

Low Power Network Discovery Algorithms

Abhijit Davare

David Nguyen

Mentor: Farinaz Koushanfar

Background

An ad-hoc sensor network is a network of numerous wireless nodes where each node has sensing, storage, computation, communication and possibly actuating resources. Each node has a limited power source that determines the lifetime of the node. Therefore, energy is the most critical factor in sensor networks in all of its design levels (i.e. hardware, network layers and applications). Numerous studies in sensor networks have already demonstrated that communication between nodes is the dominating source of energy consumption. While traditional wireless network architecture has been based on systems of static base stations, it appears that a multi-hop network, where each node communicates with a few close nodes, is most efficient in terms of energy savings and bandwidth reuse. In multi-hop networks, each node communicates with other geographically distant nodes using intermediate nodes to build a communication path. The goal of this research introduces and evaluates new power saving techniques for wireless ad-hoc sensor networks.

In order to present the flow of the algorithms within the network, we introduce the notion of a token. *Token(s)* defines a node that is currently active and is executing the algorithm. The procedures involve electing the next active node and giving the control of the algorithms (token) to that node. The token node can only send control to the nodes within its communication range (neighboring node). The algorithms for both of our project goals can utilize single or multiple tokens.

Project Goal #1

We address the problem of finding an efficient and localized information-gathering algorithm that minimizes the total number of radio communications during the task of gathering complete information from all of the nodes within a network or a specified part/subset of a network. The assumption is that each node has all the information about itself (including its own geographic coordinates and its data) and all of the information from the nodes within its communication range.

By information, we mean a type of meta-data that can be different for different applications. In the simplest case, this information is just the status (active/standby) of all the nodes within the desired area. In order to find the information contained in a node, either the node or one of its neighbors should be visited. This type of data gathering is NP-complete. Since the data gathering procedure is localized and the network is ad-hoc, at each active node the important part of the procedure is to decide where next to send the token. We provide an efficient, yet complete low-power heuristic for data gathering. By complete, we mean that we have, provably, all the required information at the termination of the algorithm. To evaluate the quality of the network discovery algorithm, we also develop a method for creating networks with known optimal solutions.

Project Goal #2

Since many nodes in the network can be idle, we can save power by turning the radios of these nodes off and putting them in standby mode. In the second part of the project, we describe

and evaluate a localized technique for reducing the energy consumption of the network by putting the maximum number of nodes in standby mode for the maximum amount of time without affecting the connectivity of the network, {Xu01,Che01}. Each node determines its status (active/standby) based on its available energy and the connectivity of the network within its neighborhood. We have developed necessary and sufficient conditions for a procedure to check locally the connectivity of the network. This same mechanism forms the basis for our efficient power management technique. Our power management technique finds the provably minimum power state of the nodes within a local neighborhood without affecting the connectivity of the nodes, while a subset of the nodes stay active and communicating.

Approach

The information gathering procedure is built on the observation that different network shapes dictate different traversal strategies. The procedure starts by exploring the shape of the network, using a geometric perimeter routing (outer face routing) technique {Bos99}. If the network is non-planar, the Gabriel Graph planarization technique is applied to enable correct perimeter routing. The nodes on the perimeter are assigned a rank of zero, while all of the other nodes in the network rank themselves according to their minimum distance from the perimeter nodes. The next active node is selected based on its rank, number of neighbors whose status is unknown, number of second level neighbors whose status is unknown and the area constraints. The area constraint is the component of the algorithm that ensures all nodes of an area are visited before leaving that area. Otherwise, the procedure will encounter the additional energy overhead of visiting redundant nodes to return to unvisited nodes from an already visited area. We illustrate the importance of articulation points in this area-based approach. Articulation points partition the graph into a number of sub-graphs. The procedure uses the bounds of each of these sub-graphs as the area constraints. We illustrate how tuning the different parameters of the procedure can yield a more stable method for traversing the graph. Also, we study the performance of the algorithm for different shapes of the network. We evaluate the performance of the algorithm against instances with a known optimal solution to study the efficiency of this approach.

We calculate the total power consumption in terms of the number of transmissions and receptions in the network. If the energy cost of a transmission is A and the energy cost of a reception is B , the total energy cost is then: $A \cdot (\text{Number of transmissions}) + B \cdot (\text{Number of receptions})$.

To evaluate accurately the performance of low power sensor network discovery algorithms, it is useful to have an optimal solution to compare against. The optimal solution consists of a network and a path through that network that gathers information from all of the nodes while visiting the fewest number of nodes. Finding the optimal path for a given network is NP-Complete, even with global information. In order to develop benchmarks, we develop a method to generate networks, keeping track of the optimal solution during generation. The procedure is to first initialize the algorithm by placing a node at a random point in the area. Then perform the following two steps iteratively:

1. Place a unique neighbor, node A , in the range of the previous node, node B , but not in the range of any nodes prior to node B (if any).
2. Place a new node, node C , in the range of node B , but not in the range of any other nodes.

This procedure continues until the desired number of nodes has been placed on the optimal path. The optimal path consists of the initial node and all of the “C” nodes in the iteration. After the optimal path has been generated, other “filler” nodes may be added to the network as long as they are in range of the path nodes. Since any number of these “filler” nodes may be added once the optimal path has been chosen, the required network density can always be obtained.

In developing a method to determine which nodes go to sleep, it is useful to know the amount of area uniquely covered by a node. Since we are interested in a complete network discovery algorithm, we do not want nodes that uniquely cover some area to be put into sleep mode. The exact approach to determine the amount of unique area covered by a node can be solved by geometric formulations. We have found, however, that this can be very computationally intensive, and may not be well suited for constrained sensor network nodes. The alternative we have chosen is a grid-based approach. To accomplish this, we overlay a grid onto the section of interest. With this, we can discretize the coverage of nodes into grid cells. In our implementation, a matrix is associated with the area of interest. Entries in the matrix indicate the nodes that cover each of the cells. Finding nodes that cover unique area is then approximately equivalent to finding matrix entries that are covered by only one node. This observation may be helpful in future research, but we have not pursued this avenue due to scope and time limitations.

The approach we used for standby mode assignment is based on the notion of local connectivity. For every node with a token, we consider the node and its neighbors as candidates for standby mode. A node is put into standby mode if it is the one whose neighbors can stay connected for the longest amount of time given that this node is put to sleep. The actual sleep time for this node is dependant on the energy of nodes in the local neighborhood.

Experimental Results

In this section, we present experimental results for the evaluation of our localized power saving strategies. The simulation environment consists of more than 3,000 lines of code in C++. There are a number of parameters involved in this study, for instance, how long a node stays asleep, density of the network, and the number of tokens in the network. To analyze the affect of these parameters on the lifetime performance of the network, we apply our algorithms to randomly generated networks over a subset of the domain.

For the network discovery algorithm, we obtained results for the expected average performance of our algorithm on various networks. We modified both the number of nodes in the network and the shape of the network. The shape of the network was represented as the ratio between the width of the network (a) and the height of the network (b). The number of nodes in the network was varied between 200 and 600, and the a/b ratio was varied between 1 and 0.6. The results are represented in Figure 1. The average performance of our algorithm is the expected number of nodes that our algorithm will visit based on the number of sections, the number of perimeter nodes, and the number of nodes with even and odd rank. As the network size is increased, the percentage of nodes visited in the average case will decrease. As the network shape becomes elongated, the average-case performance will decrease.

For the second part of our experiment, we study the lifetime of the network and measure how long the network remains operational. The network ceases operation when it is partitioned into several disconnected sections. We show the impact of our method as the percentage increase in the lifetime of the network without the sleeping strategy. Previous research in this area, such

as SPAN [6] and GAF [13], use different metrics for measuring the lifetime of the network. Specifically, these approaches use the fraction of nodes with non-zero energy. We did not adopt these metrics, since we believe that the network becomes nonfunctional only when the network is partitioned into a certain number of partitions. The network is generated by randomly placing nodes in a 1×1 square. The communication range of the nodes is specified by the user. Since changing the ranges and the number of nodes in a fixed area can generate scaled versions of the same density, we fixed the ranges to 0.13 units in our results. We modify the density of networks only by altering the number of nodes. We assume a traffic model that incorporates a continuous bit rate (CBR) traffic generator that spreads traffic among the nodes. The sleeping time T_s is a parameter in our algorithm that is expressed as a function of the amount of time the neighboring nodes will be able to communicate with each other. The number of tokens is another parameter that is defined as a certain percentage of the number of nodes in the network. For the energy consumption model, we adopted numbers from the study in [9]. The power consumption numbers according to this model are: transmission (1.9W), reception (1.5W), listening (0.75W) and sleep (0.025W). Initially, every node has energy of 500 J.

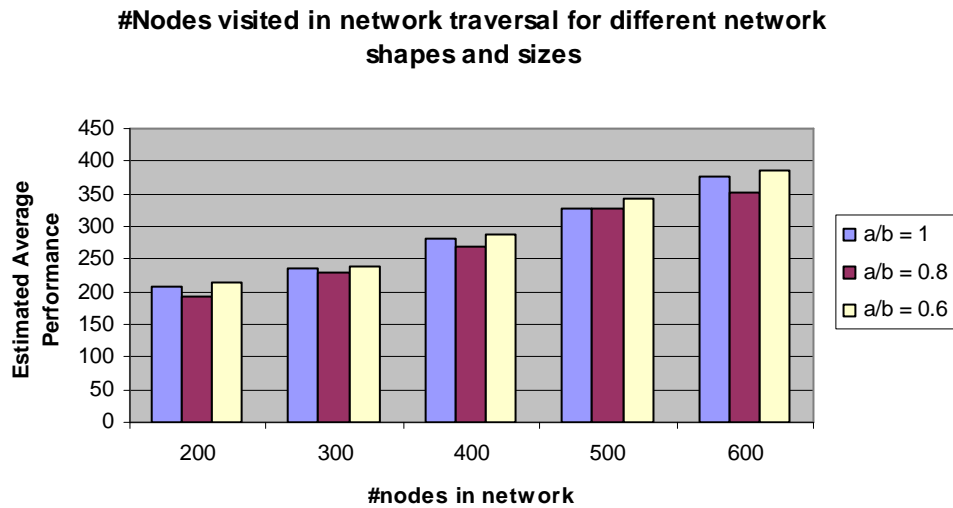


Figure 1 – Relative performance of discovery heuristic increases with network size

Table 1 illustrates the average performance of our algorithm over a range of different node densities and different number of tokens and sleeping times. The first column shows the number of nodes, the second column shows the number of tokens used by the algorithm, the third column shows the sleeping time and the last column shows the lifetime of the network. Although the lifetime increases with increasing network density, there is a wide range of variation over the different parameters involved. To more carefully analyze these relationships and dependencies, we have used 3-D plots of the lifetime of the algorithms based on the three parameters involved, i.e. number of tokens, sleeping time and the lifetime of the network. Figures 2, 3 and 4 illustrate the relationship between these parameters and network lifetime. Although there is a trend in the graph, it is not monotone and an efficient system design study requires adaptively tuning the parameters with respect to the other parameters.

Our approaches for network discovery and standby mode assignment use local information, and thus cannot utilize global network information. Clearly, if we can consider all nodes in the network for the purposes of selecting a node for standby assignment or network discovery, we can obtain a higher quality solution. However, the power requirements of this tradeoff would need to be analyzed further. A difficulty we encountered with the standby mode assignment was comparing its quality against an optimal sleeping strategy. A limitation of our analysis is that it is based on a simplified model of communication and traffic that does not explicitly take into account factors such as channel degradation, mobility, and random failure.

# of Nodes	% of tokens	T_s (%)	lifetime (%)
100	20	20	19.7
200	20	30	28.95
400	20	40	30.12
100	30	20	16.96
200	30	30	80.43
400	30	40	91.45
100	40	20	13.57
200	40	30	70.25
400	40	40	48.32

Table 1: Experimental results, showing the relationship between number of nodes, number of tokens, sleeping time (T_s) and the lifetime of the network.

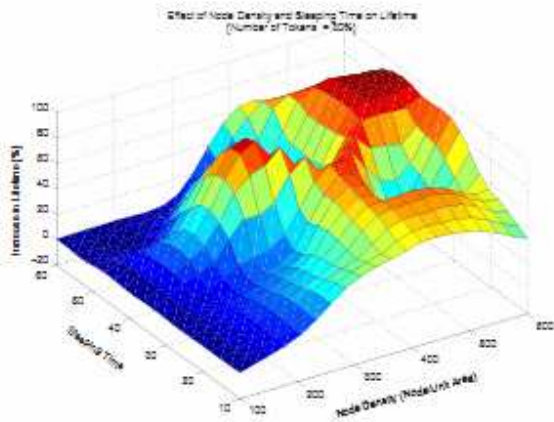


Figure 2 – The effect of node density and sleeping time on the lifetime of the network. The number of tokens is set to 20%.

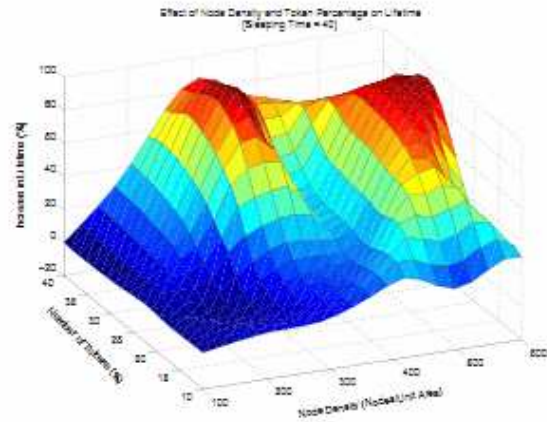


Figure 3 – The effect of node density and number of tokens on the lifetime of the network. The Sleeping time is set to 40.

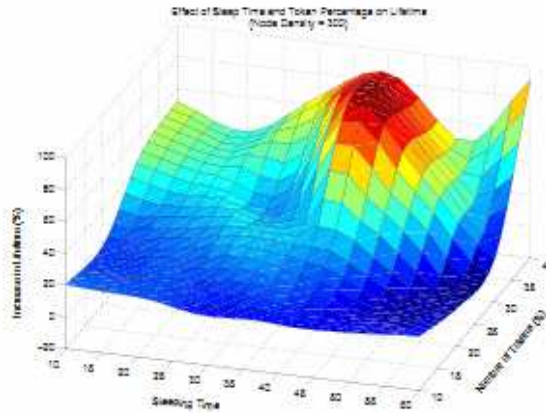


Figure 4 – The effect of sleeping time and number of tokens on the lifetime of the network. The node density is set to 300 nodes/unit area.

Conclusion

We have developed a distributed localized novel approach for power minimization in distributed wireless systems. We have developed an algorithm for complete network discovery which takes into account both geometric and graph-theoretical concepts. We demonstrate the effectiveness of this approach by evaluating the expected average performance of this algorithm on a number of network densities and shapes. As part of the second goal of this project, we develop algorithms to put nodes into standby mode to increase network lifetime. By leveraging the necessary and sufficient conditions that nodes can be placed in standby mode without adversely affecting network connectivity, we have developed algorithms and synchronization protocols that efficiently exploit the trade-off between the localization and effectiveness of optimization techniques. The effectiveness of the approach is demonstrated using simulations over a variety of parameter settings in the domain space.

References

- [Bos99] P. Bose, P. Morin, I. Stojmenovic, J. Urrutia, “Routing with guaranteed delivery in ad hoc wireless networks”, Proc. of 3rd ACM Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications DIAL M99} (1999) pp. 48-55.
- [Che01] B. Chen, K. Jamieson, H. Balakrishnan, R. Morris, “Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks”, Proceedings of the SIGMOBILE (MOBICOM)} (2001) pp. 85-96.
- [Xu01] Y. Xu, J. Heidemann, D. Estrin, “Geography-informed energy conservation for ad-hoc routing”, Proceedings of the SIGMOBILE (MOBICOM)} (2001).