Modelling Data-Centric Routing in Wireless Sensor Networks

Bhaskar Krishnamachari, Deborah Estrin, and Stephen Wicker

Computer Engineering Technical Report CENG 02-14

Department of Electrical Engineering – Systems
University of Southern California
Los Angeles, California 90089-2562
213-740-4465

Modelling Data-Centric Routing in Wireless Sensor Networks

Bhaskar Krishnamachari[*], Deborah Estrin [†], Stephen Wicker[‡]

Abstract

Sensor networks differ from traditional networks in several ways: sensor networks have severe energy constraints, redundant low-rate data, and many-to-one flows. The end-to-end routing schemes that have been proposed in the literature for mobile ad-hoc networks are not appropriate under these settings. Data-centric technologies are needed that perform in-network aggregation of data to yield energy-efficient dissemination. In this paper we model data-centric routing and compare its performance with traditional end-to-end routing schemes. We examine the impact of source-destination placement and communication network density on the energy costs, delay, and robustness of data aggregation. We show that data-centric routing offers significant performance gains across a wide range of operational scenarios.

1 Introduction

The wireless sensor networks of the near future are envisioned to consist of hundreds to thousands of inexpensive wireless nodes, each with some computational power and sensing capability, operating in an unattended mode. They are intended for a broad range of environmental sensing applications from vehicle tracking to habitat monitoring [4], [28], [35]. The hardware technology for these networks - low cost processors, miniature sensing and radio modules are here today, with further improvements in cost and capabilities expected within the next decade [4], [16], [19], [28], [29]. The applications, networking principles and protocols for these systems are just beginning to be developed [9], [10], [14], [28].

Wireless sensor networks are similar to mobile ad-hoc networks (MANETs) in that both involve multi-hop communications. However, the nature of the applications and routing requirements for the two are significantly different in several respects. First, the typical mode of communication in a sensor network is from multiple data sources to a data recipient/sink - a sort of a reverse-multicast, rather than communication between any pair of nodes. Second, since the data being collected by multiple sensors is based on common phenomena, there is likely to

[*] B. Krishnamachari is at the Dept. of Electrical Engineering - Systems at the University of Southern California, Los Angeles, CA 90089, E-mail: bkrishna@usc.edu

[†] D. Estrin is at the UCLA Computer Science Department, Los Angeles, CA 90095, E-mail: destrin@lecs.cs.ucla.edu.

[‡] S. Wicker is at the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853, E-mail: wicker@ece.cornell.edu.

be some redundancy in the data being communicated by the various sources in sensor networks. Third, in most envisioned scenarios the sensors are not mobile (though the sensed phenomena may be), so the nature of the dynamics in the two networks is different. Finally, the single major resource constraint in sensor networks is that of energy. The situation is much worse than even that in MANETs, where the communicating devices handled by human users can be replaced or recharged relatively often. The scale of sensor networks and the necessity of unattended operation for months at a time means that energy resources have to be managed even more carefully. This, in turn, precludes really high data rate communication.

For these reasons the many end-to-end routing protocols that have been proposed for MANETs in recent years are not suitable for wireless sensor networks. Alternative approaches are required.

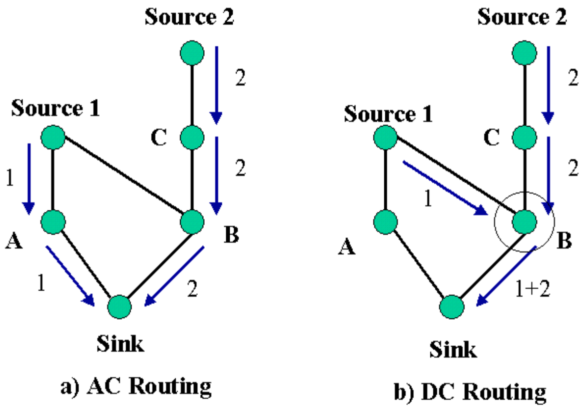
Data aggregation has been put forward as a particularly useful paradigm for wireless routing in sensor networks [13], [17]. The idea is to combine the data coming from different sources enroute – eliminating redundancy, minimizing the number of transmissions and thus saving energy. This paradigm shifts the focus from the traditional *address-centric* approaches (finding short routes between pairs of addressable end-nodes) to a more *data-centric* approach (finding routes from multiple sources to a single destination that allows in-network consolidation of redundant data).

We do not propose any new protocols in this paper, but rather attempt to address at a higher level the gains and tradeoffs that can be achieved by using the data-centric approach as opposed to the traditional address-centric approach.

The rest of the paper is organized as follows: in section 2, we define simple models of address-centric and data-centric routing protocols for the purpose of analysis. Since data aggregation is the key concept in data-centric routing, we examine optimal and suboptimal data aggregation methods, measures of performance and factors affecting these measures in section 3. In section 4 we present some theoretical results pertaining to the energy gains that can be achieved using data aggregation. This is followed by experimental results showing how the various suboptimal data-centric approaches compare with the optimal address-centric routing in terms of the average number of transmissions required per datum delivered to the data sink. Section 5 presents experimental results showing the effect of source placement, number and network density on the delay due to aggregation latency. In section 6 we discuss experimental results showing that data aggregation results in some degree of robustness to dynamics of the sensed phenomena. We caution the reader about the assumptions and omissions in our modelling in section 7. Our work is placed in the context of other relevant work in section 8. We present our conclusions in section 9.

2 Routing Models

We focus our attention on a single network flow that is assumed to consist of a single data sink attempting to gather information from a number of data sources. We start with simple models of routing schemes which use data aggregation (which we term data-centric), and schemes which do not (which we term address-centric). In both cases we assume there are some common elements - the sink first sends out a query/interest for data, the sensor nodes which have the appropriate data then respond with the data. They



a) AC Routing

b) DC Routing

Figure 1: Illustration of AC versus DC routing

differ in the manner the data is sent from the sources to the sink:

Address-centric Protocol (AC): Each source independently sends data along the shortest path to sink based on the route that the queries took ("end-to-end routing").

Data-centric Protocol (DC): The sources send data to the sink, but routing nodes enroute look at the content of the data and perform some form of aggregation/consolidation function on the data originating at multiple sources.

Figure 1 is a simple illustration of the difference between AC and DC schemes. In the address-centric approach, each source sends its information separately to the sink (source 1 routing the data labelled "1" through node A, and source 2 routing the data labelled "2" through nodes C and B). In the data centric-approach, the data from the two sources is aggregated at node B, and the combined data (labelled "1+2") is sent from B to the sink. The latter results in energy savings as fewer transmissions are required to send the information from both sources to the sink.

2.1 Differentiating Scenarios

One of goals in this paper is to understand the context in which data aggregation is useful. It is helpful to consider the following scenarios classified in terms of the type and dynamics of the data sent by the sources:

1. All sources send completely different information (no redundancy)

2. All sources send identical information (complete redundancy)

3. The sources send information with some intermediate, non-deterministic, level of redundancy

In case 1, Data aggregation cannot be performed - both AC and DC protocols will incur the same number of transmissions for the sink to receive all the data. In case 2, the AC protocol can be modified to do as well as or even better than the DC protocol by having the sink monitor the incoming information, realize that there is duplicate information coming in, and ask all but one of the sources to stop transmitting (assuming that this duplication is sustained over a period of time). In case 3, however, the AC protocol cannot be modified to do much better than the DC protocol - the non-deterministic nature of the redundancy implies that the sink cannot request some sources to shut down. Since all sources are required to transmit, and there is some redundancy in their transmission, the best option is to aggregate the information coming from these sources, which is what the DC protocol does. For the rest of the paper, it will be assumed that this scenario holds.

We now look at the notion of data aggregation in more detail.

3 Data Aggregation

Data aggregation is the combination of data from different sources, and can be implemented in a number of ways. The simplest data aggregation function is duplicate suppression - in the example of figure 1, if sources 1 and 2 both send the same data, node B will send only one of these forward. Duplication suppression is already practiced in commercial wireless messaging networks. Other aggregation functions could be *max*, *min*, or any other function with multiple inputs. For our modelling purposes in this paper we make a simplifying assumption - the aggregation function is such that each intermediate node in the routing transmits a single aggregate packet even if it receives multiple input packets. We will refer to the information received by the sink when it has obtained the messages transmitted by all sources in a given flow (whether or not these messages are aggregated) as a “datum”.

3.1 Optimal Aggregation

Say there are k sources, labelled S_1 through S_k, and a sink, labelled D. Let the network graph $G = (V, E)$ consist of all the nodes V, with E consisting of edges between nodes that can communicate with each other directly. With the assumption that the number of transmissions from any node in the data aggregation tree is exactly one, the data aggregation tree can be thought of as the reverse of a multicast tree: instead of a single source sending a packet to all receivers, all the sources are sending a single packet to the same receiver. It is well-known that the multicast tree with a minimum number of edges is a minimum Steiner tree on the network graph. The following can therefore be readily obtained:

Result 1: The optimum number of transmissions required per datum for the DC protocol is equal to the number of edges in the minimum Steiner tree in the network which contains the node set $(S_1, ...S_k, D)$.

Corollary: Assuming an arbitrary placement of sources, and a general network graph G, the task of doing DC routing with optimal data aggregation is NP-hard.

The latter follows from the NP-completeness of the minimum Steiner problem on Graphs [11].

3.2 Suboptimal Aggregation

The following are three generally suboptimal schemes for generating data aggregation trees that we examine in this paper.

1. **Center at Nearest Source (CNS)**: In this data aggregation scheme, the source which is nearest the sink acts as the aggregation point. All other sources send their data directly to this source which then sends the aggregated information on to the sink.

2. **Shortest Paths Tree (SPT)** : In this data aggregation scheme, each source sends its information to the sink along the shortest path between the two. Where these paths overlap for different sources, they are combined to form the aggregation tree.

3. **Greedy Incremental Tree (GIT)** : In this scheme the aggregation tree is built sequentially. At the first step the tree consists of only the shortest path between the sink and the nearest source. At each step after that the next source closest to the current tree is connected to the tree.

This is by no means an exhaustive list, but is representative of some of the data aggregation tree heuristics that can be implemented.

3.3 Performance measures

In exploring the gains and tradeoffs involved in data-centric protocols, we need to specify performance measures of interest. Three are examined in some detail in this paper:

- **Energy Savings**: By aggregating the information coming from the sources, the number of transmissions is reduced, translating to a savings in energy.
- **Delay**: There is latency associated with aggregation. Data from nearer sources may have to be held back at intermediate nodes in order to combine them with data from sources that are farther away.
- **Robustness**: Somewhat related to the first measure is the fact that with data aggregation there is a decrease in the marginal energy cost of connecting additional sources to the sink. This can be considered as providing some degree of robustness to dynamics in the sensed phenomena.

3.4 Source Placement Models

The chief factors that can affect the performance of data aggregation methods are the positions of the sources in the network, the number of sources, and the communication network topology. In order to investigate these factors, we study two models of source placement, the event-radius (ER) model, and the random sources (RS) model. In both models, we generate a sensor network by scattering n sensor nodes randomly in a unit square. One of these nodes is the data sink. All nodes are assumed to be able to communicate with any other nodes that are within some distance R (the communication radius). The location of the data sources depends on the models as follows:

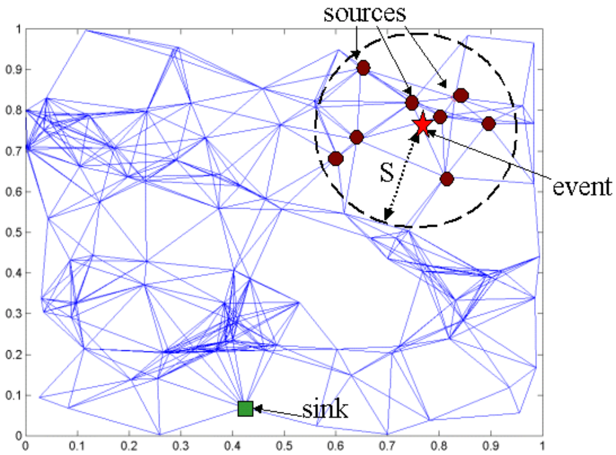


Figure 2: Illustration of the event-radius model for source positions

- **Event-Radius Model**: In this model, a single point in the unit square is defined as the location of an "event." This may correspond to a vehicle or some other phenomenon being tracked by the sensor nodes. All nodes within a distance S (called the sensing range) of this event that are not sinks are considered to be data sources. The average number of sources is approximately $\pi * S^2 * n$ (somewhat less than this if we take into account boundary effects). This model is shown in figure 2.
- **Random-Sources Model**: In this model, k of the nodes that are not sinks are randomly selected to be sources. Unlike in the

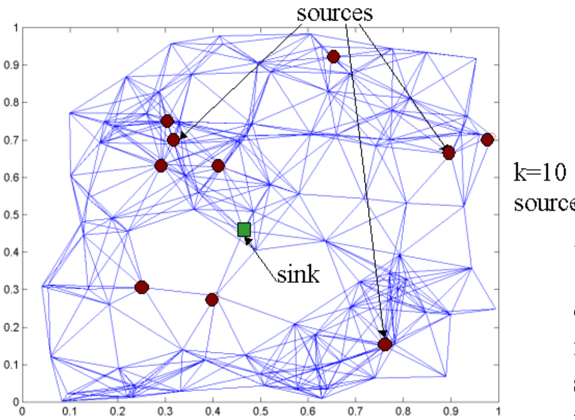


Figure 3: Illustration of the random-sources model for source positions

event-radius model, the sources are not necessarily clustered near each other. This is illustrated in figure 3.

4 Energy Savings due to Data Aggregation

4.1 Theoretical Results

We now give some analytical bounds on the energy costs and savings that can be obtained with data aggregation, based on the distances between the sources and the sink, and the inter-distances among the sources. The upshot of this section is that the greatest gains due to data aggregation are obtained when the sources are all close together and far away from the sink.

Let d_i be the distance of the shortest path from source S_i to the sink in the graph. Per datum, the total number of transmissions required for the optimal AC protocol in this case (call it N_A) is:

$$N_A = d_1 + d_2 + ...d_k = sum(d_i) \tag{1}$$

How well can the optimal DC protocol do?

Result 2: Let the number of transmissions required for the optimal DC protocol be N_D. Then $N_D \leq N_A$.

Proof: Doing data-aggregation optimally can only decrease the minimum number of edges needed compared to the situation when sources send information to the sink along shortest paths. □

Definition: The "diameter" X of a set of nodes S in a graph G is the maximum of the pairwise shortest paths between these nodes $X = max_{i,j \in S} SP(i,j)$ where $SP(i,j)$ is the shortest number of hops needed to go from node i to j in G.

Result 3: If the source nodes $S_1, S_2, \dots S_k$ have a diameter $X \geq 1$. The total number of transmissions (N_D) required for the optimal DC protocol satisfies the following bounds:

$$N_D \leq (k-1)X + min(d_i) \tag{2}$$
$$N_D \geq min(d_i) + (k-1) \tag{3}$$

Proof : (2) can be obtained by a construction - the data aggregation tree which consists of $(k-1)$ sources sending their packets to the remaining source which is nearest to the sink. This tree has no more than $(k-1)X + min(d_i)$ edges, hence the optimum tree must have no more than this. (3) is obtained by considering the smallest possible Steiner tree which would happen if the diameter were 1. In this case, the shortest path from the source node at $min(d_i)$ must be part of the minimum Steiner tree, and there is exactly one edge from each of the other source nodes to this node. □

Result 4: If the diameter $X < min(d_i)$, then $N_D < N_A$. In other words, the optimum data-centric protocol will perform better than the AC protocol.

Proof :

$$\Rightarrow N_D < (k-1)X + min(d_i) < (k)min(d_i)$$
$$\Rightarrow N_D < sum(d_i) = N_A. \quad (4)$$

□

Definition : Let us define the fractional savings, FS, obtained by using the DC protocol as opposed to the AC protocol as follows:

$$FS = (N_A - N_D)/(N_A) \quad (5)$$

FS can range from 0 (no savings) to 1 (100 percent savings). The following are the lower and upper bounds on FS, which follow directly from (2) and (3) and the above definition.

Result 5: The fractional savings FS satisfies the following bounds:

$$FS \geq 1 - ((k-1)X + min(d_i))/sum(d_i) \quad (6)$$
$$FS \leq 1 - (min(d_i) + k - 1)/sum(d_i) \quad (7)$$

To clarify the matter, assume that all the sources are at the same shortest-path distance from the sink. i.e. $min(d_i) = max(d_i) = d$.

Then we have that

$$1 - \frac{((k-1)X + d)}{kd} \leq FS \leq 1 - \frac{(d+k-1)}{(kd)} \quad (8)$$

Result 6: Assume X and k are fixed, then as d tends to infinity (i.e. as the sink is farther and farther away from the sources):

$$lim_{d\rightarrow\infty} FS = 1 - 1/k. \quad (9)$$

Proof:

In the limit, $X << d$, and $k << d$. It suffices to show that both lower and upper bounds in (8) converge to the same right hand side value:

$$lim_{d\rightarrow\infty}\left(1 - \frac{(k-1)X + d}{kd}\right) = lim_{d\rightarrow\infty}\left(1 - \frac{(k-1)X}{kd} - \frac{d}{kd}\right) = 1 - 1/k (10)$$

and

$$lim_{d\rightarrow\infty}\left(1 - \frac{(d+k-1)}{(kd)}\right) = lim_{d\rightarrow\infty}\left(1 - \frac{d}{kd} - \frac{(k-1)}{kd}\right) = 1 - 1/k \quad (11)$$

□

Essentially, what Result 6 tells us is quite intuitive. If the distance between the sink and the sources is large compared to the distance between the sources, then the optimal DC protocol gives k-fold savings. When there are 4 sources that are close together and located far-away from the sink, then the AC protocol will have about 4 times as many transmissions, i.e. there are roughly 75 % fewer transmissions with data aggregation. When there are 10 such sources, the gains are nearly 90 %, and so on...

Result 7: If the subgraph G' of the communication graph G induced by the set of source nodes $(S_1, \ldots S_k)$ is connected, the optimal data aggregation tree can be formed in polynomial time.

Proof: The proof is constructive. Start GIT. The tree is initialized with the path from the sink to the nearest source. At each additional step of the GIT, the next source to be connected to the tree is always exactly one step away (such a source is guaranteed to exist since G' is connected). At the end of the construction, the number of edges in the tree is therefore $d_{min} + (k-1)$, which is the lower bound given in relation (3). Hence the lower bound is tight and therefore optimal. The GIT construction runs in polynomial time w.r.t. the number of nodes [33]. Hence although finding the optimal data aggregation tree is NP-hard in general, in this particular situation, we have a polynomial special-case.□

Result 8: In the ER model, when $R > 2S$, the optimal data aggregation tree can be formed in polynomial time.

Proof: It is easy to see that when $R > 2S$, all sources are within one hop of each other. This is therefore a special case of result 8. Under this condition, both GIT and CNS schemes will result in the optimal data aggregation tree.□

4.2 Experimental Results

We now present our experimental results showing the energy costs of AC and DC protocols for both the ER and RS source placement models. The experimental setup is as follows: for the ER model, 5 evenly spaced values of the sensing range S from 0.1 to 0.3 are tested, while for the RS model the number of sources k is varied 1 to 15 in increments of 2. For both models the communication radius R is varied from 0.15 to 0.45 in increments of 0.05. For each combination of S or k and R 100 experiments were run. Each experiment consists of a random placement of the $n = 100$ nodes including the sink node in a square are of unit size. In some cases (particularly when the values of E or R are low) a particular experiment may result in unconnected graphs or no sources; the measurements from these cases are not taken into account while computing the averages. The error-bars shown in the plots represent the standard error in the mean.

Figures 4 and 7 show a 3D surface plot of the number of transmissions in the ideal AC protocol for both models. In both cases these costs are greatest when a) the number of sources is large and b) the communication range is low. The former is obvious as the number of transmissions should increase with sources; the latter is understandable since the number of hops between the sources and the sink (indeed between any two nodes in the network) is high when the communication radius is low.

Figure 5 compares the transmission energy costs of the various protocols as the communication range is varied, keeping the sensing range constant at 0.2 (which corresponds to about 12.5 sources on average, ignoring edge-effects). At the very bottom is the lower bound on N_D given in relation 3. In this figure it can be seen that the GITDC seems to coincide with the lower bound all throughout. This is because when S is even of moderate length, with high probability, the subgraph which lies within the circle of radius S around the event is connected, and result 7 holds. The performance of the CNSDC approaches optimal as R increases, as per result 8. The SPTDC protocol also performs well all through. In all cases there is a $50-80\%$ savings compared to the AC protocol. Figure 8 is the equivalent plot for the Random Sources Model. The first thing to note is that the lower bound is no longer tight, since the sources are placed randomly anywhere in the network and unless the network is dense (high R) the sources are unlikely to be within one hop of each other. In this setting the GITDC performs the best, fol-

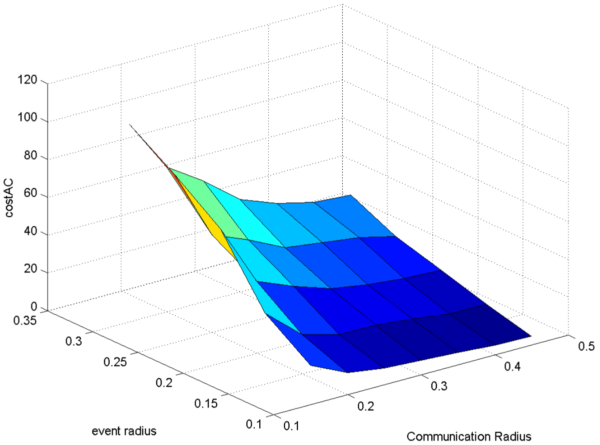


Figure 4: Cost of address-centric routing (average number of transmissions) in event-radius model

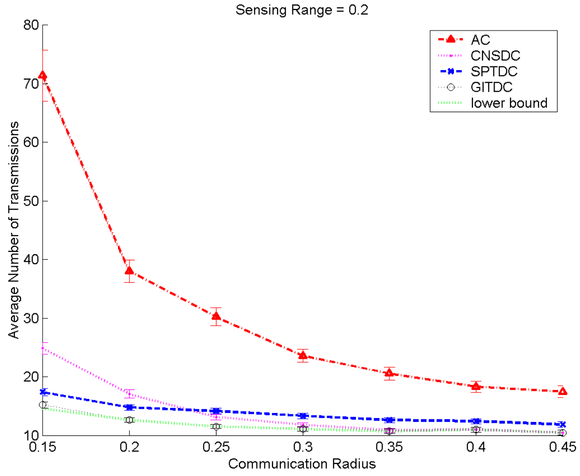


Figure 5: Comparison of energy costs versus communication radius in event-radius model

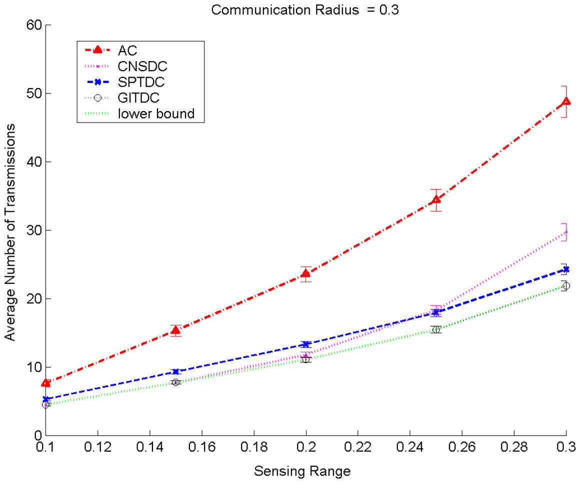


Figure 6: Comparison of energy costs versus sensing range in event-radius model

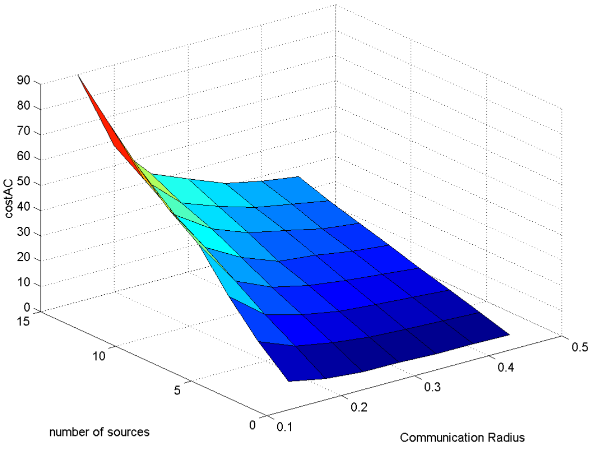


Figure 7: Cost of address-centric routing (average number of transmissions) in the random-sources model

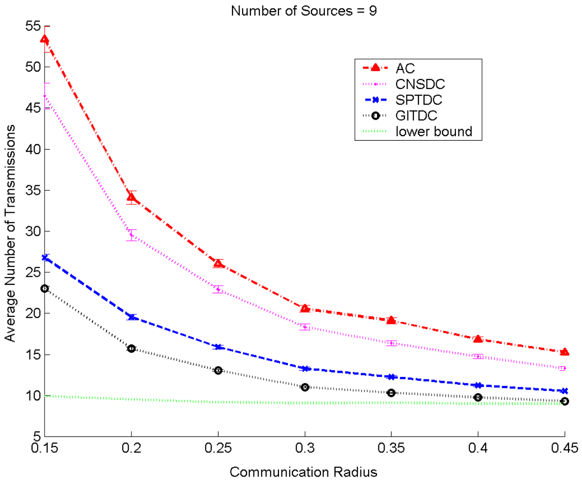


Figure 8: Comparison of energy costs versus communication radius in random-sources model

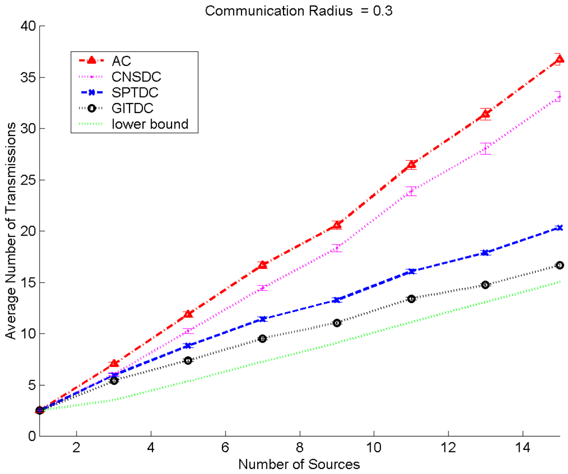


Figure 9: Comparison of energy costs versus number of sources in random-sources model

lowed by SPTDC, CNSDC and AC, respectively. CNSDC performs poorly in this setting since the sources are far apart and it doesn't pay to always aggregate at the source nearest to the sink.

Figures 6 and 9 both show that the transmission costs increase as the number of sources is increased. In the event-radius model, it can be seen that the CNSDC protocol performs poorly when the sensing range is really large. When $S = 0.3$, nearly a third of all nodes in the experiments act as sources and for many of these sources it may be faster to route directly to the sink rather than through one particular source that is closest to the sink. Figure 9 shows that the gains due to a good data aggregation technique (like GITDC) can be very significant when the number of sources is high.

To summarize, our experiments show that the energy gains due to data aggregation can be quite significant with SPTDC or GITDC particularly when there are are a lot of sources (large S or large k) that are many hops from the sink (small R).

5 Delay due to Data Aggregation

Although data aggregation results in fewer transmissions, there is a tradeoff - potentially greater delay because data from nearer sources may have to be held back at an intermediate node in order to be aggregated with data coming from sources that are farther away. This can be seen by referring back to figure 1; in figure 1b, node B which acts as the aggregating node for sources 1 and 2, is only one hop from source 1 but is two hops from source 2. Thus if both sources transmit the data simultaneously, the data from source 1 will get to B before the data from source 2 and take longer to get to the sink than it would in the no aggregation scheme shown in figure 1a. Note that this delay depends on the aggregation function - for some simple kinds of data aggregation such as duplicate suppression, there is no need for data to be withheld at an aggregating node. For more complicated forms of data aggregation, where the output aggregated packet depends on the combination of multiple input packets this delay is an issue.

It can be seen that, in the worst case, the latency due to aggregation will be proportional to the number of hops between the sink and the farthest source. When no aggregation is employed, the delay between the time when the various sources transmit data and the sink receives the first packet is proportional to the number of hops between the sink and the nearest source. Hence one way to quantify the effect of aggregation delay is to examine the difference between these two distances. This is shown in figures 10-13. The experimental setup is the same as discussed in section 4.2. The upper curve in all these figures is representative of the latency delay in DC schemes with non-trivial aggregation functions and the lower curve is representative of the latency delay in AC schemes. The difference between these curves is greatest in both models when the communication radius is low, and when the number of sources is high. In figure 13, as the number of sources increases the two curves saturate to extreme values. The upper curve saturates to a value of about 4 which is about the maximum number of hops between the sink and any node in the network. The lower curve saturates at a value close to the minimum number of hops (1).

6 Robustness due to Data Aggregation

As indicated before, with data aggregation there is a lower marginal energy cost of connecting additional sources to the sink as opposed to the AC approach. Consider the GITDC protocol, for example. At each step, the energy cost (in terms of additional edges, i.e. additional number of transmissions per datum) of connecting an additional source is simply the shortest distance of that source to the aggregation tree at the current step. In the AC protocol, however, the cost of adding an additional source is the distance of that source all the way to the sink. Figures 14 and 15 show this relationship. The x-axis represents the number of sources connected to the sink and the y-axis represents the total number of transmissions required for all sources to communicate to the sink. In both models of source placement, ER as well as RS, for a given energy budget, a greater number of sources can be connected to the sink. Thus the energy savings with aggregation can be transformed to provide a greater degree of robustness to dynamics in the sensed phenomena. For example if at any given time only a fraction of all sources can give accurate readings of the sensed phenomena, then a greater number of such readings can be processed and sent to the sink in the DC protocol even if both schemes use the same amount of total energy. The gains are, of course, smaller in the random-sources model since there is less scope for aggregation when the sources are randomly located.

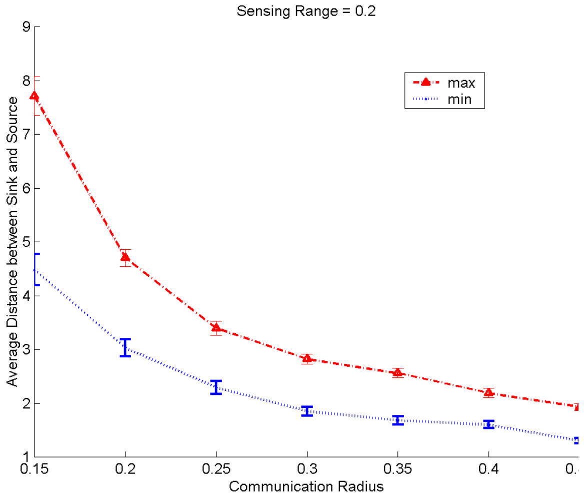


Figure 10: Distance of sink to nearest and farthest source versus communication radius in event-radius model

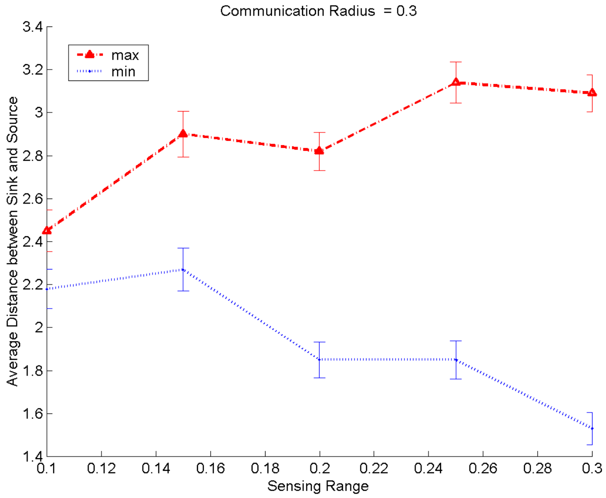


Figure 11: Distance of sink to nearest and farthest source versus sensing range in event-radius model

7 Shortcomings of the Modelling

The reader should note the simplifying assumptions we have made in the above analysis and some performance issues that have not been taken into account in our modelling. First, we have chosen to make a somewhat stark contrast between routing protocols which do and do not use data aggregation in order to highlight the effect of data aggregation on routing performance. It can be argued that our categorization of protocols into address-centric versus data-centric on this basis is overly simplistic, and has ignored the possibility of hybrid protocols which combine the advantages of both. For example, one can conceive of the sources finding routes to the sink in a traditional end-to-end manner, while the intermediate nodes are endowed with some intelligence and can perform data aggregation on messages passing through them.

Second, in discussing the energy costs of the routing protocols we have not considered any of the overhead costs involved in setting up or maintaining the routing paths from the sources to the sink. There are two reasons for this - one is that to do so we would have been forced to take into account specifics about how the routes are set up and this would affect the generality of the analysis, the second is that it is not clear that taking these costs into account would have any differential impact on AC versus DC protocols. In addition to energy savings due to data aggregation, there will also be some additional energy savings through the avoidance of globally unique IDs, which can require a significant number of bytes in the small packets typical of sensor networks. Such savings have not been modelled here.

Third, our analysis of the delay focused only

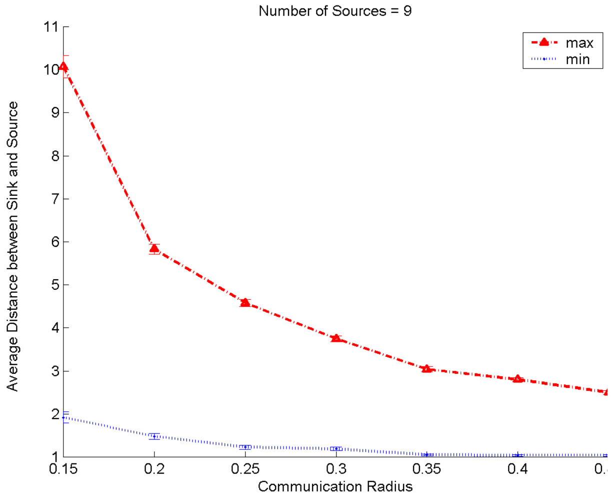


Figure 12: Distance of sink to nearest and farthest source versus communication radius in random-sources model

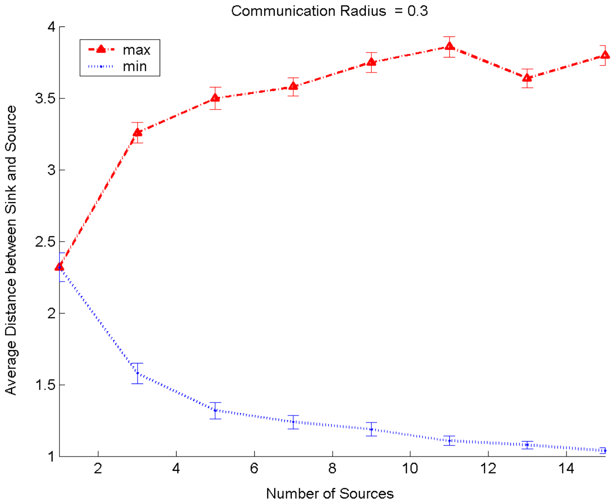


Figure 13: Distance of sink to nearest and farthest source versus number of sources in random-sources model

on the latency due to aggregation. There are two other possible sources of delay that we did not take into account - processing delay and delay due to congestion. There is likely to be an additional delay in data-centric protocols due to the processing that needs to be performed by aggregating nodes. It could be argued that this processing delay is a second order effect and unlikely to be as significant as the latency delay we analyzed. We chose not to model the delay due to congestion as this would depend on a number of additional details such as the MAC protocol used, the traffic in the network; also, again, it is not clear that congestion delay would have a differential impact on data centric versus address-centric protocols.

Finally, our analysis has focused on the case where there is a single sink. Although this is a reasonable scenario for many applications, it is reasonable to ask what would happen if there were additional sinks. One solution is to think of the different flows in that case as a superposition of many single sink data-flows. However, this would yield an over-estimate of the energy costs, as further aggregation savings can be possible if there are redundancies in the sources and the data being requested by the various sinks. This will be a topic for future study.

8 Related Work

The use of sensor networks has been envisioned in a range of settings such as industrial applications [35], vehicle tracking applications [28] and habitat monitoring [4]. A number of independent efforts have been made in recent years to develop the hardware and software architectures needed for wireless sensing. Of particular note are UC Berkeley's Smart Dust Motes [19],

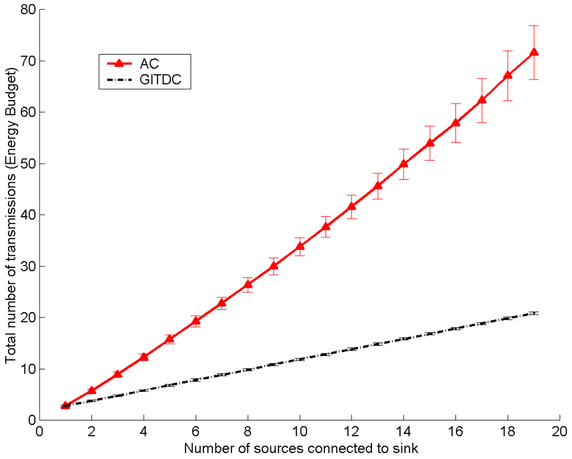


Figure 14: Average number of transmissions required to connect to a given number of sources in event-radius model

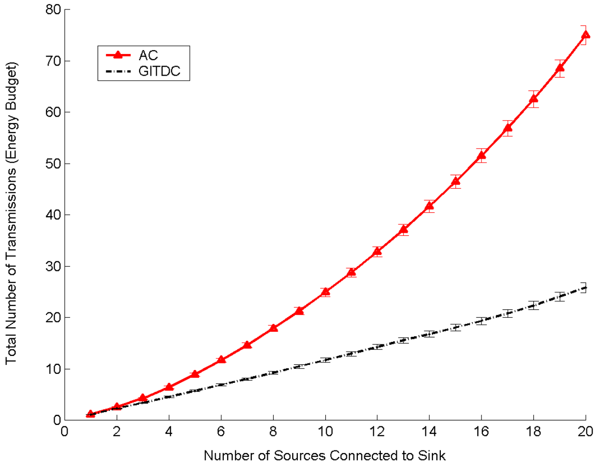


Figure 15: Average number of transmissions required for sink to receive data from a given number of sources in random-sources model

TinyOS [16], and the PicoRadio [29] project; the Wireless Integrated Network Sensors (WINS) project [28] and PC-104 based sensors [4] developed at University of California Los Angeles; and the μAMPS project at MIT [23]. The challenges and design principles involved in networking these devices are discussed in [9], [10], and [22]. Energy-efficient medium access schemes applicable for sensor networks are presented in [7], [31] and [37]. Techniques for balancing the energy load among sensors using randomized rotation of cluster heads are discussed in [15]. Some attention has also been given to developing localized self-configuration mechanisms in sensor networks [5].

The great majority of wireless routing protocols developed in recent years have been for mobile ad-hoc communication networks [27]. Depending on whether the routes are maintained at all times or if they are created afresh when needed, these are categorized into *proactive* [24], *reactive* [18], [20], [25], [26], or *hybrid* [12] protocols. Some work has also been directed to incorporating GPS-like geographical information with the routing technique [1], [21]. These approaches are all address-centric, in that they are focused on end-to-end routing between pairs of addressable nodes. There has also been work on address-centric routing protocols that conserve energy and maximize the system lifetime. In these schemes, which are applicable to sensor networks, routing metrics that incorporate energy expenditure considerations are defined for each link [6], [32].

The application-specific nature of sensor networks leads to the alternative approach we have described in this paper as data-centric. The meta-naming of data is suggested in [14] as a means to reduce transmission of redundant data for flooding-like schemes for information dissemination. *Directed diffusion* [17] is the protocol that is most like the data-centric routing models analyzed in this paper. In directed diffusion, all nodes are application-aware and communicate named data. The benefits of application specific, in-network processing and data-aggregation are quantified through experimental results in [13]. In the experimental set-up described in that paper, the form of data aggregation used (duplicate suppression) reduces the traffic by up to 42% for four sources.

The notion of in-network processing during routing is not unique to sensor networks alone. In Active Networks, intermediate routing nodes can perform customized computations on and modify the contents of messages passing through them on a per-user or per-application basis [34]. Limited router-assist techniques have also been proposed for multicast on the internet which would permit intermediate routers to look at special router-assist fields on packets in order to eliminate redundant signaling and perform subcasting [3].

Optimal data aggregation, as we have shown in this paper, requires the formation of a minimum Steiner tree, a well known NP-complete problem arising in many networking contexts [36]. The greedy incremental tree (GIT) heuristic scheme described in our paper (also known as the nearest participant first algorithm) is a well-known approximation algorithm for this problem[33]. It is known to have an approximation ratio of 2 (i.e. the tree that it outputs can have no more than 2 times as many edges as the optimum). A distributed version of this algorithm is discussed in [2]. The best known approximation algorithm for the minimum Steiner tree problem has an approximation ratio of about 1.55 [30].

Finally, we mention here in passing that there is another sense in which the phrase "data-centric networking" has been used [8]; namely to describe an approach to ubiquitous computing in which human users are identified not with static computing devices but with their personalized services and data.

9 Conclusions

We have modelled and analyzed the performance of data-centric routing in wireless sensor networks. We identified and investigated some of the factors affecting performance, such as the number of placement of sources, and the communication network topology. The formation of an optimal data aggregation tree is generally NP-hard. We presented some suboptimal data aggregation tree generation heuristics and showed the existence of polynomial special cases.

The modelling tells us that whether the sources are clustered near each other or located randomly, significant energy gains are possible with data aggregation. These gains are greatest when the number of sources is large, and when the sources are located relatively close to each other and far from the sink. We discussed how these energy gains can be translated into increased robustness to dynamics in sensed phenomena. The modelling, though, also seems to suggest that aggregation latency could be non-negligible and should be taken into consideration during the design process. Data-centric architectures such as directed diffusion should support a Type of Service (TOS) facility that would permit applications to effect desired tradeoffs between latency and energy.

Acknowledgments

The authors would like to acknowledge the help of Chalermek Intanagonwiwat.

References

[1] S. Basagni, I. Chlamtac, V. R. Syrotiuk and B. A Woodward. "A distance routing effect algorithm for mobility (DREAM)," In *ACM/IEEE International Conference on Mobile Computing and Networks (MOBICOM'98)*, pages 76-84, October 1999, Dallas, Texas.

[2] F. Bauer and A. Varma. "Distributed algorithms for multicast path setup in data networks," *in Transactions on Networking*, vol. 4, no. 2, 181-191, 1996.

[3] B. Cain, T. Speakman, D. Towsley, "Generic Router Assist (GRA) Building Block Motivation and Architecture," *RMT Working Group, Internet-Draft <draft-ietf-rmt-gra-arch-01.txt>*, work in progress, March 2000.

[4] A. Cerpa *et al.*, "Habitat monitoring: Application driver for wireless communications technology," *2001 ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean*, Costa Rica, April 2001.

[5] A. Cerpa and D. Estrin, "ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies," *unpublished.*

[6] J.-H. Chang, L. Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks," *INFOCOM*, 2000.

[7] Seong-Hwan Cho and A. Chandrakasan, "Energy Efficient Protocols for Low Duty Cycle Wireless Microsensor Networks", *ICASSP 2001*, May 2001.

[8] M. Esler, J. Hightower, T. Anderson, and G. Borriello "Next Century Challenges: Data-Centric Networking for Invisible Computing: The Portolano Project at teh University of Washington," *ACM/IEEE International Conference on Mobile Computing and Networks (MobiCom '99)*, Seattle, Washington, August 1999.

[9] D. Estrin, R. Govindan, J. Heidemann and S. Kumar, "Next Century Challenges: Scalable Coordination in Sensor Networks," *ACM/IEEE International Conference on Mobile Computing and Networks (MobiCom '99)*, Seattle, Washington, August 1999.

[10] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the World with Wireless Sensor Networks," *International Conference on Acoustics, Speech and Signal Processing (ICASSP 2001)*, Salt Lake City, Utah, May 2001.

[11] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness*, Freeman, San Francisco, 1979.

[12] Z. J. Haas, and M. R. Pearlman, "The Zone Routing Protocol (ZRP) for Ad-Hoc Networks," *IETF MANET, Internet-Draft*, December 1997.

[13] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin, and D. Ganesan, "Building Efficient Wireless Sensor Networks with Low-Level Naming," *18th ACM Symposium on Operating Systems Principles*, October 21-24, 2001.

[14] W.R. Heinzelman, J. Kulik, and H. Balakrishnan "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99)*, Seattle, Washington, August 15-20, 1999, pp. 174-185.

[15] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *33rd International Conference on System Sciences (HICSS '00)*, January 2000.

[16] J. Hill *et al.*, "System Architecture Directions for Networked Sensors," *ASPLOS*, 2000.

[17] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *ACM/IEEE International Conference on Mobile Computing and Networks (MobiCom 2000)*, August 2000, Boston, Massachusetts

[18] D.B. Johnson, and D.A. Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networking," in *Mobile Computing*, T. Imielinski and H. Korth, editors, Kluwer Academic Publishing, 1996.

[19] J. M. Kahn, R. H. Katz and K. S. J. Pister, "Mobile Networking for Smart Dust", *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 99)*, Seattle, WA, August 17-19, 1999

[20] Karp, B. and Kung, H.T., Greedy Perimeter Stateless Routing for Wireless Networks, in Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000), Boston, MA, August, 2000, pp. 243-254.

[21] Y.B. Ko and N.H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 1998, pp. 66-75.

[22] W.W. Manges, "Wireless Sensor Network Topologies," *Sensors Magazine*, vol. 17, no. 5, May 2000.

[23] Rex Min, Manish Bhardwaj, Seong-Hwan Cho, Amit Sinha, Eugene Shih, Alice Wang, and Anantha Chandrakasan, "Low-Power Wireless Sensor Networks", *VLSI Design 2000*, January 2001.

[24] S. Murthy, and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," *MONET*, vol. 1, no. 2, pp. 183-197, October 1996.

[25] V.D. Park, and M.S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *IEEE INFOCOM '97*, Kobe, Japan, 1997.

[26] C. Perkins, "Ad Hoc On-Demand Distance Vector (AODV) Routing," *IETF MANET*, Internet-Draft, December 1997.

[27] C. Perkins, *Ad Hoc Networking*, Addison-Wesley, 2000.

[28] G.J. Pottie, W.J. Kaiser, "Wireless Integrated Network Sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 551-8, May 2000.

[29] J. Rabaey, J. Ammer, J.L. da Silva Jr. and D. Patel, "PicoRadio: Ad-Hoc Wireless Networking of Ubiquitous Low-Energy Sensor/Monitor Nodes ," *Proceedings of the IEEE Computer Society Annual Workshop on VLSI (WVLSI'00)*, Orlando, Florida, April 2000.

[30] G. Robins, A. Zelikovsky, "Improved Steiner Tree Approximation in Graphs," *Proc. of ACM/SIAM Simposium on Discrete Algorithms*, pp. 770-779, 2000.

[31] S. Singh, C.S. Raghavendra, "Power efficient MAC protocol for multihop radio networks," *Proceedings PIMRC'98*, Sept. 1998.

[32] S. Singh, M. Woo, C.S. Raghavendra, "Power-aware routing in mobile ad-hoc networks," *ACM/IEEE International Conference on Mobile Computing and Networks (MOBICOM'98)*, pages 76-84, October 1999, Dallas, Texas.

[33] H. Takahashi and A. Matsuyama, "An approximate solution for the steiner problem in Graphs," *Math. Japonica*, vol. 24, no. 6, pp. 573-577, 1980.

[34] D.L. Tennenhouse, J.M. Smith, W.D. Sincoskie, D.J. Wetherall, and G.J. Minden, "A survey of active network research," *IEEE Communications Magazine*, vol. 35, no. 1, pp. 80-86, January 1997.

[35] J. Warrior, “Smart Sensor Networks of the Future,” *Sensors Magazine*, March 1997.

[36] P. Winter, “Steiner Problem in Networks: A survey,” *Networks*, vol. 17, no. 2, pp. 129-167, 1987.

[37] A. Woo and D.E. Culler, “A Transmission Control Scheme for Media Access in Sensor Networks,” *ACM/IEEE International Conference on Mobile Computing and Networks (MobiCom ’01)*, July 2001, Rome, Italy.