Lifetime Bounds, Optimal Node Distributions and Flow Patterns for Wireless Sensor Networks

Enrique J. Duarte-Melo, Mingyan Liu and Archan Misra

Abstract— **In this paper we investigate the expected lifetime and *information capacity* of a data-gathering wireless sensor network. The information capacity is defined as the maximum amount of data (bits) transferred before the first sensor node death due to energy depletion, while ignoring various signaling overheads, such as MAC and routing. We develop a fluid-flow based computational framework that extends existing approaches, which are based on precise knowledge of the layout/deployment of the network, i.e., exact sensor positions. Our method, on the other hand, computes the expected information capacity for a given *distribution* of sensor node layout and sensor data rates, rather than any particular instance of sensor deployment. Our flow model assumes a a continuous density of sensor deployment, and is particularly appropriate for sensor environments characterized by highly dense node deployments. This continuous-space flow model is then discretized into grids and solved using a linear programming approach. Numerical studies show that this model produces very accurate results, compared to averaging over results from random instances of deployment, with significantly less computation. We then use our model to determine optimal node distributions for a linear network and the properties of optimal routing that maximizes the lifetime of the network.**

Index Terms— **Mathematical programming, optimization, system design, wireless sensor networks, lifetime, capacity, sensor deployment, node distribution, optimal routing, fluid flow model,**

I. Introduction

Maximizing the *functional lifetime* of individual sensor nodes is clearly one of the biggest design objectives for any wireless sensor network deployment. This lifetime, and the amount of information that can be collected, depends on (a) the layout of the sensor network, (b) the initial battery capacity on the individual sensor nodes, (c) the characteristics of the sensor data generated at the individual nodes, and (d) the communication costs in transferring such generated data to a set of designated *collector* nodes. In this paper, we present a mathematical framework that accepts each of the above variables as input, and outputs an estimate on the maximum amount of sensory data that can be collected. We develop a linear programming tool that allows us to rapidly compute the lifetime of a sensor network and study the dependence of this lifetime on various parameters of interest. In particular, we use this tool to determine the optimal node distribution for sensor network layouts, and also study the properties of the traffic paths that maximize the network lifetime. We expect that our model will provide sensor network designers a fast computational tool to analyze the aggregate behavior of specific sensor networks of interest, which are often characterized by fairly high node densities.

We consider a wireless sensor network that is deployed over a specific geographic area (the “field”). Nodes of the sensor network are engaged in *sensing and collecting* data from the field, and then transporting it to one or more *collectors* (which may themselves consist of individual or multiple nodes) for further processing. The operations of data sensing and data forwarding may be done continually, periodically or intermittently. Our goal is to determine limits on how long the network can last, and more importantly, how much data the network can collect. In this paper, we concentrate on maximizing the *information capacity* of the network, defined as the maximum amount of information (bits) that can be transferred from the sensing field to the collector regions until the *first* sensor node gets completely drained of its battery and dies, as well as the *lifetime* of the network, defined as the time till this first sensor death.

Our work in this paper is inspired by that of [1], which presents a linear programming approach for computing the lifetime of a specific sensor network deployment. In [1], the data collection and transfer process is represented by a fluid flow model; maximizing the network lifetime is then equivalent to a linear flow maximization problem. The problem formulation in [1] is, however, based on the *precise location* of the individual sensors: should the sensor locations or their data rate be subject to even small changes, the lifetime needs to be recomputed from scratch. Accordingly, this linear programming model cannot handle scenarios where the network topology

E. J. Duarte-Melo and M. Liu are with the Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, {ejd,mingyan}@eecs.umich.edu; A. Misra is with the T. J. Watson Research Center at IBM, NY, {archan@us.ibm.com}.

cannot be deterministically described.

In contrast, our approach aims to determine the network lifetime, based on *probability distributions* of the node densities over the sensing field, and of the distribution of raw sensory data rates over that field. Our focus on describing node layouts and sensor data rates in terms of distributions is motivated by many important sensor networking scenarios, where the specific node deployment is not generated manually, but rather *randomly* according to certain distributions. For example, in many exploration and battlefield scenarios, nodes may simply be dropped remotely (from an airplane or a ship) over a physical area. We cannot control the precise location of the individual nodes but only the aggregate node distribution (at a suitably coarse granularity). In such scenarios, a specific sensor network instance is really a single sample path out of all possible deployment outcomes based on a given distribution.

As in [1], we reduce the lifetime determination problem to a fluid-based flow maximization problem, where the maximization is over routing choices or flow distributions. The key to developing a fluid flow model to handle distributions of nodes is to assume that these distributions are represented by functions that are continuous (over the sensing and collector regions). This approximation is particularly appropriate for sensor network environments that, in contrast to conventional computing networks, exhibit much higher node density. Highly dense sensor fields allow us to treat individual sensor nodes as *fungible* and study the aggregate properties of groups of nodes. The continuous-space fluid flow model is then numerically solved through *discretization*, i.e., by breaking up the continuous field into small, but discrete, individual grids. Interestingly, our numerical results shall show that this approach not only generates very accurate estimates on lifetime and information delivered for dense sensor fields, but remains accurate even for fields with sparsely deployed sensors.

In general, network lifetime and its information capacity are two distinct metrics: maximizing one does not necessarily imply the maximization of the other. However, if the information generation rate is not location-varying, and if each node eventually relays all packets/bits that it receives from other nodes, then we shall show that *maximizing the amount of information transferred is equivalent to simply maximizing the lifetime*. More interestingly, we shall see that maximizing this lifetime results in virtually zero residual energy at all other nodes at the time of death of the first node. In other words, although we have defined the network lifetime to be defined by the death of the first node, maximizing it also maximizes the amount of data that is delivered till *the death of all the sensor nodes*.

As in [1] the notion of *time* in this paper has a rather unconventional meaning, in that we only consider time elapsed when a node is either actively transmitting or receiving. We do not take into account the time a node spends idling. Alternatively, it is *as if* all transmissions and receptions can happen concurrently. In reality, they need to be properly scheduled (e.g, MAC) to avoid collisions, and one has to wait from time to time for its scheduled transmission. Our model focuses only on the *operational lifetime*: by assuming an ideal condition where nodes spend no power in an idle state, and by ignoring any signaling-related overhead, our model provides an upper bound for any practically realizable network. Our formulation also abstracts the communication overhead in terms of "communication energy per bit". While this value may vary with changes to the specific physical layer settings (e.g., the raw bit rate, error correcting codes, etc.), it does not affect the overall applicability of our model.

We shall see that the computation of the bound reduces to a flow maximization problem, where we essentially compute the relative usage of different paths (from a sensor node *generating* data to a collector acting as a *data sink*) for transferring the sensed data. In general, there are an infinite number of routing strategies that can result in the same overall flow rates. Our goal is thus not on determining *a* precise routing strategy, but on studying the relative usage of different paths by *any* well-designed routing strategy. As proved in [1], any such lifetime bound is, however, *always realizable* in practice.

The remainder of the paper is organized as follows. We present the details of our formulation, the solution technique, and a critique of this modeling framework, in Section II. In Section III we present numerical results of our model under various parameter settings. Section IV shows how we use this modeling method to obtain optimal node distribution that maximizes the information capacity in the example of a linear network, as well as properties of the optimal routing strategy. Related work is presented in Section V and Section VI concludes the paper.

II. Problem Formulation

In this section we develop a fluid-flow model for maximizing the lifetime or the total information delivered/transferred by the sensor network and discuss its unique features.

A. A Continuous Model

Suppose we have a sensing field with very densely deployed sensors. At its extreme, the field may be regarded as being *continuously* filled with sensors. Accordingly, let the following continuous functions represent various network parameters as a function of the location (x, y) in the sensing field.

$\rho(x, y)$: The number of sensors per unit space (e.g., m^2) at point (x, y). For example, if N sensors are uniformly deployed over a sensing field of area A, then this density is $\rho(x, y) = N/A$, for all $(x, y) \in A$ by using A to denote both the size and the range of the area.

$i(x, y)$: The information rate density, the amount of information (e.g., number of bits) generated per second per unit space at point (x, y). For example, if every sensor is generating b number of bits/sec, then $i(x, y) = b \cdot \rho(x, y)$.

$e(x, y)$: The initial energy density, or the amount of energy (e.g., joule) present in the beginning per unit space at point (x, y). Suppose in the beginning every sensor carries e joules, then $e(x, y) = e \cdot \rho(x, y)$. We assume that batteries are not rechargeable.

Given these definitions we have the following identities: $\int\int_A \rho(x, y)dxdy = N$, $\int\int_A e(x, y)dxdy = E$, and $\int\int_A i(x, y)dxdy = B$, where N is the total number of sensors in the field, B is the total number of bits generated per second by the field, and E is the total amount of energy available in the beginning. The above definitions can also be generalized to time-dependent parameters.

For simplicity of notation, we will use σ to represent a point in a two-dimensional space, i.e., let $\sigma = (x, y)$, $\sigma^{'} = (x^{'}, y^{'})$, and so on. Our previous definitions can now be written as $\rho(\sigma)$, $i(\sigma)$ and $e(\sigma)$. Note that $d\sigma = dxdy$. Define also the "flows" - $f(\sigma, \sigma^{'})$ to be the amount of data delivered/transmitted from location σ to location $\sigma^{'}$. This value has the unit of "number of bits per unit-source-space per unit-sink-space" or equivalently "number of bits per unit-space-squared". Note that flows are quantities over time, i.e., the total amount from one location to another over a period of time, which is determined by the optimization problem. In general, the data is transported to a collector (or base station), whose location σ_* can be either within or outside the sensing field. Let A denote the area of the sensing field (where the sensors are distributed), and C denote the area of the base station/collectors. Without loss of generality, we can assume that A and C are non-overlapping (as long as a collector/base station is not considered a sensor simultaneously). This distinction becomes trivial when the density functions are replaced by sampling functions at single points, as we will show later.

For our analysis, we do not consider time dependence. Thus, $i(\sigma)$ is only a function of the location σ, but is constant over time. All information is transmitted to the collector. Nodes eventually transmit all data received and do not keep any of the data by the time the network lifetime ends. We then have the following formulation (**P**) for maximizing the total information transferred from A to C:

$$
\begin{aligned}
\max_{f} \quad & t \cdot \int_{\sigma \in A} i(\sigma) d\sigma \quad \sim \quad \max_{f} \ t && (1) \\
\text{S.t.} \quad & \int_{\sigma^{'} \in A} f(\sigma, \sigma^{'}) d\sigma^{'} + \int_{\sigma^{'} \in C} f(\sigma, \sigma^{'}) d\sigma^{'} && \\
= \quad & \int_{\sigma^{'} \in A} f(\sigma^{'}, \sigma) d\sigma^{'} + i(\sigma) \cdot t, \quad \forall \sigma \in A && (2) \\
& \int_{\sigma^{'} \in A} f(\sigma, \sigma^{'}) p_{tx}(\sigma, \sigma^{'}) d\sigma^{'} + && \\
& \int_{\sigma^{'} \in C} f(\sigma, \sigma^{'}) p_{tx}(\sigma, \sigma^{'}) d\sigma^{'} && \\
& + \int_{\sigma^{'} \in A} f(\sigma^{'}, \sigma) p_{rx} d\sigma^{'} + t \cdot \epsilon_s(\sigma, i(\sigma)) && \\
\leq \quad & e(\sigma), \quad \forall \sigma \in A && (3) \\
& f(\sigma, \sigma^{'}) \geq 0, \quad \forall \sigma, \sigma^{'} \in A \cup C && (4) \\
& f(\sigma, \sigma^{'}) = 0, \quad \forall \sigma = \sigma^{'} && (5) \\
& f(\sigma, \sigma^{'}) = 0, \quad \forall \sigma \in C, \forall \sigma^{'} \in A\ . && (6)
\end{aligned}
$$

The equivalency ($\sim$) in (1) is due to the fact that $i(\sigma)$'s are time-invariant and given. The first constraint (2) is a statement of *flow conservation*, i.e., over the lifetime of a sensor, the total amount transmitted must equal the total amount received plus total amount generated/sensed. (3) is the *energy constraint*, i.e., the total energy consumed by a sensor, including transmission, reception, and sensing, cannot exceed the initial energy equipment; (4) is the non-negativity constraint; (5) states that any sensor should not transmit to itself. and (6) means that data does not flow from the collector *back* to the sensors. In a practical scenario there might be broadcasts from the collector to the nodes. However, if we assume that the collector is not energy constrained, then this model merely concentrates on data delivery and remains valid. Here $p_{tx}(\sigma, \sigma^{'})$ is the *energy* dissipation instead of *power* dissipation, in transmitting from location σ to $\sigma^{'}$, in J/bit. p_{rx} is the energy dissipation in receiving. $\epsilon(\cdot)$ is the energy spent in sensing, and $e(\cdot)$ is the initial energy.

The formulation (**P**) is equivalent to an a more generic "max-min" formulation that allows nodes to have arbitrarily different lifetimes t, and that maximizes the minimum of these arbitrary lifetimes.

Some important points of the above model should be noted. Implicitly $i(\sigma) \geq 0$ and $\int_{\sigma \in A} i(\sigma)d\sigma > 0$ are assumed to ensure that the optimization does not become trivial. In (2) the conservation principle is expressed in terms of rate, i.e., in terms of bits per unit space rather than bits. The actual conservation comes by considering the inflow/outflow over the infinitesimal area $d\sigma = dx\ dy$, which gives

$$\begin{aligned} &\left(\int_{\sigma^{'}\in A} f(\sigma,\sigma^{'})d\sigma^{'}\right) d\sigma + \left(\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma^{'}\right) d\sigma \\ = \ &\left(\int_{\sigma^{'}\in A} f(\sigma^{'},\sigma)d\sigma^{'}\right) d\sigma + i(\sigma)d\sigma\cdot t, \end{aligned} \tag{7}$$

where the three integrands can be written in terms of a point $(x,y)\in\sigma$ as $\int_{\sigma^{'}\in A} f(x,y,\sigma^{'})$, $\int_{\sigma^{'}\in C} f(x,y,\sigma^{'})$, and $\int_{\sigma^{'}\in A} f(\sigma^{'},x,y)$, respectively, by using the "intermediate value theorem"[1]. In essence, σ in function $f(\cdot)$ refers to a single point, but the conservation principle refers to the infinitesimal area around that point. Since $d\sigma$ cancels out on all terms in (7), we get (2).

The total amount of information delivered to the collector is $\int_{\sigma\in A}\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma d\sigma^{'}$. Note that, in this model,

$$\int_{\sigma\in A}\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma d\sigma^{'} = \int_{\sigma\in A} i(\sigma)d\sigma\cdot t, \tag{8}$$

by taking one more integral over $\sigma\in A$ on both sides of (2). Thus, the *objective of maximizing lifetime is equivalent to maximizing total amount of data delivered*. As a matter of fact, it seems we can completely eliminate t from the formulation by replacing t in the formulation (**P**) by the equivalent relationship defined by (8). The optimization problem (**P**) can thus be simplified to a maximization on a set of arbitrary non-negative flow variables $f(\cdot,\cdot)$ and is equivalent to the optimization problem (**P1**) shown in Figure 1: For the rest of our discussion we will concentrate on formulation (**P1**) rather than (**P**). Accordingly, we shall focus on directly maximizing the information capacity, rather than the indirect *lifetime* variable.

B. Solution Approach – Discretization

The formulation (**P1**) in Section II is in itself intractable, since it is an infinite-dimensional optimization problem due to the continuous and integral nature of its elements. An immediate thought is to solve the discretized version of this formulation. This corresponds to dividing the sensing field into grids of equal or variable sizes. This inevitably introduces error. However, if we consider a highly densely populated sensing field, then with relatively high resolution grids, we expected the partitioning or discretization to produce reasonably accurate results.

$$\begin{aligned}
\max_{f} \quad & \int_{\sigma\in A}\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma d\sigma^{'} & (9)\\
\text{S.t.} \quad & \int_{\sigma^{'}\in A} f(\sigma,\sigma^{'})d\sigma^{'} + \int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma^{'} & \\
= \quad & \int_{\sigma^{'}\in A} f(\sigma^{'},\sigma)d\sigma^{'} + & \\
& i(\sigma)\frac{\int_{\sigma\in A}\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma d\sigma^{'}}{\int_{\sigma\in A} i(\sigma)d\sigma}, & \\
& \forall\sigma\in A & (10)\\
& \int_{\sigma^{'}\in A} f(\sigma,\sigma^{'})p_{tx}(\sigma,\sigma^{'})d\sigma^{'} + & \\
& \int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})p_{tx}(\sigma,\sigma^{'})d\sigma^{'} + & \\
& \int_{\sigma^{'}\in A} f(\sigma^{'},\sigma)p_{rx}d\sigma^{'} + & \\
& \{\frac{\int_{\sigma\in A}\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma d\sigma^{'}}{\int_{\sigma\in A} i(\sigma)d\sigma}\}\cdot\epsilon_s(\sigma,i(\sigma)) & \\
\leq \quad & e(\sigma), \quad \forall\sigma\in A & (11)\\
& f(\sigma,\sigma^{'})\geq 0, \quad \forall\sigma,\sigma^{'}\in A\cup C & (12)\\
& f(\sigma,\sigma^{'}) = 0, \quad \forall\sigma=\sigma^{'} & (13)\\
& f(\sigma,\sigma^{'}) = 0, \quad \forall\sigma\in C,\forall\sigma^{'}\in A & (14)
\end{aligned}$$

Fig. 1. Formulation P1

Following this thought, suppose the sensing field is partitioned into M smaller non-overlapping areas/zones, indexed by $m, m=1,2,\cdots,M$. each of size A_m. That is $A_m\cap A_n=\phi$ for $m\neq n$, and $A_1\cup\cdots A_M = A$. Again we will abuse the notation and let A_m indicate both the size and area itself. Then the original objective function (9) becomes

$$\begin{aligned}
\max\nolimits_f \quad & \int_{\sigma\in A}\int_{\sigma^{'}\in C} f(\sigma,\sigma^{'})d\sigma d\sigma^{'} = \int_{\sigma\in A} f(\sigma,\sigma_C)Cd\sigma \\
= \quad & \sum_{m=0}^{M}\int_{\sigma\in A_m} f(\sigma,\sigma_C)Cd\sigma = \sum_{m=0}^{M} f(\sigma_m,\sigma_C)A_mC,
\end{aligned}$$

where σ_m is some location within areas A_m, and σ_C is some location within C. The first and third equalities are due to the theorem of intermediate value since $f()$ is a continuous function over the two-dimensional space.

The two constraints can be discretized in a similar way. For example, we obtain the following discretized version of the flow conservation constraint and energy constraint:

[1] It is while applying the intermediate value theorem that we require the functions $i(\sigma)$ and $e(\sigma)$ to be continuous in constraints (2) and (3) respectively.

$$\begin{aligned} & \sum_{k\in M} f(\sigma_m^1, \sigma_k^1) A_k + f(\sigma_m^2, \sigma_C(m)) C \\ = & \sum_{k\in M} f(\sigma_k^2, \sigma_m^3) A_k + i(\sigma_m^4) t \quad \forall\, m \in M \end{aligned} \tag{15}$$

and

$$\begin{aligned} & \sum_{k\in M, k\neq m} f(m,k) p_{tx}(m,k) A_k + f(m,C) p_{tx}(m,C) C + \\ & \sum_{k\in M, k\neq m} f(k,m) p_{rx} A_k + t\cdot \epsilon_s(m, i(m)) \\ \leq\ & e(m)\cdot A_m \quad \forall\, m \in M, \end{aligned} \tag{16}$$

where $e(m)$ is the approximated nominal energy density within area m, and $\sigma_m^i, i = 1, \cdots 4$ are points within area A_m.

This discretization has the effect of creating a regular or possible irregular grid/partition in the field. Every region A_m is a rectangle/sector in the grid/partition where the information and energy density of the region A_m are concentrated on a single point within the rectangle/sector. In essence, this method is using a grid version of the network, where nodes are deployed at points in the grid, to estimate the average information capacity of a network with certain node distribution pattern. What we really want is the expected information capacity over the distribution of sensor node deployment, denoted by $E[C(\cdot)]$, where $C(\cdot)$ is the capacity of the network implicitly defined through the fluid flow model. This expected value can currently only be obtained via averaging over a large number of random cases. The continuous fluid flow model followed by discretization presented in this paper essentially computes the information capacity of the grid deployment, which may be thought of as the "expected deployment" with the right partitioning, imprecisely denoted by $C(E[\cdot])$. Thus whether our model is valid hinges upon whether $C(E[\cdot])$ is a good approximation for $E[C(\cdot)]$. Note that $C(E[\cdot]) = E[C(\cdot)]$ is trivially true if $C(\cdot)$ is a linear function of its arguments. However, here the information capacity is only implicitly defined via a math program.

On the other hand, the grid network can be viewed as a "perturbed" version of any random layout, and vice versa. Using results from the stability of linear programs (LP), see for example [2], one can show that the LP defined through the continuous model followed by discretization is indeed stable. This means that when the node locations are perturbed within a certain range, it produces bounded difference on the objective function value. The key then is to show that the disturbance introduced by approximating the averages of random layouts using the grid/partitioned version is within this stability range. While we believe this to be true, especially when node density is high, more analysis is needed to confirm this. However, it is out of the scope of this paper, where we shall use numerical studies to empirically validate our method. Note also that the stability of the LP does not imply that the information capacity computed from a random "sample path" is close to that of another, nor does it imply that the information capacity computed by a random sample path is always close to the expected capacity $E(C(\cdot))$–indeed, our model is useful precisely because (as we shall see in Section III.A), the information capacity of a particular deployment may differ significantly from the mean value.

C. Discussion and Critique of the Model

This formulation is really a generalization of the formulation in [1]. To see this, note that if we know the precise location of each sensor, then the continuous functions become impulse ($\delta()$) functions sampling at these particular locations. Consequently the integration operations reduce to summation at those sampled locations, which becomes the same as the formulation in [1]. Due to this generality, our formulation produces a much more powerful result in that it represents the average capacity of the network for a *distribution* of the deployment (expressed by the density function) rather than for a particular deployment.

Our models assume that $f(\cdot,\cdot)$, $p_{tx}(\cdot,\cdot)$, etc. are continuous functions of position (x, y) (to apply the intermediate value theorem). Hence, applying them to real-life scenarios, where parameters may exhibit discontinuity, will inevitably introduce some errors. For example, most real communication adapters have a few discrete levels for p_{tx}, which is then a step-wise constant function of the (continuous-valued) distance to the receiver. In addition, our solution approach inevitably involves coarse or fine-grained approximation. The utility of this method thus depends on the relative size of this approximation error. In the next section we will present numerical experiments to quantify some of these errors.

By modeling sensor deployment as a continuous function, (**P1**) provides a way to obtain the capacity for different node density functions (deployment patterns), and thus study the impact of different node distributions on the maximum achievable capacity. For example, as we show in Section IV this analytical tool can be used to study non-linear deployment of sensors, and the degree to which we need to deploy a larger number of sensors close to the collector node to counteract the higher energy that these nodes expend in relaying information from sensors further away.

III. Numerical Experiments

The main purpose of our numerical experiments is to examine whether our model can provide accurate results, and whether such results are sensitive to changes in a range of parameters as well as the granularity of the discretization. Therefore, almost all results presented in this section are in the form of comparing the result of our fluid flow formulation (**P1**) using our computation method, with that obtained by averaging over 100 instances of the sample path fluid flow models each for one random deployment. These results are compared under equivalent conditions. Specifically, the number of nodes in the sensing field, the location of the collector, as well as the size of the field will be kept the same for each pair of comparison. The total energy will be kept at $E = 1$ joule.

We adopt the following energy model. Total energy consumed by a sensor in transmission is $E_t(r) = (e_t + e_d r^{\alpha})b$, where e_t and e_d are specifications of the transceivers used by the nodes, r is the transmission distance, b is the number of bits sent, and α depends on the characteristics of the channel and will assumed to be time invariant. Energy consumed in receiving is $E_r = e_r b$. Finally $E_s = e_s b$ is the energy spent in sensing/processing data that is quantized and encoded into b bits. Again e_r depends on the transceivers. In this section we will use the following parameter values taken from [1]: $e_t = 45\text{x}10^{-9}$, $e_r = 135\text{x}10^{-9}$, and $e_s = 50\text{x}10^{-9}$, all in J/bit, and $e_d = 10\text{x}10^{-12}$ in J/bit-meter$^{\alpha}$. α ranges between 2 and 4. As stated earlier, we will ignore idling energy in this model. Since time elapses in the same way for every sensor node, by the time the first node dies every sensor will have experienced equal amounts of idle time.

A. Varying Grid Size

Suppose 225 nodes are uniformly distributed over a square field of size 1000×1000 square-meters (lower left corner at $(0, 0)$). The collector is located at (500,-1000). The results obtained by averaging over 100 random deployments (AVG) are shown in Table I. We included the 95% confidence interval (C.I.) as well as the maximum and minimum values among these random samples, all in bits. From these results, we see that the information capacity of a particular instance of node deployment can be almost 10% lower or higher than the mean value.

α	AVG	95% C.I.	Min	Max
2	46615	[46292 , 46938]	43593	49577

TABLE I
Average information capacity from 100 random deployments.

To solve (**P1**), grids of equal-sized squares are employed. For the rest of our discussion we will use the term "number of grids" to indicate the number of such squares in a grid partition. As the number of grids increases, the computation is done on an increasingly finer granularity. In each grid/square, energy and information densities are concentrated on a single point at the center of the grid/square. Table II lists selected results from this experiment by varying the size/number of grids, and Figure 2 shows the complete results, both for $\alpha = 2$.

# of Grids	P1	% error
225	46885	0.58%
196	46884	0.58%
144	46873	0.55%
100	46843	0.49%
64	46840	0.48%
36	46801	0.40%
16	46623	0.02%
9	46384	-0.49%
4	45872	-1.6%

TABLE II
Varying number of grids ($\alpha = 2$).

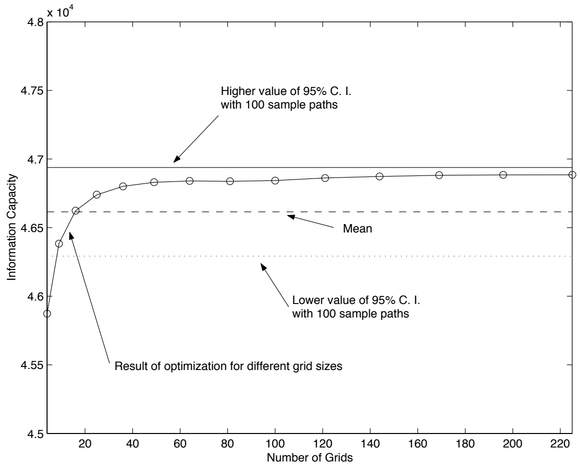

Fig. 2. Varying number of grids ($\alpha = 2$).

The above results first of all showed very good accuracy of our model, with almost all results within the 95% confidence interval. Secondly, we see that the coarser grained computation (with fewer number of squares/grids) generates equally accurate results as the finer grained computation. This suggests that we could obtain sufficiently accurate results with very few number of grids (as few as 9). As the number of grids increases, the estimate given by our model also increases[2]. This is because as the number of grids increases, each grid has more options as to how to send its data to the collector and therefore the capacity is higher. There seems to be a certain level of granularity, corresponding to a certain number of squares/grids in the partition, that comes closest to the mean. In the case of Table II, this number is 16. Further study is needed to see whether this number can be pre-determined.

All results are obtained in Matlab. Results in Table II are obtained in a matter of seconds or minutes (the finer the grid the longer it takes to solve the optimization problem). On the other hand, to obtain the results in Table I we need hours of computation. Thus our model can indeed serve as a very powerful computational tool.

B. Insensitivity to Varying Parameters

In this subsection we examine the robustness of our model by varying a range of parameters. We first vary the size of the field by considering 10×10, 100×100 and 1000×1000, as shown in Table III. α is set at 2, the number of nodes and square grids in the partition is 225 for all cases, and the total energy of the network is again held constant. Note that for a smaller field this implies a larger information capacity since the average transmission range is also smaller. We see that in all cases the result of our optimization model closely approximates that obtained by averaging over random deployments.

Field size	P1	AVG	% error
10^2	10137000	10138000	-0.01%
100^2	2342900	2322800	0.86%
1000^2	46885	46615	0.58%

TABLE III
VARYING FIELD SIZE.

Next we examine the effect of varying the attenuation factor α. Table IV shows the results for different values of α. Again we see that our model gives very good estimates. Note that the information capacity for the cases $\alpha = 3.5$ and $\alpha = 4$ falls below 1 bit. This is because (1) the linear program generates results assuming bits are infinitely divisible; and (2) the parameters we have used are such that very little information can be transferred under these conditions. With proper scaling, these values can be made more realistic.

α	P1	AVG	% error
2	46885	46615	0.58%
2.5	1282.9	1274.3	0.67%
3	34.8598	34.8813	-0.06%
3.5	0.9544	0.9464	0.84%
4	0.0263	0.0258	1.94%

TABLE IV
VARYING α.

C. Varying the Number of Sensor Nodes

In this subsection we examine the effect of varying the number of sensor nodes in the field, while keeping the total energy constant. We will keep the size of the sensing field to 1000×1000 and $\alpha = 2$. The number of sensor nodes in the field is varied from 4 to 225. Note that such a change does not affect the results from our model (**P1**), since (**P1**) only relies on the node distribution, energy distribution, and the granularity of the partition. Having different number of physical nodes results in the same discretized version of (**P1**) in this case. The comparison results are shown in Figure 3. In each case the result is compared to the result of (**P1**) using the same number of squares in the grid as the number of nodes.

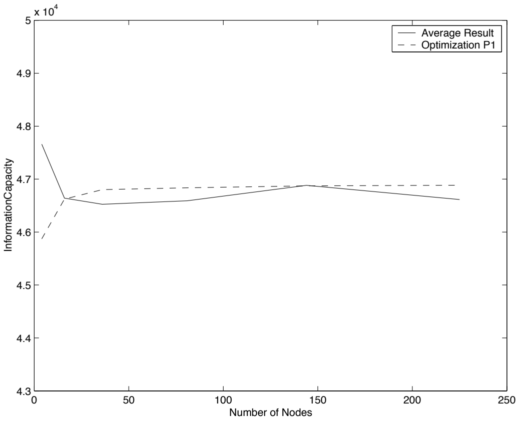

Fig. 3. Effect of varying the number of sensor nodes

From these results we see that a change in the number of nodes does not affect our results as long as the number of nodes is not too small (below 16 in this case). When the number of nodes is very small, e.g., 4, the error between our method and averaging over random deployments increases. We believe this has to do with the stability condition of the corresponding LP. As the number of nodes decreases, the "distance" between any two randomly generated deployments also increases. Thus the LP based on the regular grid deployment represents too large a perturbation.

[2] We did not evaluate partitions with more than 225 grids since we only have 225 nodes in the field.

D. Non-uniform Node Distributions

All previous results employ a uniform node distribution across the sensing field. In this subsection we will examine different node distributions while fixing the total number of nodes at 225. Specifically we will consider the linear sloped node distribution shown in Figure 4, where the node density linearly increases over the field as it gets closer to the collector. To obtain a sequence of node distributions, we vary the slope by changing the length of the line segments $\overline{AB}$ and $\overline{CD}$ shown in Figure 4. The uniform distribution is a special case with $\overline{AB} = \overline{CD} = 1 \cdot c$, where c is a normalizing constant. For simplicity we will assume that the per-node energy and information generation rate remain constant. Accordingly, the energy density and information generation functions are non-uniform (linearly scaled versions of the node distribution) as well.

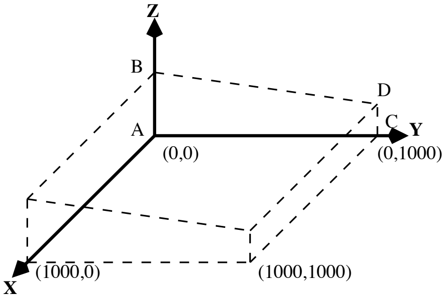

Fig. 4. “Sloped” distribution

Under such non-uniform distributions, the discretization of (**P1**) is done by partitioning the field into un-equal sized rectangles while keeping the total energy in each rectangle grid constant. The results are shown in Table V by using 225 number of such rectangle grids. Again our model produces very accurate estimates. It should be noted that the increasing capacity with the increase in the slope is to be expected since in this case more bits are generated at locations closer to the collector. A more interesting scenario to be examined in the next section is to keep the information density uniform, while node distribution is kept non-uniform, i.e., using some nodes purely as relays.

$(\overline{AB}, \overline{CD})$	P1	AVG	% error
$(2c, 0)$	57162	57322	-0.28%
$(1.75c, 0.25c)$	54602	54769	-0.3%
$(1.5c, 0.5c)$	52013	52215	-0.38%
$(1.25c, 0.75c)$	49431	49424	0.014%
$(1c, 1c)$	46885	46615	0.58%

TABLE V

VARYING NODE DISTRIBUTION.

E. Residual Energy

Figure 5 shows the amount of residual energy of every node in the two-dimensional network we have being using, with 225 nodes in a 1000 × 1000 network, using (**P1**). Note that (**P1**) only seeks to maximize the amount of data delivered till the first sensor node dies, and equivalently, the time till the first death. It does not seek to maximize the time to the death of the last sensor in the network. However, as we can see from this result, the maximal is actually achieved by balancing the lifetime of each individual sensor. As a result, when the first sensor dies, there is virtually no energy left in any of the other sensors, either. Therefore, in effect (**P1**) maximizes the total amount of data delivered till all sensors die. Note that the residual energy values are not exactly zero due to numerical tolerance used when solving the problem in Matlab.

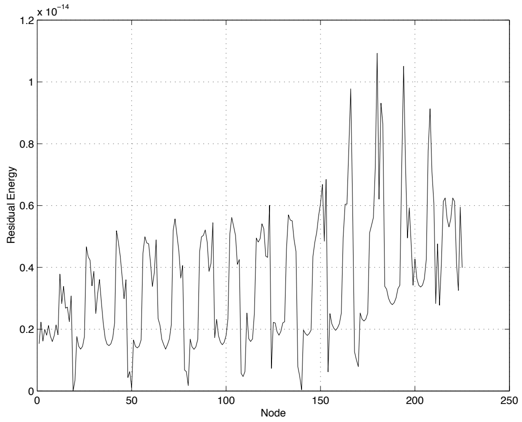

Fig. 5. Residual node energy

IV. Optimal Node Distributions and Flow Patterns

In this section we use (**P1**) to investigate the effect node distribution has on the network lifetime and capacity. We will also examine the optimal flow allocation patterns to derive insights into the properties required of a well-designed routing protocol. We emphasize that our model provides a rapid computational tool that facilitates the investigation of alternative node distributions. In this section we will solely focus on a “linear” network where sensor nodes as well as the collector are lined up on a straight line, although our method can be equally applied to two-dimensional as well as three-dimensional networks. This is because working in one dimension makes the result much easier to represent and interpret.

Our network consists of a line segment of length D between $(0, 0)$ and $(D, 0)$, on which sensor nodes are distributed. Our computation will be based on dividing the line segment into a constant number (M) of grids. The collector is located at $(D+\frac{D}{M},0)$, a distance D/M away from one end of the line segment.

A. Optimal Node Distribution

One question of obvious interest is what would be an optimal node distribution that could maximize the network lifetime as well as total amount of data deliverable. While most reported work on sensor networks assumes a uniform node distribution, it is intuitively clear that the network should last longer if we place more nodes closer to the collector, the point of traffic concentration. Following this, even though there are infinitely many types of possible distributions, we will only consider the following family of exponential node distributions:

$$f_X(x) = cx^a, \quad a \geq 0 \ , \tag{17}$$

where x is the location on the line segment, c is a normalizing constant, and a will be varied to obtain different distributions. Note that $a = 0$ and $c = 1/D$ gives the uniform distribution. As a increases, more nodes are being deployed closer to the collector. For simplicity, below the term "optimal distribution" will refer to the best distribution among this family of distributions.

For each a value, we create $M = 100$ grids and set $D = 1000$. We will assume that the field has a constant information distribution across the network. Therefore if we place more nodes closer to the collector, then nodes closer to the collector will generate less data while nodes far away will generate more. Each node will also contain equal amount of initial energy. *The size of a grid is chosen such that each grid contains an equal amount of probability mass of the node distribution.* In other words, each grid contains an equal number of nodes and possesses equal initial energy.

Figure 6 shows the amount of data delivered by the network for different values of a. Here we have used two different energy models for comparison purposes. "Energy Model A" refers to the model presented and used in the previous section, while "Energy Model B" is the same model with a different set of parameters taken from [3]: $e_t = 50\text{x}10^{-9}$, $e_r = 50\text{x}10^{-9}$ both in J/bit, and $e_d = 100\text{x}10^{-12}$ in J/bit-m$^{\alpha}$. In this section α is set to 2. We see that the maximum amount of data is delivered when a is approximately 0.7677 in the case of model A and approximately 0.8 in the case of model B. $a = 0$ corresponds to a uniform distribution and $a = 1$ corresponds to a linear density. With a uniform distribution, nodes closer to the collector spend too much energy relaying data for other nodes; for a very high value of a, nodes far away spend too much energy because they are generating the majority of the data. The maximizing node distribution essentially achieves the optimal balance between these two effects.

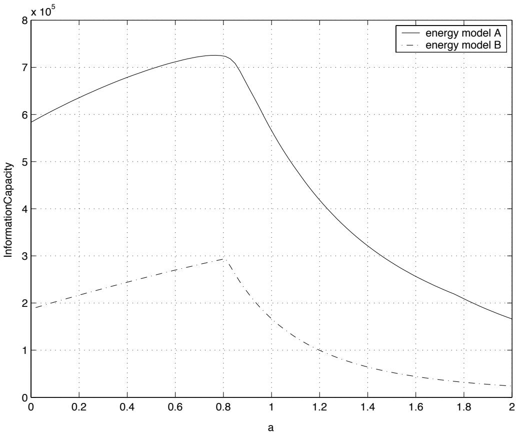

Fig. 6. Varying node distributions.

It's interesting that except for the scale, the two models generate strikingly similar results. They show that the the uniform distribution can result in a capacity loss of about 18% and 35% from the optimal value for the two models, respectively. We also see that, once a exceeds 1, causing most nodes to be deployed too close to the collector, the results deteriorate quickly.

B. Residual Energy

Figure 7 shows the residual node energy in the linear network scenario under two types of node distributions using energy model A: (a) the uniform distribution, and (b) the optimal distribution (exponential with $a = 0.7677$). Like earlier results in Figure 5 for the two-dimensional layout, we see that there is virtually no energy left in the network by the time the first sensor nodes dies, i.e., all nodes in effect die at the same time. Again, the residual energy values are not exactly zero (or even fall below zero) due to numerical tolerance used in Matlab.

C. Optimal Flow Patterns and Routing Implications

Our linear programming framework essentially computes the optimal flows between any two points σ and $\sigma^{'}$. By studying the optimal flow allocation patterns, we can obtain useful insights into the characteristics that an optimal (or close to optimal) routing strategy should possess. For example, if it turns out that most of the flow mass is concentrated over small-distance links, it follows that the routing algorithm should prefer a larger number of small-distance hops over a small number of larger-distance hops. Since the routing pattern obtained

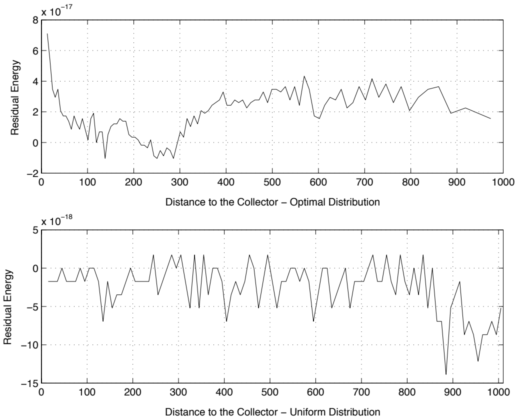

Fig. 7. Residual node energy

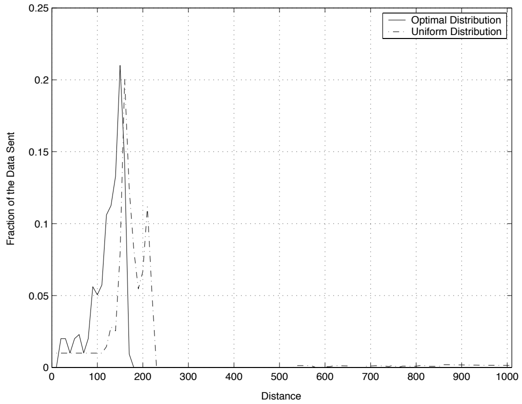

Fig. 8. Fraction of data vs. distance.

from our computation is specific to the particular grid used, we focus on *extracting* the essential principles of the optimal traffic paths.

We consider two measures of the optimal flow patterns. The first is the fraction of data transmitted over a given distance (as a function of the distance), i.e., a histogram of the flow mass vs. the link distances. The second is the average transmission distance of a node (averaged over the total amount of data transmitted by that node), as a function of its distance to the collector. Both these measures attempt to reveal the (average) size of a transmission hop. The first measure also shows whether all flows use hops of similar distance, or whether the flow mass is distributed over a wide range of hop distances. The second one shows how the length of the average hop changes depending on the location of the nodes.

We will determine these two characteristics with the optimal node distribution ($a = 0.7677$) and the uniform distribution ($a = 0$), both under energy model A. Figure 8 shows the fraction of data transmitted over a given distance for both cases. For a given distance d, this fraction is defined as $\frac{1}{M}\sum_{i=1}^{M}\frac{\text{flow}_i^d}{\text{flow}_i^T}$, where flow_i^d is the total flow out of node i that is transmitted over a distance d and flow_i^T is the total flow out of node i. In the uniform distribution case, where the transmission distance has discrete values (multiples of the distance of a single hop), the curve is generated by connecting values at these discrete points. In the non-uniform distribution case, where each node has a different set of possible transmission distances, we combine data transmitted over 10 meter segments, and represent it as a single point. For example, the value at 300 meters represents all transmissions over 290 to 300 meters.

From Figure 8 we see that there is a clear “distance threshold” beyond which data rarely travels in one hop. This threshold seems to be around 180 meters in the optimal case, and around 220 meters in the uniform case. This result indicates that a node’s transmission range can be limited to a certain level and not have a big effect on how much data can be delivered. In addition, the majority of data is transmitted over distances between 100 and 180 meters in the optimal case, and between 140 and 220 meters in the uniform case. Note that nodes closer to the collector do not have the option of transmitting over large distances. So in that sense this result is somewhat biased towards smaller distances.

Perhaps most importantly, this figure shows that a significant portion of the flow mass is transported over medium-sized hops (in the uniform case, the distance between neighboring grids is 10 meters), and not directly to the nearest neighbor. Various proposals for minimum-energy routing in multi-hop wireless networks (e.g., [4]), on the other hand, prefer a large number of small distance hops to minimize the communication energy. Such protocols are often designed for ad-hoc networks, where all nodes are equally likely to be both sources and sinks of data. In contrast, for sensor network environments where most sensor nodes are either sources or relays, it appears that using direct transmissions to more distant neighbors is preferable, since it reduces the forwarding burden on intermediate nodes. We also see that the optimal node distribution has a smaller range than the uniform case. For the optimal node distribution, nodes further away from the collector have more data to transmit (the information generation rate is constant over the entire field) than in the uniform distribution, and accordingly prefer smaller distance hops to reduce their transmission costs.

Figure 9 shows the average distance a node transmits its data as a function of its distance from the collector. The node closest to the collector will always send directly to it and therefore its average distance is always

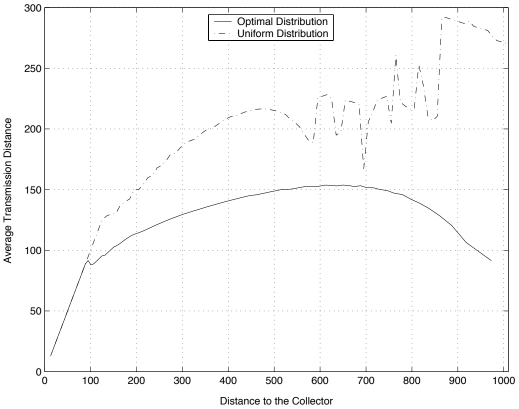

Fig. 9. Average transmission distance.

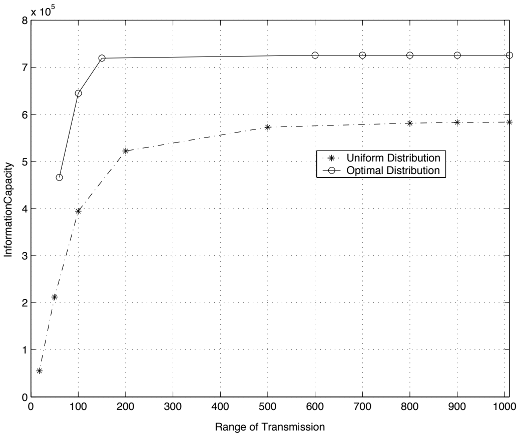

Fig. 10. Data delivered vs. limit of transmission range

its distance to the collector. As nodes are placed further away, we see that their average distance continuous to increase, although not as fast as their distance to the collector. This shows an increased reliance on having their data relayed by other downstream nodes. Finally, we see that for nodes far away from the collector, their average distance decreases under the optimal distribution but continues to increase under the uniform distribtion. This shows that under the optimal distribution such nodes transmit most of their data to increasingly closer relaying nodes. Since regions farther from the collector have a smaller node density, the corresponding nodes have higher data generation rates. Accordingly, they conserve their transmission energy by choosing shorter hops, which is less of a need under the uniform distribution. Note also that the curve for the optimal case is near perfectly smooth: small changes in node location only create small changes in the average transmission distance. *This suggests that near-optimal routing might be constructed using location information.*

D. Limited Transmission Range

In the previous subsections, we have allowed the maximum transmission range of each node to be unbounded. We now consider the impact of specifying a maximum limit on the transmission range. Since the previous studies show that the bulk of the data flow occurs over intermediate hop distance, we expect that limiting this transmission range to moderate values should not significantly reduce the information capacity.

To consider a bounded transmission range, we add the following extra constraint to (**P1**): $f(\sigma, \sigma') \cdot \left[d(\sigma, \sigma') - r\right]^{+} = 0$ for all σ, σ', where $[x]^{+}$ takes value x or 0, whichever is greater. $d(\sigma, \sigma')$ denotes the distance between one location/grid σ and the other σ', and r denotes the maximum range of transmission allowed.

Figure 10 shows the total amount of data that could be delivered to the collector for different maximal transmission ranges. We see in the case of optimal node distribution, relaxing the transmission range constraint to be beyond 150 meters (the “knee”) makes virtually no difference. In the uniform case, the total amount of data does continuously increase as the transmission range limit is relaxed. However, this increase significantly slows down beyond 200 meters (the “knee”). In each case, an increase in the transmission range limit results in significant gain in information capacity when the range is below the “knee”. For the optimal distribution, the smallest transmission range shown is 60 meters–any lower value causes the nodes farthest from the collector to become disconnected. Similarly, for the uniform distribution, the smallest range shown (18 meters to avoid division by zero in Matlab) is the minimum needed to maintain connectivity. These results indicate that sensor nodes should be equipped with moderately powerful radio interfaces, capable of direct transmission over reasonably large distances, to achieve close-to-optimal information capacity.

V. Related Work

Several different approaches have been used to measure or quantify the lifetime of sensor network deployments. As already stated, the fluid-model modeling technique used in [1], [5] is closest to our approach–in contrast to our emphasis on distributions over a sensing field, [1], [5] considers lifetime bounds for a specific instance of sensor network deployment. [1] Further determined lifetime bounds in the presence of a) traffic aggregation (where intermediate nodes would compress the incoming data), and b) multiple locations (where a sensor node could be located at multiple discrete points with different probabilities). In a related problem, a similar linear program is used in [6] to determine how routing should be done in order to increase network lifetime.

The power consumption of specific sensor network technologies and deployments has also been studied in [7], [3]. However, these studies are not concerned with the computation of theoretical bounds. Instead, they focus on novel algorithms and protocols for reducing the routing-related energy overhead in sensor networks. For example, [3] proposed LEACH, a clustering protocol that uses data aggregation over a hierarchical topology to reduce the power consumed by individual sensor nodes. Alternatively, the lifetime of specific sensor network topologies has also been studied using hybrid automata modeling [8]. In [8], a model-based simulator is used to determine the variation in network lifetime with changing distances between the sensing nodes and the collector node.

VI. Conclusion

The paper presented a modeling methodology that drastically reduces the time needed for determining the expected information capacity of a data-gathering wireless sensor network. With this framework we are able to derive the expected lifetime and information capacity of any *distribution* of sensor nodes rather than just particular sample paths of the node deployment. We conducted various numerical experiments, under a variety of parameters. We showed that results generated under this formulation are quite insensitive to the change in a range of parameters, including field size, grid size and the attenuation parameter α. We then used this method to determine for a linear network the optimal node distribution within a family of distributions that can maximize the network lifetime and maximize the total amount of data delivered by the network. Our numerical studies show that, to maximize the network lifetime, sensor nodes must transmit a significant fraction of their packets directly to longer-distance neighbors (using medium-range radios), rather than simply forwarding it to their nearest downstream neighbor. In future work, we plan to use our linear programming tool to further study how various parameters of dense sensor networks impact its operational lifetime.

References

[1] M. Bhardwaj and A. P. Chandrakasan, "Bounding the lifetime of sensor networks via optimal role assignments," in *IEEE INFOCOM*, 2002.

[2] K. G. Murty, *Linear Programming*, John Wiley and Sons, 1983.

[3] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy efficient communications protocols for wireless microsensor networks," in *Hawaii International Conference on System Sciences (HICSS '00)*, January 2000.

[4] J. Gomez and A. Campbell, "Power-aware routing optimization for wireless ad hoc networks," in *High Speed Networks Workshop (HSN)*, June 2001.

[5] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan, "Upper bounds on the lifetime of sensor networks," in *IEEE ICC*, 2001.

[6] J. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *IEEE INFOCOM*, 2000.

[7] S. Lindsey and C. Raghavendra, "Pegasis: Power efficient gathering in sensor information systems," in *IEEE Aerospace Conference*, March 2002.

[8] S. Coleri, M. Ergen, and T. Koo, "Lifetime analysis of a sensor network with hybrid automata modeling," in *1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, September 2002.