Bounding the Lifetime of Sensor Networks Via Optimal Role Assignments

Manish Bhardwaj, Anantha P. Chandrakasan

Abstract—**A key challenge in ad-hoc, data-gathering wireless sensor networks is achieving a lifetime of several years using nodes that carry merely hundreds of joules of stored energy. In this paper, we explore the fundamental limits of energy-efficient collaborative data-gathering by deriving upper bounds on the lifetime of increasingly sophisticated sensor networks.**

I. INTRODUCTION

Wireless networks composed of thousands of highly integrated sensor nodes hold the promise of sensing that is far superior, in terms of quality, robustness, cost and autonomous operation, to that offered by using a few, ultra high precision macro-sensors [1]. Such sensor networks are expected to find widespread use in a variety of applications including remote climate monitoring and seismic, acoustic, medical and intelligence data-gathering. Due to their compact form factors, wireless sensor nodes are severely energy constrained. Furthermore, replacing batteries on up to thousands of nodes in possibly harsh terrain is infeasible. Hence, it is well accepted that the key challenge in unlocking the potential of such networks is maximizing their post-deployment active lifetime.

Effort aimed at increasing the lifetime of sensor networks is two pronged. First, the node and the physical link must be made as energy efficient as possible. See [2], [3], [4] for some representative work. Second, the *collaborative strategy* i.e. the strategy that governs how nodes cooperate to perform the sensing operation, must be energy efficient as well. Work in this area has dealt with different aspects of the problem. Work reported in [5] highlighted the need for metrics other than those used in traditional networks when energy is an issue. Various energy-aware routing heuristics were also proposed in this paper. The first demonstration of near-optimal maximum lifetime routing in ad-hoc networks was [6], [7]. Minimum-energy, but infinite lifetime ad-hoc networks were the subject of [8]. In [9], [10] energy-aware heuristics were used to guide protocol design in networks that support certain types of collaboration.

In this paper, our main objective is not to propose a new collaborative protocol that leads to greater network lifetime. Rather, it is to bound the network lifetime that *any* collaborative protocol can ever hope to achieve. In previous work we computed such bounds for basic data gathering scenarios using simple, non-constructive proof techniques [11]. While this approach results in easy-to-use, closed form expressions for lifetime bounds, it does not factor in network topology and does not accommodate aggregation of data streams. In this paper, we propose a new approach which, in principle, permits derivation of bounds for networks with arbitrarily complex capabilities, although the computational costs of such derivations may be prohibitive. We then show that for several practically useful scenarios, including sensor networks with a specified topology that allow aggregation, this approach in fact leads to polynomial time bound derivation.

In the next section we discuss the operation of sensor networks in greater detail, define lifetime and discuss node energy models. We introduce the role assignment framework in section 3 and use it to derive bounds for a variety of data gathering scenarios. This is followed by some illustrations of the new technique. We end with a summary.

II. PRELIMINARIES

A. Basic operation

The goal of a sensor network is to gather information from a specified region of observation ($\mathcal{R}$) and relay it to an energy-unconstrained *basestation* (B) (figure 1). This information originates due to one or more *sources* located in $\mathcal{R}$. At any given instant, nodes in a sensor network can be classified as *live* or *dead* depending on whether they have any energy left or not. By assuming different roles, live nodes collaborate to ensure that whenever a source resides in $\mathcal{R}$, it is sensed using a minimum specified number of sensors (k) and the resultant data relayed to B. In the collaborative model we assume, a role is composed of one or more of the following *sub-roles*:

• Sensor: The node observes the source via a sensor, digitizes this information, post-processes it and produces data

The authors are with the Department of EECS, Massachusetts Institute of Technology (MIT), Cambridge, MA 02139. M. Bhardwaj is supported by an IBM Fellowship. This research is sponsored by the Defense Advanced Research Projects Agency (DARPA) Power Aware Computing/Communication Program and the Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-00-2-0551. E-mail: {manishb, anantha}@mit.edu.

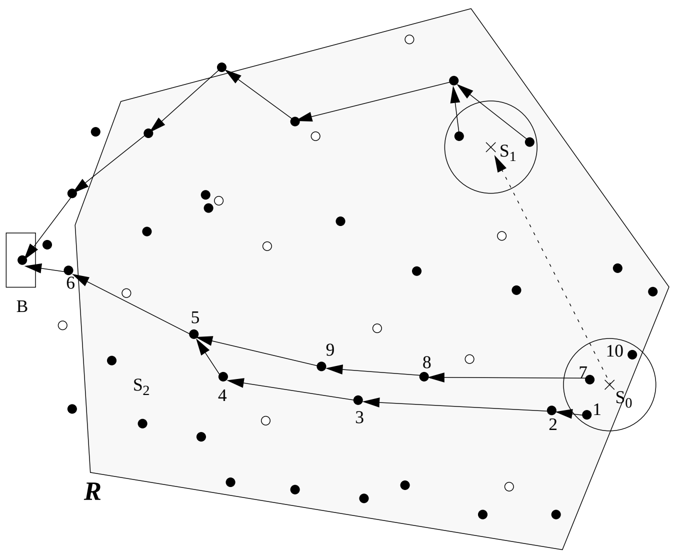

Fig. 1. A sensor network gathering data from a circularly observable source (denoted by a $\times$) residing in the shaded region R. Live nodes are denoted by $\bullet$ and dead ones by $\circ$. The basestation is marked B. In this example we require that at-least two nodes sense the source. When the source is at S_0, nodes 1 and 7 assume the role of sensors and nodes $2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ form the relay path for data from node 1 while nodes $7 \rightarrow 8 \rightarrow 9 \rightarrow 5 \rightarrow 6$ form the relay path for data from node 7. Data might be *aggregated* into one stream at node 5. This is not the only feasible role assignment that allows the source to be sensed. For instance, node 10 could act as the second sensor instead of node 7 and $10 \rightarrow 7 \rightarrow 8 \rightarrow 4 \rightarrow 5 \rightarrow 6$ could form the corresponding relay path. Also, node 6 might aggregate the data instead of node 5 etc. Finally, note how the sensor, aggregator and relay roles must change as the source moves from S_0 to S_1. At every instant, the following decisions must be made: which sensor(s) to use, whether to aggregate or not, where to aggregate, what fraction to aggregate, how to route data to the aggregator, how to route aggregated data and how to account for changes in source location. This paper demonstrates a computationally feasible methodology to upper bound the lifetime that *any* collaborative protocol that makes these decisions can ever hope to achieve.

which must now be relayed to the basestation. Hence the sensor sub-role is really a "sense and transmit" sub-role and is qualified by a single attribute – the node that receives the raw sensor data or simply the destination node.

- Relay: The node simply forwards the received data onward without any processing. A relay sub-role is qualified by two attributes – the source and destination nodes of the data being relayed.
- Aggregator: The node receives two or more raw data streams and then aggregates them into a single stream. While the actual mechanism is application dependent, the underlying motivation is the same – the total volume of data to be routed to the basestation is reduced and the quality of the aggregated stream is higher than that of the raw streams from which it is derived [12]. Consider a sensor network that detects tank intrusion in a specified region. Several nodes might declare a tank present with varying levels of confidence. All these "tank detected" messages may be routed to a node that aggregates them into a single message with a revised confidence measure. Aggregation here corresponds to *data-fusion*. As another example, consider a sensor network collecting acoustic data. When an acoustic event occurs, sensors record it with varying signal-to-noise ratios (SNRs). In this case aggregation might entail *beamforming* these different streams to obtain a single aggregated stream with enhanced SNR [13]. We classify aggregation into *non-hierarchical* and *hierarchical* varieties. In the former, all raw streams are aggregated at a single node, while the latter version allows aggregation of partially aggregated streams. The aggregation sub-role is qualified by two attributes – the set of nodes transmitting raw data to be aggregated and the destination node receiving the aggregated stream.

Note that while nodes change their roles with time, we assume that their locations are fixed.

B. *Defining Lifetime*

A data gathering network can be in one of the following states:

1. Source present in region but network not sensing. This is termed "loss of coverage".
2. Source present and network sensing while satisfying user dictated constraints. This state is termed "active".
3. Source present and network sensing but not satisfying user dictated constraints. This state is termed "quality failure".
4. No source present in the region.

In non-mission-critical applications, a reasonable definition of lifetime is the cumulative active time of the network (i.e. whenever the network is active its lifetime clock is ticking, otherwise not). In mission-critical applications, lifetime is defined as the cumulative active time of the network until the first loss of coverage or quality failure. In this paper, we adopt this latter definition of lifetime. Note that active lifetime is different from *physical* lifetime. For instance, a sensor network deployed to detect tank intrusion can "live on" forever (ignoring battery degradation, leakage etc.) in the absence of activity. But it can only detect, say, 1000 hours of tank intrusion.

C. *Node Composition and Energy Models*

Despite the many implementations [1], [14], [15], integrated wireless sensor nodes have the same overall composition illustrated in figure 2.

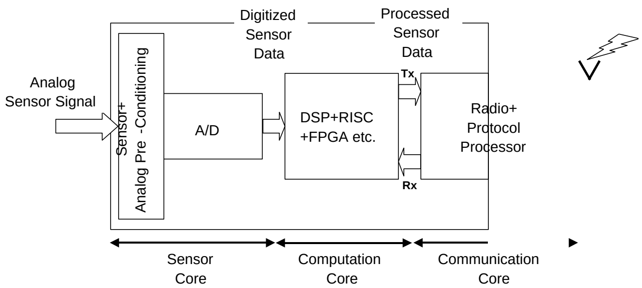

Fig. 2. Composition of the wireless sensor node.

C.1 Sensor Core

We assume that the energy needed to sense a bit is a constant (α_3). Thus, for a sensing rate given by r bits/sec, the sensing power is simply $p_{sense} = \alpha_3 r$. A typical value of α_3 is 50 nJ/bit [10].

C.2 Computation Core

We assume the following model for the power dissipation when n_{agg} raw streams are aggregated into a single stream [16]:

$$p_{comp} = n_{agg}\alpha_4 r = n_{agg} p_{agg} \tag{1}$$

where r is the rate (in bits/sec) of each of the n_{agg} streams (and also that of the output stream) and α_4 is a constant. Note that p_{comp} just represents the energy dissipated in the computational core. The energy costs of receiving the streams and transmitting the aggregated stream are accounted for separately. The parameter α_4 can be anywhere from a pJ/bit to 10s of nJ/bit depending on the type of aggregation and the architecture used.

C.3 Communication Core

We use the following models for the communication core [17]:

$$p_{tx}(n_1, n_2) = (\alpha_{11} + \alpha_2 d(n_1, n_2)^n) r \tag{2}$$
$$p_{rx} = \alpha_{12} r \tag{3}$$

where $p_{tx}(n_1, n_2)$ is the power dissipated in node n_1 when it is transmitting to node n_2, $d(n_1, n_2)$ is the distance between the two nodes, n is the path loss index, and the αs are positive constants. Typical values of these parameters are α_{11}=45 nJ/bit, α_{12}=135 nJ/bit, α_2=10 pJ/bit/m^2 (n=2) or 0.001 pJ/bit/m^4 (n=4) [10].

III. Bounds Using Optimal Role Assignments

In previous work [11], we tackled the following problem:

Given the region of observation ($\mathcal{R}$), the source radius of observability (d_S), the node energy parameters (α_{11}, α_{12}, α_2, α_3 and n), the number of nodes deployed (N), the initial energy in each node (E), what is the upper bound on the active lifetime (t) of *any* network established using these nodes which gathers data from a source residing in $\mathcal{R}$ according to a specified spatial p.d.f. $l_{source}(x, y)$[1].

The key steps we used to derive tight or near-tight bounds were:
1. Computing the minimum cumulative energy needed to relay a bit over a certain distance, where the minimum is calculated over all possible multi-hop topologies. If p_{tx} is a convex function of distance, this minimum turns out to be easy to compute.
2. Deriving a lower bound on the expected power dissipation in an ad-hoc network.
3. Using energy conservation to derive an upper bound on lifetime using the lower bound on average power.

Using this technique, the bound on the lifetime of a network gathering data from a source residing in a certain region is given by:

$$t \leq \frac{N.E}{\left((\alpha_{11} + \alpha_{12})\frac{n}{n-1}\frac{d_{geom} - d_S}{d_{char}} - \alpha_{12} + \alpha_3\right) r} \tag{4}$$

where,

$$d_{char} = \sqrt[n]{\frac{\alpha_1}{\alpha_2(n-1)}}$$

and d_{geom} is a function of the geometry of $\mathcal{R}$ and the p.d.f. $l_{source}(x, y)$. Note that these bounds hold in a limit-theorem sense i.e. the probability the bound holds can be made arbitrarily close to 1.

The bounds derived using this technique allow quick estimates of the maximum possible lifetime. However, they do not factor in the topology of the network and are derived for non-aggregating networks. We now introduce a new framework that eliminates these deficiencies.

A. The Role Assignment Framework

We introduce the simple and intuitive concepts of roles and role-assignments in sensor networks.

Definition 1 (Role). *A role is composed of one or more instances of the three sub-roles introduced earlier – sensing,*

[1]When we say a source "resides in a region" according to a spatial p.d.f., we mean that its successive locations in that region are i.i.d. random variables that are governed by that p.d.f.

relaying and aggregating. At most one sensor sub-role can be assumed. Several relay sub-roles can be assumed only when data streams being relayed originated from distinct sensor nodes. Similarly, a role can have several aggregation sub-roles only if those sub-roles aggregate data from distinct originating sensors. The unique role that has no sub-roles is called the dormant role.

Definition 2 (Role Assignment). *An assignment of roles to nodes in a network constitutes a role assignment.*

Definition 3 (Feasible Role Assignment (FRA)). *A role assignment is termed feasible if it:*

1. Results in data being relayed from the minimum specified number of sensors to the basestation, and,

2. Has no redundancy i.e. no sub-role in any node can be deleted while still obeying the first property,

We denote FRAs by f and the set of all feasible role assignments by $\mathbf{F}$. The power dissipated in node i when FRA f is being sustained by the network is denoted by $p(i, f)$. The energy models discussed earlier lead to simple, closed form expressions for $p(i, f)$.

Consider the example network in figure 3. If we constrain node 1 to be the sensor, there are 4 FRAs given by,

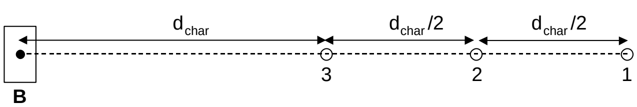

Fig. 3. A collinear 3-node network with node 1 as the assigned sensor (d_{char} is 134 meters).

$$
\begin{aligned}
F = &\{f_1, f_2, f_3, f_4\}, \text{ where,}\\
&f_1 : 1 \rightarrow B\\
&f_2 : 1 \rightarrow 2 \rightarrow B\\
&f_3 : 1 \rightarrow 3 \rightarrow B\\
&f_4 : 1 \rightarrow 2 \rightarrow 3 \rightarrow B
\end{aligned}
$$

There are in fact 8 FRAs, but we have ignored the "self-crossing" ones like $1 \rightarrow 3 \rightarrow 2 \rightarrow B$ because they are clearly sub-optimal in collinear networks. Note that there can be no aggregation since only one, pre-assigned node senses.

Consider, the same network, but with the constraint that both 1 and 2 must sense and aggregation is allowed. This leads to,

$$
\begin{aligned}
&F = \{f_i | 1 \le i \le 12\}, \text{ where,}\\
&\left.\begin{array}{l}
f_1 : 1 \rightarrow B; 2 \rightarrow B\\
f_2 : 1 \rightarrow 2 \rightarrow B; 2 \rightarrow B\\
f_3 : 1 \rightarrow 3 \rightarrow B; 2 \rightarrow B\\
f_4 : 1 \rightarrow 2 \rightarrow 3 \rightarrow B; 2 \rightarrow B\\
f_5 : 1 \rightarrow B; 2 \rightarrow 3 \rightarrow B\\
f_6 : 1 \rightarrow 2 \rightarrow B; 2 \rightarrow 3 \rightarrow B\\
f_7 : 1 \rightarrow 3 \rightarrow B; 2 \rightarrow 3 \rightarrow B\\
f_8 : 1 \rightarrow 2 \rightarrow 3 \rightarrow B; 2 \rightarrow 3 \rightarrow B
\end{array}\right\} \text{Non-aggregating}\\
&\left.\begin{array}{l}
f_9 : 1 \rightarrow \mathbf{2} \rightarrow B; \mathbf{2} \rightarrow B\\
f_{10} : 1 \rightarrow \mathbf{2} \rightarrow 3 \rightarrow B; \mathbf{2} \rightarrow 3 \rightarrow B\\
f_{11} : 1 \rightarrow \mathbf{3} \rightarrow B; 2 \rightarrow \mathbf{3} \rightarrow B\\
f_{12} : 1 \rightarrow 2 \rightarrow \mathbf{3} \rightarrow B; 2 \rightarrow \mathbf{3} \rightarrow B
\end{array}\right\} \text{Aggregating}
\end{aligned}
$$

The reader may wish to verify that this is an exhaustive list of all non-self-crossing FRAs that allow aggregation. The first 8 FRAs are non-aggregating. In the first FRA (f_1) for instance, the data from node 1 (which is sensing) is routed straight to the basestation and the same is true for node 2 (also sensing). Consider now an aggregating FRA, say f_{12}. Here, node 3 is the aggregator[2]. Node 1 is sending its raw sensor data to 3 via 2 while 2 is sending its data straight to 3. 3 is aggregating these two streams into one and sending the aggregated stream straight to the basestation.

Definition 4 (Feasible Collaborative Strategy). *A collaborative strategy specifies the FRAs used by a network to fulfil its contract. Specifically, a collaborative strategy is simply a $|\mathbf{F}|$-tuple where the i^{th} element specifies the time (possibly zero) for which the corresponding FRA is sustained. A collaborative strategy is termed feasible (with respect to the network's initial energy state) if, after its execution, all nodes have non-negative residual energy. The lifetime achieved by a feasible collaborative strategy is simply the sum of the elements of the $|\mathbf{F}|$-tuple.*

We denote feasible collaborative strategies by c and the (infinite) set of all such strategies by $\mathcal{C}$. The lifetime achieved by c is denoted by $t(c)$.

Proposition 5 (Lifetime Bound). *The lifetime t, achieved by a wireless sensor network is upper bounded thus:*

$$t \le \max_{c \in \mathcal{C}} \; t(c) \tag{5}$$

Proof: At any instant, an active network must sense the source using the specified minimum number of sensors and deliver bits to the basestation. Every bit is delivered via some assignment of roles to nodes. In practical networks, there are energy overheads incurred for, say, protocol operations (like media access control etc.). Hence,

[2]The aggregator has been emphasized in aggregating FRAs.

the lifetime computed in the absence of such overheads is a valid upper bound on lifetime. Finally, note that, by construction, using role assignments that are non-feasible cannot improve the bound. Hence we take the maximum only over strategies that use FRAs.

Finding the quantity on the RHS of (5) turns out to be a straightforward linear programming problem as shown in table I. The first set of constraints is obvious - it makes no physical sense to sustain a FRA for negative time. The second set of constraints are energy conservation constraints, one for each node, with the LHS denoting the energy consumed and the RHS the initial energy the node started out with. The solution of this linear program yields the optimal collaborative strategy and as a result a bound on the lifetime of the network.

Objective :

$$\max \qquad t = \sum_{i=1}^{|F|} t_i$$

where t_i corresponds to the time for which FRA f_i is sustained.

Constraints :

$$t_j \geq 0 \quad : \quad 1 \leq j \leq |F| \tag{6}$$

$$\sum_{j=1}^{|F|} p(i, f_j) t_j \leq e_i \quad : \quad 1 \leq i \leq N \tag{7}$$

TABLE I
LINEAR PROGRAM FOR DETERMINING OPTIMAL COLLABORATIVE STRATEGY

For the example network in figure 3 with no aggregation, the optimal collaborative strategy turns out to be (0,0.375,0.375,0.625) sec i.e. the first FRA is not used at all, the second is used for 0.375 sec and so on. This yields a lifetime bound of 1.375 sec[3]. For the aggregating case with two sensors, the optimal turns out to be t_6 = 0.3192, t_8 = 0.8938 and t_{10} = 0.3192 sec, with other FRAs sustained for zero time, yielding a bound of 1.532 sec.

In the next four sections, we will use this role assignment framework to include network topology, sets of potential sensors, aggregation and source movement.

[3]The reader should not be alarmed to see lifetimes of mere seconds! This is because the initial energy was purposely set to 180 nJ to yield lifetimes around a second. Real-world nodes start off with several hundreds or thousands of joules. Note that lifetime bounds derived vary linearly with the initial energy per node.

B. Topology Sensitive Bounds

While the framework in table I is conceptually straightforward to use to tackle topology, it suffers from a computational drawback since the number of FRAs grows exponentially with the number of nodes. Interestingly however, we will show that for a broad class of role assignment problems, we can derive our bounds in a time that is *polynomial* in the number of nodes. Specifically, we reason that if the role assignment problem can be transformed to *resemble* a network-flow problem, a computationally tractable solution might follow. This reasoning is motivated by prior work in the area of energy efficient multi-hop routing in ad-hoc networks [6], [7].

To illustrate this idea, consider once again the network in figure 3. One way to view the optimal strategy is that f_2 is sustained for 0.375 sec, f_3 for 0.375 sec and f_4 for 0.625 sec. In other words, f_2 and f_3 are each responsible for shipping $\frac{3}{11}$ of the data while f_4 ships the remaining $\frac{5}{11}$. It follows that link $1 \rightarrow 2$ carries $\frac{8}{11}$ of the data. Similarly link $1 \rightarrow 3$ is responsible for shipping $\frac{3}{11}$, all of which is due to f_3. Figure 4, demonstrates this transformation. We

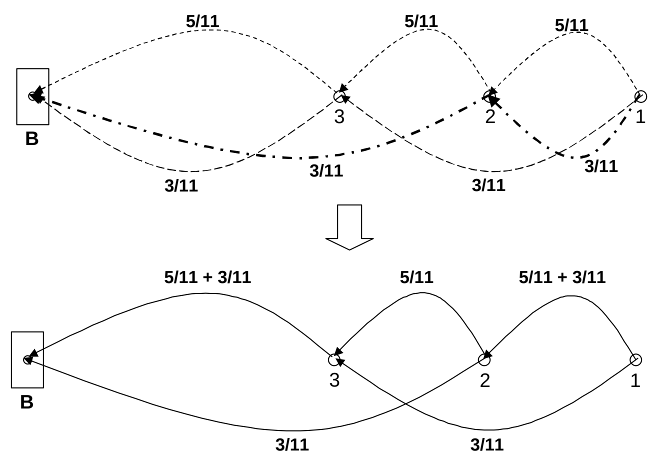

Fig. 4. Deriving a flow view from a role assignment view.

have transformed the role assignment view to the network flow view:

$$\left\{ \begin{array}{l} t_1 = 0 \text{ sec} \\ t_2 = 0.375 \text{ sec} \\ t_3 = 0.375 \text{ sec} \\ t_4 = 0.625 \text{ sec} \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} f_{12} = \frac{8}{11} \\ f_{13} = \frac{3}{11} \\ f_{1B} = 0 \\ f_{23} = \frac{3}{11} \\ f_{2B} = \frac{5}{11} \\ f_{3B} = \frac{8}{11} \end{array} \right\}$$

where f_{ij} is the flow from node i to node j. Note that we are justified in calling the above view a flow because it satisfies the following properties expected of any valid flow:

1. **Non-negativity** of flow.
2. **Conservation** of flows at all nodes but the sensor. In other words the total flow out of a node is the same as the total flow into a node.

It is fairly straightforward to see that for the class of networks with an assigned sensor (as in the flow construction example above), every flow view that is constructed from a collaborative strategy will have these properties. We now ask the reverse question - can one always derive a feasible collaborative strategy given a flow and the total lifetime?[4] It turns out that we can, since every flow can be expressed as a sum of cycles and paths from the source to sink with non-negative weights [18]. These paths correspond to FRAs for the simple case of a fixed point source with an assigned sensor. This case turns out to be the maximum lifetime multi-hop routing which was first reported in [6], [7]. We can now replace the program in table I with that in table II. We have labelled the pre-assigned sensor node 1 and the basestation is node N+1. The first two constraints simply state that this is a valid flow. The third constraint normalizes the flow out of the sensor. This ensures that if the flows above can be sustained for time t, then t is simply the lifetime of the network. The last condition states that the total energy drain of any node be no greater than the initial energy present in that node. Replacing each unknown flow f_{ij} with a new unknown $r_{ij} = f_{ij}t$ ensures that the program is linear in the new unknowns. Also, the number of constraints and variables are now polynomial in the number of nodes and we can compute the upper bound in polynomial time.

Objective :

$$\max \quad t$$

Constraints :

Non-negativity of flow:

$$f_{ij} \geq 0 \tag{8}$$

Conservation of flow:

$$\sum_{\substack{s\in[1,N+1]\\ s\neq i}} f_{si} = \sum_{\substack{d\in[1,N+1]\\ d\neq i}} f_{id} : \quad i \in [2, N] \tag{9}$$

Total sensor flow:

$$\sum_{d\in[2,N+1]} f_{1d} - \sum_{s\in[2,N+1]} f_{s1} = 1 \tag{10}$$

Energy constraints:

$$t\Bigg(\sum_{\substack{d\in[1,N+1]\\ d\neq i}} p_{tx}(i,d) f_{id} + \sum_{\substack{s\in[1,N+1]\\ s\neq i}} p_{rx} f_{si} + \underbrace{p_{sense}}_{\textit{For node 1 only}} \Bigg) \leq e_i : \quad i \in [1, N]$$

TABLE II

PROGRAM FOR COMPUTING BOUND USING FLOW VIEW.

[4]Knowledge of the total lifetime allows one to determine the *absolute*, rather than relative, durations for which the FRAs must be sustained.

C. Set of Potential Sensors

In practical networks, we must deal with a set of potential sensors (S) of which a specified number (k, $1 \leq k \leq m = |S|$) must be active. For instance, in the example shown in figure 1, we required $k = 2$. Also, $S = \{1, 7, 10\}$ when the source is at location S_0. The case of a pre-assigned sensor considered in the last section is simply corresponds to $k = m = 1$. While our original linear program (table I), can handle this new problem, the problem is again computational complexity. Rather, we will resort to the flow view again. Two modifications to the program in table II allow us to capture this "k of m" sensors problem:

Objective :

$$\max \quad t \tag{12}$$

Constraints :

Non-negativity of flow:

$$f_{ij} \geq 0 \tag{13}$$

Flow conservation:

$$\sum_{\substack{s\in[1,N+1]\\ s\neq i}} f_{si} = \sum_{\substack{d\in[1,N+1]\\ d\neq i}} f_{id} : \quad i \in [1, N], i \notin S \tag{14}$$

Overall flow from sensor set (S):

$$\sum_{i\in S} \left(\sum_{\substack{d\in[1,N+1]\\ d\neq i}} f_{id} - \sum_{\substack{s\in[1,N+1]\\ s\neq i}} f_{si} \right) = k \tag{15}$$

Non-consumption of flow in sensors:

$$\sum_{\substack{d\in[1,N+1]\\ d\neq i}} f_{id} - \sum_{\substack{s\in[1,N+1]\\ s\neq i}} f_{si} \geq 0 : \quad i \in S \tag{16}$$

Limit on total flow from any single sensor:

$$\sum_{\substack{d\in[1,N+1]\\ d\neq i}} f_{id} - \sum_{\substack{s\in[1,N+1]\\ s\neq i}} f_{si} \leq 1 : \quad i \in S \tag{17}$$

Energy constraints:

$$t\Bigg\{\sum_{\substack{d\in[1,N+1]\\ d\neq i}} p_{tx}(i,d)f_{id} + \sum_{\substack{s\in[1,N+1]\\ s\neq i}} p_{rx}f_{si} + \underbrace{p_{sense}\left(\sum_{\substack{d\in[1,N+1]\\ d\neq i}} f_{id} - \sum_{\substack{s\in[1,N+1]\\ s\neq i}} f_{si}\right)}_{\textit{Term to be included for potential sensors only}}\Bigg\} \leq e_i : \ i\in[1,N] \tag{18}$$

First, (10) has been modified to (15) because instead of unit flow from an assigned sensor, we now desire the net sensor flow out of S to be k. Simply equating the *total* flow out of S to be k does not guarantee that the resulting flow solution will have a meaningful equivalent collaborative strategy. We have to guard against two possibilities – nodes in S that consume data or that produce too much data[5]. To see why the latter situation is a problem, consider, a scenario where m is 5 and k is 2. Solving the program might yield a solution where one of the nodes in S accounts for the entire flow of 2. Clearly, such a flow cannot be translated to a collaborative strategy where at least two nodes sense when the network is active. Constraint (16) precludes sensors from consuming data while (17) prevents sensors from producing too much data. It turns out that these constraints are not just necessary but also sufficient to guarantee the existence of an equivalent strategy. In other words, given a set of m flows (one for each sensor in S), all of which are no greater than 1 and add up to k, we can always guarantee that there exists a schedule where exactly k of m sensors are active at any time. We omit the simple constructive proof. Once again using a change of variables identical to that used in the previous section, we end up with a linear program which can be solved in polynomial time. Hence, lifetime bounds for the k of m problem can also be derived in polynomial time.

D. Bounds for Aggregating Networks

Aggregation can significantly increase the lifetime of a network by reducing the volume of data that needs to be relayed to the basestation. We now generalize the k of m scenario to include non-hierarchical aggregation. Instead of simply deciding which sensors to use from the potential sensor-set and how to route data, we must now decide what fraction of the data to aggregate, how many and which nodes to use as aggregators, how to route data to the aggregator(s) and from the aggregators to the basestation. Of course, we must also find out how to change these decisions with time (which corresponds to how long we should sustain these FRAs) so as to maximize lifetime.

This represents an important conceptual leap since we are exploring the fundamental tradeoff between computation and communication (as captured by the cost of aggregating versus the cost of relaying data, respectively), a theme which has not seen rigorous analysis in previous work. We want to repeat that the general role assignment framework in table I is capable of producing a bound, but will get computationally burdensome for large N. Hence, we focus on a transformation to a flow-centric view instead.

While bits can originate from several sensors in a non-aggregating network, they undergo no change till they reach their common, final destination – the basestation. Hence, at any instant, there is a single commodity flowing through the network. Consider an aggregating network with, say, three, potential aggregators - nodes 4, 6 and 7[6]. Now, bits have *four* potential destinations - the basestation and these aggregators. Bits that go unaggregated or that are produced as a result of aggregation comprise the first commodity (which we abbreviate as *unagg*). Bits destined for aggregation at one of these aggregating nodes comprise the other three commodities (which we term *agg* commodities). This leads to $1+|P|$ commodities in general. This is reminiscent of a *multi-commodity flow* problem except that the commodities are not distinguished based on their source but their destination and flows are not conserved when raw streams are aggregated.

In the program that follows, flow $f_{ij,z}$ indicates flow on the link $i \to j$ carrying commodity z i.e. destined for aggregation at node z. We use $z = 0$ for the case when the flow will not be aggregated i.e. for the *unagg* commodity. Hence $z \in \{0\} \cup P$ denotes z running over all commodities. Given a node, any flow originating from or terminating in it is said to be *related* to the node if the commodity it carries is destined for the node. Hence, only flows of type $f_{ij,i}$ or $f_{ji,i}$ are related to node i. It follows that nodes not in P have no related flows.

Objective :

$$\max \quad t \tag{19}$$

General Constraints :
Non-negativity of flow:

$$f_{ij,z} \geq 0 : \quad i,j\in[1,N+1], z\in\{0\}\cup P \tag{20}$$

[5]Note that (14) imposes flow conservation only on nodes $\notin S$.

[6]We denote the set of potential aggregators by P.

Absence of related aggregated flow in output:

$$f_{id,i} = 0 : \qquad i \in P, d \in [1, N+1], d \neq i \tag{21}$$

Overall flow from the sensor set S:

$$\sum_{\substack{z \in \{0\} \cup P \\ z \neq i}} \sum_{i \in S} \left(\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} - \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} \right) = k \tag{22}$$

Energy constraints:

$$t \Bigg\{ \sum_{\substack{d \in [1,N+1] \\ d \neq i}} p_{tx}(i,d) \sum_{z \in \{0\} \cup P} f_{id,z} + \underbrace{p_{agg} \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,i}}_{\substack{\text{Term to be included for} \\ \text{potential aggregators only}}}$$

$$+ \underbrace{p_{sense} \sum_{\substack{z \in \{0\} \cup P \\ z \neq i}} \left(\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} - \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} \right)}_{\substack{\text{Term to be included for} \\ \text{potential sensors only}}} +$$

$$\sum_{\substack{s \in [1,N+1] \\ s \neq i}} p_{rx} \sum_{z \in \{0\} \cup P} f_{si,z} \Bigg\} \leq e_i : i \in [1, N] \tag{23}$$

Conservation Constraints :
Conservation of *agg* commodities in the basestation:

$$\sum_{s \in [1,N]} f_{s(N+1),z} = \sum_{d \in [1,N]} f_{(N+1)d,z} : z \in P \tag{24}$$

Conservation of flow in nodes that neither sense nor aggregate:

$$\sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} = \sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} \quad : \quad i \in [1,N] - \{S \cup P\}, z \in \{0\} \cup P \tag{25}$$

Conservation of *unrelated* flow in aggregators that do not sense:

$$\sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} = \sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} : \quad i \in P - S, z \in P, z \neq i \tag{26}$$

Aggregation Constraints : Compression of *related* flow in aggregators that do not sense:

$$\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,0} - \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,0} = \frac{1}{k} \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,i} \quad : \quad i \in P - S \tag{27}$$

Non-consumption of flow in sensors :
Non-consumption of unrelated flows in sensors:

$$\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} \geq \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} : \quad i \in S, z \in P, z \neq i \tag{28}$$

Non-consumption of the *unagg* commodity in sensors that are not aggregators:

$$\sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,0} \geq \sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,0} : \quad i \in S - P \tag{29}$$

Non-consumption in sensors that are aggregators:

$$\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,0} - \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,0} \geq \frac{1}{k-1} \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,i} \quad : \quad i \in S \cap P \tag{30}$$

Limits on sensor flows :
Limits on total flow from any single sensor:

$$\sum_{z \in \{0\} \cup P, z \neq i} \left(\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} - \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} \right) \leq 1 : i \in S \tag{31}$$

Limits on sensor flow destined for aggregation:

$$\sum_{\substack{d \in [1,N+1] \\ d \neq i}} f_{id,z} - \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{si,z} \leq \frac{1}{k} \sum_{\substack{s \in [1,N+1] \\ s \neq i}} f_{sz,z} \quad : \quad i \in S, z \in P, z \neq i \tag{32}$$

We now explain the constraints in the program. Constraint (21) states that if a flow originates from an aggregator, it must not carry any commodity that is destined for that aggregator. (22) is essentially the same as (15) with the only difference that since we have multiple commodities here, we must sum over all of them. Constraint (23) is a similar extension of (18).

Constraint (24) ensures that *agg* commodities are conserved at the basestation whereas *unagg* commodities are unconstrained[7]. In nodes that neither sense nor aggregate i.e. which only route, conservation constraints are the same as before – every commodity is conserved as expected (25). In the case of an aggregating node that does

[7] An interesting side effect of aggregation is that it can be beneficial to route *agg* commodities at the basestation back into the network!

not sense, *unrelated* flows must be conserved (26) while the the net *unagg* flow is augmented by the volume of the aggregated stream (27). Note that this volume is simply $1/k^{\text{th}}$ the volume of the total inflow destined for aggregation.

The single non-consumption constraint (16) stated earlier now leads to three constraints. *Related agg* commodities are already taken care of by (27). Constraint (28) handles *unrelated agg* commodities and is conceptually identical to (16). Next, consider the *unagg* commodity. For sensors that are not aggregators, the constraint on the *unagg* commodity is straightforward (29). For sensors that are also aggregators, the *unagg* output must not only be greater than the *unagg* input, but greater by the volume of the aggregated stream produced at the node (30). Note that we use $1/(k-1)$ and not $1/k$ since one stream destined for aggregation must come from the node itself.

The final constraints are the limits on sensor flows, which were motivated in the last section. The first of these (31) is the same as (17) and prevents any sensor from monopolizing the output from the sensor group. We also need to prevent a sensor from monopolizing the sensor flow *to a particular aggregator* which is achieved by (32).

We have omitted the more complex case of hierarchical aggregation due to lack of space. The interested reader is referred to [19], where we show that bounds can be obtained in a computationally feasible fashion for a constrained form of hierarchical aggregation.

E. Extensions to Arbitrary Regions

In the previous sections, we showed how bounds can be refined to include topology, potential sensor sets and aggregation. Throughout, we implicitly assumed a fixed point source. We now extend our framework to allow sources that reside in regions.

Two insights lead to a solution of this problem. First, we transform the problem of *a single source residing in a region* to a problem of *multiple sources at fixed points*. Next, we use a simple trick to extend our single-source framework to accommodate multiple sources.

We use the network in figure 5 to illustration the first idea. We divide the region of observation ($\mathcal{R}$) into sub-regions characterized by the set of potential sensors. In our example, there are twelve sub-regions. Consider sub-region e, which is characterized by $S = \{1, 2, 3\}$. Also, we expect the source to reside in e for a fraction of time (say, η) equal to the integral of the location p.d.f. evaluated over e. We can similarly characterize every sub-region. We now have a problem which is identical to sensing 12

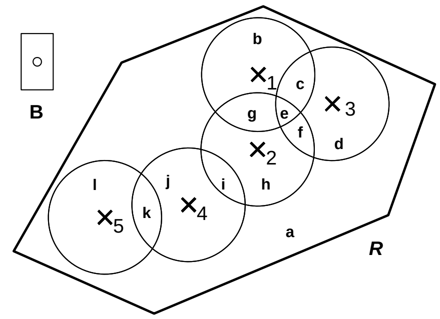

Fig. 5. Dividing a region into sub-regions based on sets of potential sensors.

sources – one in every sub-region[8]. We want to determine the maximum possible lifetime t such that the source residing in sub-region j can be sensed for a certain fraction η_j of the lifetime.

We extend our single source framework to attack the multiple source problem as follows. We deal with L sources by splitting each node into L *virtual* nodes, one corresponding to each source. We then setup constraints for the l^{th} source ($1 \leq l \leq L$) as we did for the single source case, with the following modifications:

1. Flows $f_{ij,z}$ are now labelled $f_{ij,z,l}$ where the new subscript indicates that the flows correspond to source l.
2. The overall flow from the sensor-set is set to $\eta_l k$ instead of k (cf. (22)). Similarly, the RHS in (31) is changed from 1 to η_l.
3. We do not impose the energy constraint (23).

After setting up these constraints for the L sources, we setup a *single* energy constraint for each node by constraining the sum of the energy consumptions of its L virtual copies to be less than the initial energy in that node. The objective is the same as before - maximizing t. We claim that this new program solves our optimization problem.

To see why, first observe that any solution produced by this program is in fact feasible. We can start with the first source, map the flows corresponding to this source to the actual nodes (we can do this since the flows obey single source constraints) and allow them to run for $\eta_1 t$ time. We then do this, in sequence, for all the L sources. Our unified energy constraint guarantees that we cannot run out of energy in the interim.

Next, we claim that any solution to the multiple sources problem can always be expressed in the form above i.e. using the language of virtual nodes. Consider any such solution. It must correspond to a feasible collaborative strategy for sensing these multiple sources. We can group all FRAs which are used for sensing the first source, all FRAs

[8] Fixed at *any* point in that sub-region.

used for the second source and so on[9]. Since each of these groups of FRAs can be translated into flows that obey single source constraints, we end up with a virtual node view.

Note that this virtual-node based approach leads to a solution that runs in time that is polynomial in the number of nodes and locations, L. In practical sensors networks, the region of observation ($\mathcal{R}$) is much larger than the source's region of observability. Furthermore, the node density and source observability radius are fixed parameters that do not change with $\mathcal{R}$. Under these assumptions, L is proportional to the area of $\mathcal{R}$ and as a consequence to the number of nodes N.

Finally, it is worth mentioning that the same technique follows through if we are dealing with multiple moving sources whose trajectories are specified.

IV. Illustrations

We present two illustrations of the techniques developed in this paper. The first example demonstrates how bounds behave when the number of sensors that need to observe the source change. In figure 6, we have a set of ten potential sensors. We vary the number of desired observers from one to the maximum possible, ten. For each value of k, our k-of-m program presented earlier computes the bound on lifetime. The resulting bounds on lifetime are in figure 7.

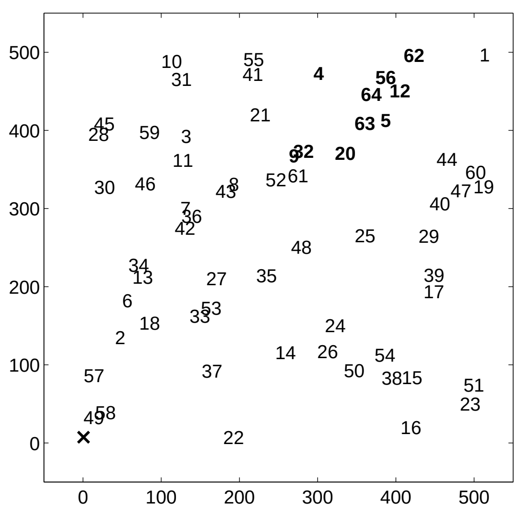

Fig. 6. The 64 node network with S={4,5,9,12,20,32,56,62,63,64} and k varying from 1 to 10 that is the basis for figure 7. The basestation is marked $\times$ and distances are in meters.

Next, we see the impact of aggregation via the example network in figure 8. Here, we have a set of three potential sensors and we desire maximum quality (k=3), but we

[9]Note that if a FRA allows several sources to be sensed simultaneously, then it can be split into constituent FRAs that sense individual sources.

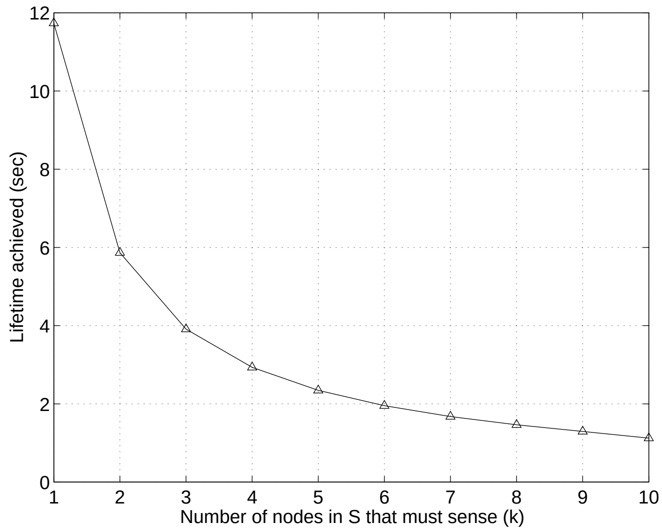

Fig. 7. Lifetime bounds of the network in figure 6 as a function of the minimum number of nodes that must sense (k).

allow non-hierarchical aggregation. We ran our programs for the aggregated and unaggregated versions. They reveal that aggregation has the potential to increase lifetime by a factor of 2.67 over the unaggregated case if aggregation costs are small compared to receive costs.

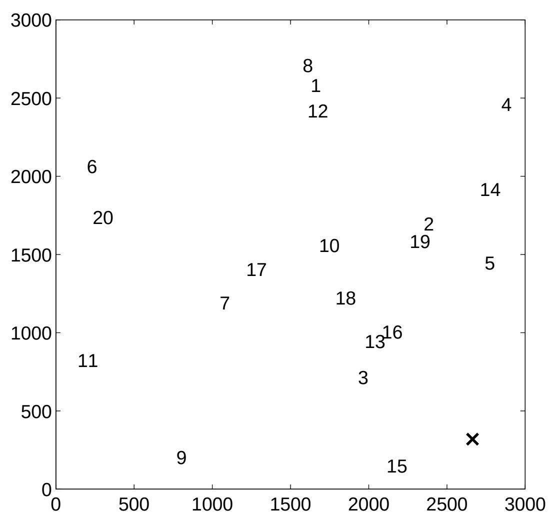

Fig. 8. An example 20 node network with S={1,8,12}, k=3 and P={1:20}.

V. Conclusions

In previous work, we derived tight or near-tight but topology insensitive bounds for non-aggregating sensor networks where the source resided in a region according to a location p.d.f. In this paper, we generalized our bounds to the case of aggregating networks with specified topology and even source movement.

These bounds were derived by employing the formalism of feasible role assignments (FRAs). We argued that there is a finite set of assignments of roles to nodes that allow sensing in a non-redundant manner. Every bit received at the basestation must have employed one of these FRAs. The question then is – what FRAs must we use and in what proportion such that lifetime is maximized. The resulting lifetime provides a bound on the lifetime of actual networks.

The role assignment technique is conceptually simple and extremely powerful since it can allow arbitrarily complex ways of gathering data and still yield crisp bounds via a linear program. However, it is computationally burdensome. We showed in this paper that a class of role assignment problems permit a transformation to linear programs based on network flows that can be solved in polynomial time. It is important to emphasize that not all role assignment problems can be similarly transformed. But several ones of practical importance – pure routing, non-hierarchical and constrained hierarchical aggregation, multiple or moving sources, sources with specified trajectories – are amenable to such a transformation. An interesting open question is whether a similar transformation exists for networks that allow generalized hierarchical aggregation.

While the framework here accommodates fairly sophisticated data gathering scenarios, other practical concerns, chief amongst them the energy spent in the medium access control (MAC), remain to be incorporated. It is our hope that the techniques reported here will provide a starting point in constructing the ultimate bounds on the lifetime of data gathering wireless sensor networks.

REFERENCES

[1] G. Pottie and W. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, pp. 51–58, May 2000.

[2] K. Bult *et al.*, "Low power systems for wireless microsensors," in *Proc. of the International Symposium on Low Power Electronics and Design (ISLPED'96)*, pp. 17–21, Aug. 1996.

[3] A. Chandrakasan *et al.*, "Design considerations for distributed microsensor systems," in *Custom Integrated Circuits Conference (CICC)*, pp. 279–286, May 1999.

[4] E. Shih *et al.*, "Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks," in *Proc. of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 272–286, July 2001.

[5] S. Singh, M. Woo, and C. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proc. of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*, pp. 181–190, Oct. 1998.

[6] J.-H. Chang and L. Tassiulas, "Routing for maximum system lifetime in wireless ad-hoc networks," in *Proc. of the 37th Annual Allerton Conference on Communication, Control and Computing*, Sept. 1999.

[7] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *Proc. of IEEE INFOCOM 2000*, vol. 1, pp. 22–31, 2000.

[8] V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1333–1344, Aug. 1999.

[9] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Routing Protocols for Wireless Microsensor Networks," in *Proc. 33rd Hawaii International Conference on System Sciences (HICSS '00)*, Jan. 2000.

[10] W. Heinzelman, *Application-Specific Protocol Architectures for Wireless Networks*. PhD thesis, Massachusetts Institute of Technology, 2000.

[11] M. Bhardwaj, T. Garnett, and A. Chandrakasan, "Upper bounds on the lifetime of sensor networks," in *Proc. of the International Conference on Communications (ICC'01)*, June 2001.

[12] G. J. Pottie, "Hierarchical information processing in distributed sensor networks," in *International Symposium on Information Theory*, p. 163, 1998.

[13] K. Yao, C. Reed, R. Hudson, and F. Lorenzelli, "Beamforming performance of a randomly distributed sensor array system," in *Workshop on Signal Processing Systems (SIPS 97 - Design and Implementation)*, pp. 438–447, 1997.

[14] J. M. Rabaey, M. J. Ammer, J. L. da Silva, Jr., D. Patel, and S. Roundy, "Picoradio supports ad hoc ultra-low power wireless networking," *Computer*, vol. 33, pp. 42–48, July 2000.

[15] R. Min *et al.*, "An architecture for a power-aware distributed microsensor network," in *Proc. of the IEEE Workshop on Signal Processing Systems:Design and Implementation (SiPS'00)*, 2000.

[16] A. Wang, W. Heinzelman, and A. Chandrakasan, "Energy-scalable protocols for battery-operated microsensor networks," in *Proc. of the IEEE Workshop on Signal Processing Systems (SiPS'99)*, pp. 483–492, 1999.

[17] T. Rappaport, *Wireless Communications: Principles & Practice*. New Jersey: Prentice-Hall, Inc., 1996.

[18] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1 ed., Feb. 1993.

[19] M. Bhardwaj, "Power-Aware Systems," Master's thesis, Massachusetts Institute of Technology, 2001.