (8) is the natural limitation imposed by the fact that all the traffic generated at the network should be delivered to the destination. The second equation comes from the fact that no load is desired to exit the geographical area of the network or enter into it through the boundary. It is important to notice that equations (8) do not give $\vec{D}$ uniquely.

Conversely, if we have a $\vec{D}$ that satisfies equations (8), we can find the paths that can be used to send the traffic of sources to the destination. In order to define the routes based on the values of $\vec{D}$, we need to define the concept of *load flow lines*. These lines are similar to the electric flux lines in electrostatic theory [1] [2][4]. The load flow lines are a family of two dimensional curved lines that are always tangent to the direction of the $\vec{D}$ and their orientation is the same as the orientation of the $\vec{D}$. The load flow lines cannot intersect except at a source or the destination; if they intersect, at the point of intersection the direction of the field would not be single-valued. The other property of the load flow lines is that these lines always start at the sources and end at the destination; this fact is because the value of divergence in equations (8) is positive at the sources, and it is negative only at the destination.

Based on the definition of the load flow lines, the path corresponding to each sensor can be easily found: to find p_i, we start at the place of sensor i and follow the direction of the load flow lines. The paths generated by this scheme can only end at the destination. Mathematically, the path starts at the sensor i, and it is always tangent to the direction of $\vec{D}$. It is straightforward to see that if we plug the paths generated by this method in equation (3), we will get the original value of $\vec{D}$.

III. Energy Efficient Routing

In the previous section we established the basic concept of load vector field, and described its connection to routing. Thus, given $\vec{D}$, we can obtain paths and based on the paths we will find the routes to the destination. However, equations (8) do not specify $\vec{D}$ uniquely. The remaining issue is to decide what additional condition(s) to place on $\vec{D}$ so the resulting vector field generates a desirable set of routes.

In this section, we try to find the paths that give energy efficiency. We define the following scalar field for the residual energy of the network as a function of z:

$$\omega(z,t) = \lim_{|S|\to 0} \frac{1}{|S|} \sum_{sensor i \in S} e_i(t) \tag{9}$$

in which $e_i(t)$ denotes the residual energy of sensor i at time t. Like for the case of defining $\vec{D}$, the above definition implies that $\omega(z,t)$ is zero except for a set of measure 0 in the z variable. We solve this problem exactly in the same way as we explained it for $\vec{D}$.

We approach the problem of finding energy efficient routes in two steps: in the first step, we assume that initial distribution of the energy in the network is uniform (i.e., $\omega(z,0) = c$). In the second step, we remove this assumption and try to find the paths that give the best performance given an arbitrary distribution of the residual energy in the network.

A. Uniform Initial Residual Energy

For this case, we assume that the energy is distributed uniformly at the start, and so we try to make the communication load as even as possible to make an even decrease in the residual energy of the sensors.

In order to find the routes that give energy efficiency, we use the fact that the communication activity at place z of the network is $|\vec{D}(z)|$. In other words, at place z, $|\vec{D}(z)|$ transmissions are done per unit of time. This means that the rate of decrease of energy at z is proportional to $|\vec{D}(z)|$. No transmissions will be possible through the places with zero energy. In order to find energy efficiency, the intuition we follow is that by making $|\vec{D}|$ as uniform as possible, we will obtain routes that will cause the traffic to be highly dispersed throughout the network. So we avoid over-utilization of energy at some place of the network that causes early death of sensors, while some other place of the network is under-utilized.

The uniform load distribution can be formulated as minimizing the following cost function:

$$J(\vec{D}) = \int_A \left|(\vec{D} - \vec{D}_{av})\right|^2 \mathrm{ds} \tag{10}$$

in which $\vec{X_{av}}$ is the average value of the vector field $\vec{X}$ on the set A, and it can simply be defined as:

$$\vec{X}_{av} = \frac{1}{|A|} \int_A \vec{X}\, \mathrm{ds}. \tag{11}$$

The quadratic form of the cost function in equation (10) causes the load to be distributed as evenly as possible. It prevents having high loads somewhere in the network while the resources are under-utilized somewhere else. One interesting fact about this cost function is that it is similar to the definition of energy in electrostatic theory. The above optimization problem can be summarized as:

$$\begin{array}{l} \text{Minimize } J(\vec{D}) = \int_A \left|(\vec{D} - \vec{D}_{av})\right|^2 \mathrm{ds} \\ \text{Subject to:} \\ \vec{\nabla} \cdot \vec{D} = \rho \\ D_n(z) = 0 \;\; z \in \text{Boundary of } A \end{array} \tag{12}$$

The following lemma provides the key to finding the solution of the optimization problem defined by (12).

Lemma 1: If $\vec{D^*}$ denotes the optimal solution of equation (12), then it satisfies:

$$\vec{\nabla} \times \vec{D^*} = 0 \tag{13}$$

In the above equation $\vec{\nabla}\times$ is the two dimensional curl operator, and it is defined in the following way for a vector field $\vec{F} = [F_x \; F_y]$:

$$\vec{\nabla} \times \vec{F} = \left(-\frac{\partial F_x}{\partial y} + \frac{\partial F_y}{\partial x}\right)\hat{k} \tag{14}$$

in which $\hat{k}$ is a unit vector perpendicular to $\hat{\imath}$ and $\hat{\jmath}$. More precisely, $\hat{k} = \hat{\imath} \times \hat{\jmath}$.

The proof of Lemma 1 is given in the Appendix. Based on the result of this lemma, we can write a set of partial differential equations for the optimal $\vec{D^*}$:

$$\vec{\nabla} \cdot \vec{D^*} = \rho \qquad \vec{\nabla} \times \vec{D^*} = 0 \tag{15}$$

The above equations are similar to Maxwell's equations in the electrostatic theory. In the theory of partial differential equations it is proved that the above equations along with the boundary condition given by (8) give $\vec{D}^*$ uniquely.

Mathematically, a vector field for which $\vec{\nabla} \times \vec{D} = 0$ is called a conservative vector field. It is proved that such a vector field can be expressed as the gradient of a scalar field. In other words:

$$\vec{D} = \vec{\nabla} U \tag{16}$$

in which U is a scalar function known as the potential function. Then the set of equations defined by (15) reduces to:

$$\nabla^2 U = \rho \tag{17}$$

in which the operator ∇^2 is defined as:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \tag{18}$$

The boundary conditions for $\vec{D}$ implies that the gradient of U is zero on the boundary along the direction that is normal to the boundary. In other words:

$$\vec{\nabla} U(z) \cdot \hat{n}(z) = 0 \qquad z \in \text{Boundary of } A \tag{19}$$

in which $\hat{n}(z)$ is a unit vector normal to the boundary.

The partial differential equation defined by (17) is known as the Poisson equation. The potential function found as the solution of this equation has very nice interpretations. Firstly, the potential function gives a rough idea of how much effort by the network is needed to send data from a source to the destination. This effort is proportional to the potential difference of the source and the destination. Secondly, the potential function gives insight into the routing. Based on equation (16), the routing is done in the direction of the gradient of the potential function. Some concerns like the possibility of forming routing loops are naturally avoided since the potential function changes strictly monotonic in the nodes that form a path from the source to the destination.

B. Non-Uniform Initial Residual Energy

So far we have solved the problem of finding paths that give energy efficiency for the case in which the initial residual energy is uniform. However, this assumption is not realistic in practice. Since all sensors try to send their messages to the central node, the rate of energy utilization at the sensors in the neighborhood of the central nodes is higher than the other nodes. So it is logical to have a higher residual energy density (i.e., $\omega(z, 0)$) at the neighborhood of the central node in the network design process. This goal can be achieved either by having more energy in the sensors that are closer to the central node, or if the sensors are identical, we can have a higher density of the them in the neighborhood of the central node; obviously, when the density of the nodes increases, the density of the residual energy increases accordingly since the nodes have the same amount of initial energy.

Another issue that should be noted is that due to stochastic nature of the generation of events, it might happen that the energy of some sensors is depleted faster than the others. So we can update paths after a while to take this issue into consideration and make less utilization of the sensors with low residual energy to save them for a longer time.

To deal with this problem, we make a change in the optimization problem of equation (10):

$$\begin{array}{l} \text{Minimize } J = \int_A K(z)|\vec{D(z)} - \vec{D}_{av}|^2 \, \mathrm{ds} \\ \text{Subject to:} \\ \vec{\nabla} \cdot \vec{D} = \rho(z) \\ D_n(z) = 0 \quad z \in \text{Boundary of } A \end{array} \tag{20}$$

in which $K(z)$ is a scalar weight function that represents where in the network we prefer to have a higher load and where we want to have less communication activity in order to save the energy of the nodes. A high value of $K(z)$ means that we do not want to have a high communication activity at z, and on the other hand a low value of $K(z)$ means that the area around z has enough energy. By changing the relative values of $K(z)$, we can penalize routing on some places versus the other places.

The following lemma gives the key to solve the above optimization problem.

Lemma 2: If $\vec{D^*}$ denotes the optimal solution of equation (20), then it satisfies:

$$\vec{\nabla} \times (K\vec{D^*}) = 0 \tag{21}$$

The proof of this lemma is given in the appendix. Based on the above, we make the following definition:

$$\vec{E^*} = K\vec{D^*} \tag{22}$$

Then we will have the following set of partial differential equations:

$$\vec{\nabla} \cdot \vec{D^*} = \rho \qquad \vec{\nabla} \times \vec{E^*} = 0 \tag{23}$$

The solution of equations (23) gives the optimal value of $\vec{D}$. The boundary conditions for solving (23) are the same boundary conditions that we introduced for $\vec{D}$ before.

The potential function that we defined formerly can be defined for this case in a similar way. Here we have $\vec{\nabla} \times \vec{E^*} = 0$, and mathematically, this means that there is a potential function U such that $\vec{E^*} = \vec{\nabla} U$. This potential function has all the useful properties that we explained before: the direction of its gradient is the direction of optimal $\vec{D}$, and its value at every place represent the amount of routing effort required to deliver the messages from that place to the destination. It can easily be shown that the potential function U satisfies the following partial differential equation for this case:

$$\vec{\nabla}^2 U - \frac{\vec{\nabla} K . \vec{\nabla} U}{K} = K\rho. \tag{24}$$

The boundary condition for U is the same as that in equation (19).

There is one final issue about the way we should choose $K(z)$. This scalar field should be assigned in a such way that it has a high value at the places with low residual

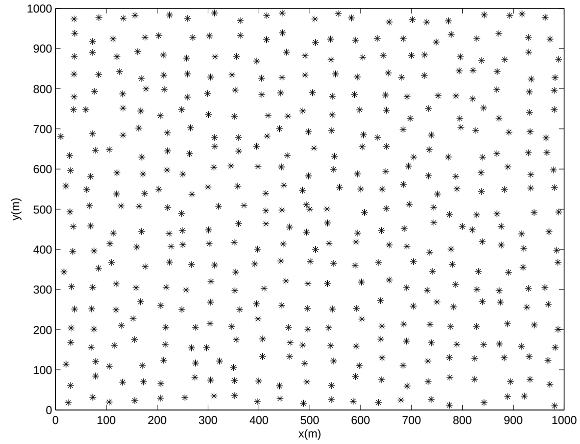

Fig. 2. The placement of sensors in the network terrain

energy. There might be different ways of doing so, but one easy assignment of $K(z)$ can be done in the following way:

$$K(z) = \frac{1}{\omega(z,t)}. \tag{25}$$

As equation (25) shows, $K(z)$ depends on t. In practice, we do not need to change the value of $K(z)$ very frequently, which causes frequent change of the paths and routes. The update of $K(z)$ can be done with a low frequency, and once after a considerable change in the residual energy of the network happens.

IV. Simulation Results

In this section we will show the results of the simulation for the proposed method of distributing the load in the network. In this simulation scenario sensors of the network are distributed in a $1000m \times 1000m$ square. The network area has been partitioned into $21 \times 21 = 441$ equal squares by equally spaced horizontal and vertical grids. The number of sensors $N = 441$, and in each small square, a sensor has been placed randomly. The central node has been placed in the center of the network area. The generation of the events inside each small square is done according to a Poisson process with a rate $0 < \lambda_i < 1$. The sensor inside each small square is responsible for all events that happen inside that square.

We have distributed the initial energy of 20000 units among the sensors. As it was stated before, the nodes closer to the cental node need more energy since they have to do more switching. So the assignment is done such that the initial energy of the sensors is inversely proportional to their distance from the central node. It should be emphasized again that if the sensors are identical in terms of their energy, the assignment of initial residual energy is done by changing the density of the sensors in the network. Each transmission or switching needs one unit of energy, and the transmission range of each sensor is $85m$.

Figure 2 shows the placement of the nodes in the network. The relatively high number of sensors allows us to find routes by approximating paths with the relaying

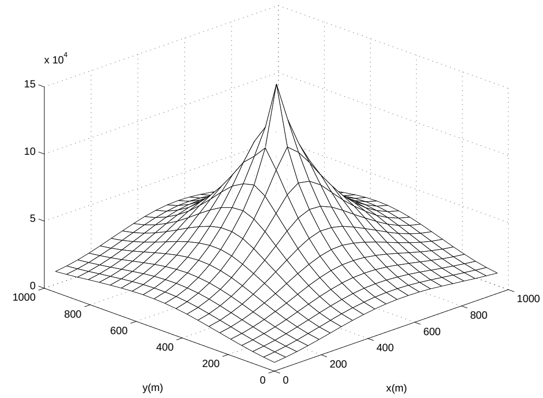

Fig. 3. The value of the potential function U

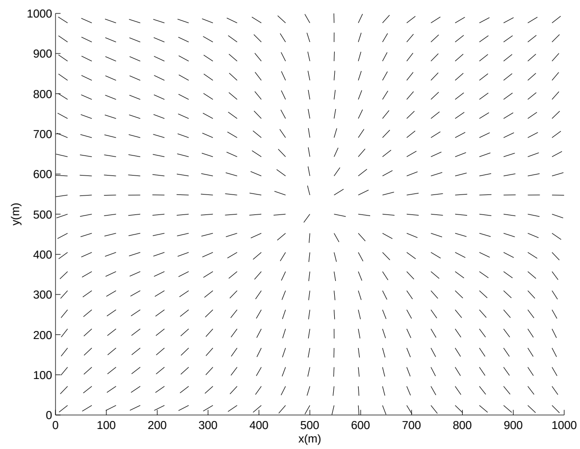

Fig. 4. The direction of $\vec{D}^*$: the line segments show the direction of the $\vec{D}^*$.

sensors.

We have numerically solved the set of partial differential equation given by (24) with the boundary conditions given in equation (19) on the 21×21 grid to find the potential function U. Furthermore, we have made use of equation (25) for finding the value of $K(z)$ in terms of the residual energy. The resulting U is shown in the figure 5. The value of $\vec{E}^*$ is found by taking the gradient of U, and finally $\vec{D}^*$ is calculated from $\vec{E}^*$ by using equation (22). Figure 4 shows the direction of $\vec{D}^*$ at different places of the network. The line segments in this figure show the direction of the optimal load density vector field $\vec{D}^*$. The paths from the sensors to the destination are found by following these segment lines, and the routes are calculated by approximating the paths by the sequence of relaying sensors. The resulting routes from the all sensors to the destination has been plotted in figure 5.

To have a basis of comparison we have also calculated the routes that use the shortest path to the destination. Figure 6 shows the routes calculated by this method. By comparing this figure with figure 5, it can be seen that in the case of using optimal $\vec{D}^*$ the routes are more evenly spaced in the network.

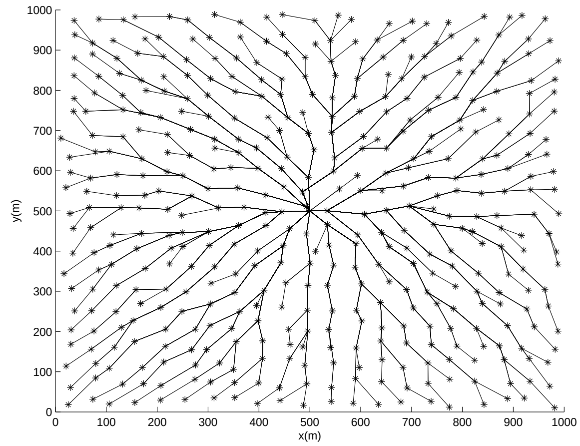

Fig. 5. The routes from all sensors to the destination. These routes are found by using $\vec{D^*}$.

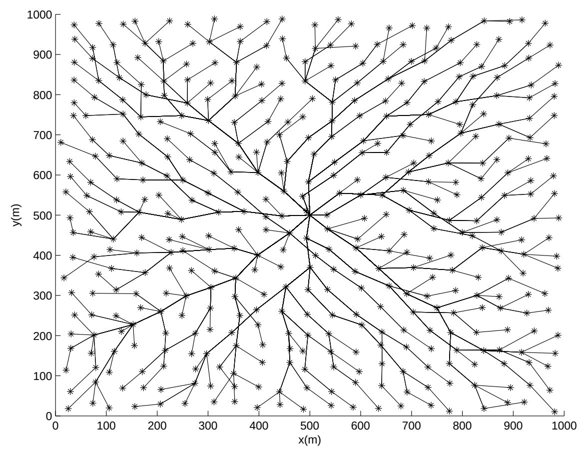

Fig. 6. shortest path routes to the destination.

To evaluate the difference of using $\vec{D^*}$ for routing to the case in which we use the shortest path routes, we turn the network on at $t = 0$, and let it run until the nodes run out of energy and no more communication is possible to the central node. Our simulations show that for the case of using $\vec{D^*}$, the total number of 2112 messages are sent to the destination, and for the case of shortest path routing the total number of delivered messages is 1734.

We have done several other experiments with the same conditions as the above experiment but with different randomly generated locations of the nodes and the traffic sources. The results are shown in Table I. The second column of this table shows the total number of delivered messages for the case in which we use the routes generated by $\vec{D^*}$, and its third column shows the same quantity for the shortest path case. It can be seen that in all cases the number of delivered messages is increased considerably, and the average increase is 27%.

Exp.	$\vec{D}^*$ Routes.	Shortest path	improvement
1	2112	1734	22%
2	1901	1465	30%
3	1928	1681	15%
4	1744	1278	36%
5	1839	1233	49%
6	1761	1592	10%
7	1749	1414	24%
8	1774	1193	49%
9	1918	1437	33%
10	2073	1487	39%
11	1911	1691	13%
12	1725	1452	19%

TABLE I

THE COMPARISON OF THE PERFORMANCE OF THE ROUTES FOUND BASED ON $\vec{D}$ WITH THE SHORTEST PATH ROUTES FOR SEVERAL EXPERIMENTS.

V. CONCLUSION

In this paper we introduced an approach for the purpose of routing in the sensor networks that gives energy efficiency, and increases the network life. The main idea of our routing approach is to find routes that avoid using places of the network that have small residual energy and to make a higher utilization of the places with higher residual energy. We showed that for this purpose a set of partial differential equations similar to the Maxwell's equations in the theory of electrostatic should be solved. By solving these equations, we found the routes that give a considerable improvement in the network performance in terms of energy efficiency and the life of the sensors.

From a practical point of view, the central node can collect all the information like the position of the sensors and the residual energy to find the routes by running our proposed method. The routes can be updated once in a while when a considerable change in the residual energy has occurred.

The set of assumptions that we stated in Section II is not restrictive. We have assumed the knowledge of $r(z)$, the density of rate of events in the network. This quantity may be unknown at the start of the network, but it can be estimated by the central node based on the frequency of receiving messages from the different sensors.

We have assumed that the locations of the sensors are known. This does not require the complicated position finding instruments like GPS devices at each individual sensors. The location is needed relative to the central node, and since the sensors are not mobile, for each sensor it can be set at the installation of the network.

Our other assumption is the knowledge of the residual energy of the nodes as a function of time. This is also a non-restrictive assumption. If the initial energy of a sensor is known to the central node at time $t = t_0$, the residual energy at any time $t_1 > t_0$ can be found by subtracting the spent energy of that sensor in the interval $[t_0\ t_1]$ from the initial energy. Obviously this calculation can be done by the central node by making use of the central knowledge of routes and observing the received messages. To avoid the accumulative errors, each sensor may send its residual energy to the central node once after a while.

REFERENCES

[1] McQuistan, B. R., “Scalar and Vector Fields: A Physical Interpretation,” John Wiley & Sons, 1965.
[2] Jackson , J. D., “Classical Electrodynamics,” Third Edition, John Wiley & Sons, 1999.
[3] Matthews, P.C., “Vector Calculus,” Springer 1998.
[4] Cheng, D. K., “Field and Wave Electromagnetics,” Second Edition, Addison-Wesley, 1989.
[5] Sohrabi, K.; Gao, J.; Ailawadhi, V.; and Pottie, G. J., “Protocols for Self-Organization of a Wireless Sensor Network,” IEEE Personal Communications, Vol. 7, Issue 5, pp. 16-27.
[6] Akyildiz, I. F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E., “A Survey on Sensor Networks ,” IEEE Communications Magazine,” Vol. 40, No. 8, pp. 102-116, August 2002.
[7] Rentala, P.; Musunnuri, R.;Gandham, S. and Saxena U., “Survey on Sensor Networks ,” Available online at: http://www.mdpi.net/sensors.
[8] Estrin, D.; Govindan, R.; Heidemann, J. and Kumar, S., “Next Century Challenges: Scalable Coordination in Sensor Networks ,” Proceedings of 5th annual IEEE/ACM international conference on Mobile Computing and Networking, pp. 263-270, 1999.
[9] Heinzelman, W. R.; Anantha Chandrakasan, A.; and Balakrishnan, H., “Energy-efficient communication protocol for wireless microsensor networks,” In Proceedings of the Hawaii International Conference on System Sciences, Maui, Hawaii, January 2000.
[10] Kulik, J.; Heinzelman, W. R. and Balakrishnan, H., “Negotiation-based protocols for disseminating information in wireless sensor network,” ACM Wireless Networks, vol. 8, no. 2-3, pp. 169-185, 2002.
[11] Ye, F.; Chen, A.; Liu, S. and Zhang, L., “ A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks,” Proceedings of 10th international conference on computer communication and networks, pp. 304-309, 2001.

Appendix

A. Proof of Lemma 1

In order to prove the lemma, it suffices to prove that for every closed contour C we have:

$$\oint_C \vec{D} \cdot \vec{dl} = 0 \tag{26}$$

First we prove the above fact for a contour C that is a rectangle like that in Figure 7. In this figure we define two other equally spaced rectangles inside and outside C, and call those C_{in} and C_{out} respectively. The distance between the edges of C_{in} and C_{out} is assumed to be equal for the four pair of corresponding edges, and we denote it by β. Assume T is the area surrounded between C_{in} and C_{out} . We divide T into four parts: T_1, T_2, T_3, and T_4 as illustrated in Figure 7.

Now we define the following vector field:

$$\vec{\delta}(z) = \begin{cases} \epsilon\hat{i} & if \;\; z \in T_1 \\ \epsilon\hat{j} & if \;\; z \in T_2 \\ -\epsilon\hat{i} & if \;\; z \in T_3 \\ -\epsilon\hat{j} & if \;\; z \in T_4 \\ 0 & otherwise. \end{cases} \tag{27}$$

in which ϵ is a small positive constant. Equation (27) defines a vector field that makes a lossless counterclockwise rotation in T. It can easily be verified that:

$$\vec{\nabla} \cdot \vec{\delta} = 0 \tag{28}$$

Now we observe the fact that if we define $\vec{D_1} = \vec{D^*} + \vec{\delta}$, then $\vec{\nabla} \cdot \vec{D_1} = \rho$, and hence, $\vec{D_1}$ is in the feasible set of the optimization problem defined by equation (12). The

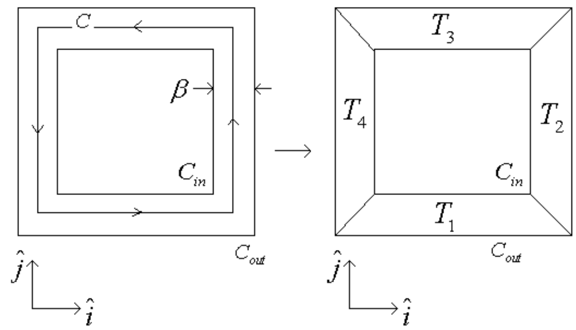

Fig. 7. Illustration of the notations for the proof of Lemma 1.

variation of the cost function defined by equation (10) after adding $\vec{\delta}$ to $\vec{D}^*$ can be written as:

$$\begin{aligned} &\Delta J = J(\vec{D}_1) - J(\vec{D}^*) = \int_A |\vec{D}^* + \vec{\delta} - \vec{D}^*_{av} - \vec{\delta}_{av}|^2\, \mathrm{ds} \\ &- \int_A |\vec{D}^* - \vec{D}^*_{av}|^2\, \mathrm{ds} \end{aligned} \tag{29}$$

Since $\vec{\delta}(z) = 0$ for $z \notin= T$, and $\vec{\delta}_{av} = 0$ we have:

$$\begin{aligned} &\Delta J = \int_T (|\vec{D}^* + \vec{\delta} - \vec{D}^*_{av}|^2 - |\vec{D}^* - \vec{D}^*_{av}|^2)\, \mathrm{ds} \\ &= 2\int_T \vec{D}^* \cdot \vec{\delta}\, \mathrm{ds} + \int_\mathrm{T} |\vec{\delta}|^2\, \mathrm{ds} \end{aligned} \tag{30}$$

If we assume ϵ is small enough, we can ignore the term that has ϵ^2. Then:

$$\Delta J = 2\int_T \vec{D}^* \cdot \vec{\delta}\, \mathrm{ds} \tag{31}$$

On the other hand, for small enough β and ϵ we have:

$$\int_T \vec{D}^* \cdot \vec{\delta}\, \mathrm{ds} = \epsilon\beta \oint_\mathrm{C} \vec{\mathrm{D}}^* \cdot \vec{\mathrm{dl}} \tag{32}$$

From the theory of calculus of variations the value of ΔJ should be zero since $\vec{D}^*$ minimizes the cost function defined by equation (10). Hence:

$$\oint_C \vec{D}^* \cdot \vec{dl} = 0 \tag{33}$$

So far we have proved the validity of equation (34) for a rectangular contour. To complete the proof it suffices to observe the fact that the area surrounded by an arbitrary contour C_1 can be divided into many rectangular elements, and the integral over the boundary of C_1 defined by equation (34) is the sum of integrals over the boundaries of the small rectangles.**QED**

B. Proof of Lemma 2

This lemma can be proved in a similar way as the Lemma 1. By following the same track of proving of Lemma 1, it can be proved that for every closed contour C we have:

$$\oint_C K\vec{D} \cdot \vec{dl} = 0 \tag{34}$$

which means $\oint_C \vec{E}^* \cdot \vec{dl} = 0$. Mathematically, this is equivalent to the fact that $\vec{\nabla} \times \vec{E}^* = 0$. **QED**