A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks

Fan Ye, Alvin Chen, Songwu Lu, Lixia Zhang
UCLA Computer Science Department, Los Angeles, CA 90095-1596

Abstract

Wireless sensor networks offer a wide range of challenges to networking research, including unconstrained network scale, limited computing, memory and energy resources, and wireless channel errors. In this paper, we study the problem of delivering messages from any sensor to an interested client user along the minimum-cost path in a large sensor network. We propose a new cost field based approach to minimum cost forwarding. In the design, we present a novel backoff-based cost field setup algorithm that finds the optimal costs of all nodes to the sink with one single message overhead at each node. Once the field is established, the message, carrying dynamic cost information, flows along the minimum cost path in the cost field. Each intermediate node forwards the message only if it finds itself to be on the optimal path, based on dynamic cost states. Our design does not require an intermediate node to maintain explicit "forwarding path" states. It requires a few simple operations and scales to any network size. We show the correctness and effectiveness of the design by both simulations and analysis.

1 Introduction

The state-of-the-art hardware and communication technologies have made it feasible to manufacture a large number of inexpensive and simple sensors with wireless networking capabilities in a cost-effective fashion. In many emerging application scenarios (e.g., battlefield surveillance, large-area and perimeter monitoring in agriculture, and autonomous ocean scientific sampling), a large number of such simple immobile nodes are deployed in a vast geographical area to monitor activities or environmental conditions [14, 1]. When a sensor detects activities or unusual behaviors, it generates report messages and delivers them to the interested user. The user thus receives its interested information from multiple sensing sources.

In this paper, we study the problem of minimum cost data delivery from any given source to the interested client (called a sink) in a large sensor network. The source can be any sensor node in the network and there may be an unbounded number of intermediate nodes between the source and the client. Our design has been driven by the following three goals:

- **Optimality**: Most data forwarding protocols are designed based on a chosen optimality criterion. Our design seeks to achieve minimum cost forwarding. Some popular cost criteria include hop count, energy consumption, and delay etc.
- **Simplicity**: The cost-effectiveness factor of a sensor network dictates that nodes have limited computing capability and memory resources. We seek to minimize the number of operations performed and the states maintained at each sensor node that participates in data forwarding. In particular, we do not maintain any explicit "forwarding path" states. We do not even need an ID for an intermediate node.
- **Scalability**: Since unconstrained scale is an inherent feature of a sensor network, the solution has to scale to large network size.

This work explores a new design paradigm to a scalable solution to minimum cost forwarding – a *cost field* based approach. In this approach, we first set up a cost field starting from the given sink node, and its value at each intermediate node denotes the minimum cost to reach the sink from that node. In principle, the cost field concept is similar to the natural gravity field that drives water flowing from a high post to a low post. Once the cost field is established, any sensor can deliver the data to the sink along the minimum cost path. To this end, each message carries the minimum required cost from the source to the sink, and its consumed cost starting from the source so far to current intermediate node. As the message travels from the source to the sink, each intermediate node decides to forward the message only if the consumed cost plus the cost at this node (i.e., the minimum cost from this node to the sink) is equal to the source's cost. This way, we can achieve minimum-cost path forwarding without maintaining explicit path information (in terms of which nodes are next-hop nodes along the path) at any intermediate node.

Our design meets all the three goals identified above. Some additional features are: (1) No forwarding path states are needed; each node only needs to maintain the minimum cost from this node to the sink; (2) Once the cost field is set up, any sensor can deliver the data to the sink. This is important if the user has interests in observations from multiple sensors – the typical case in a sensor network. (3) From the forwarding perspective, each intermediate sensor does not need an ID or an address.

Two main contributions of this work are: (1) We propose a generic cost-field based approach to minimum cost forwarding;

0-7803-7128-3/01/$10.00 (C) 2001

(2) We devise a novel backoff-based cost field setup algorithm, which establishes the optimal cost field in a single pass. The effectiveness of our design has been confirmed by both simulations and analysis.

The rest of the paper is organized as follows. Section 2 discusses related work and additional design issues. Section 3 presents our field-based forwarding approach, together with the backoff-based field establishment protocol. Section 4 evaluates the proposed protocol through simulations, and Section 5 concludes the paper.

2 Related Work

Wireless sensor networks have spurred much interest in the networking research community. A number of proposals have addressed various aspects in sensor network design [1, 5, 7, 13, 14]. [1] proposes the directed diffusion approach to forwarding in sensor networks. [5, 13] study energy-efficient protocol design. [7] proposes a self-organizing protocol design approach for sensor networking protocols. [14] describes efforts in building real sensors. Compared with these related works, our design addresses the problem of finding optimal costs for sensor nodes to an interested user with very low message overhead. The algorithm does not have path states in nodes. Intermediate nodes in the network do not need to have IDs.

Routing has been a very active research area in the context of ad hoc networks in recent years, many proposals have appeared in the literature [2, 4, 6, 8]. However, the scale of ad hoc networks is typically much smaller compared with sensor networks, and these proposals typically assume a much smaller network size (e.g., DSR assumes a network size of 6-8 hops [4]). These proposals also maintain path states, and require addresses for nodes. Hierarchical proposals such as landmark [3] may scale with a large number of nodes, but it requires hierarchical address space. Our design provides a new approach to scalable forwarding without partitioning the network and organizing it hierarchically.

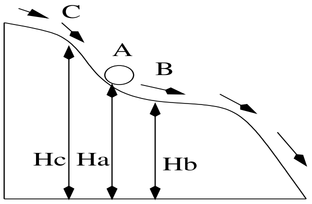
Several proposals [9, 11, 12] have also studied the issue of energy efficient design in wireless networks. However, these protocols do not address the unique issues of unconstrained network scale of sensor networks.

3 Field Based Optimal Forwarding

This section presents the design details of our field based minimum cost forwarding scheme. We first introduce the cost field concept in Section 2.1, then describe how to use the cost field to realize minimum cost forwarding in Section 2.2. In Section 2.3, we provide a novel backoff based field setup algorithm with analytical properties.

3.1 Cost Field Concept

The cost field based design is inspired by a common, natural phenomenon (shown in Figure 1): mountain water flows down into a valley, and the altitude at each point serves as the guide to direct the water from a high post to a low post, and eventually to the valley's bottom, which has the lowest altitude. Similarly, we set up a *cost field*.



Water at A on the slope goes only
to B, since B's altitude is less than
that of A's, while Hc>Ha, so A will not go
to C.

Figure 1: Cost Field Example

At each node, the *cost field* is defined as the minimum cost from that node to the sink on the optimal path. The field has only one state at each node – the minimum cost to the sink. It is the only state kept by each intermediate node. The link cost can take any common form, such as hop count, consumed energy, or delay.

3.2 Minimum-Cost Path Forwarding

Once the cost field is established, messages may flow to the sink along the minimum cost path. To eliminate path states, when a message is sent out by a source, it carries the minimum cost from the source to the sink. This message also carries the total cost that it has consumed so far starting from the source to the current intermediate node. Then a sender simply broadcasts the message in the wireless channel without targeting any specific neighbor (this is why no IDs are needed for its neighbors). A neighboring node hearing the message decides to forward the message only if the sum of the consumed cost (carried in the message header) and the cost at this node matches the source's cost (also in the message header). Hence, it achieves minimum-cost path forwarding without maintaining explicit path information (in terms of which nodes are next-hop nodes along the path) at any intermediate node.

We illustrate the above procedure through an example (Figure 2). Consider the source seeks to deliver a report message REP to the sink along the minimum cost path. Assume that the minimum costs OL_B, OL_C, OL_{source} from B, C and the source to the sink are $90, 100, 200$, respectively. The message carries a total cost budget state 200 initially at the source. This is the minimum cost required to reach the sink from the source. When a node broadcasts the message, it includes in the message how much budget has been consumed so far from the source to itself. Suppose when A broadcasts the message, the total amount of consumed cost from the source to A is 110 (including $A's$ broadcast cost). After B and C hear the message, both first make sure they are "closer" to the sink than sender A by comparing their costs with A's cost (included in A's broadcast). A node with a greater cost will drop the message. Suppose that both B and C pass the comparison, they then calculate the

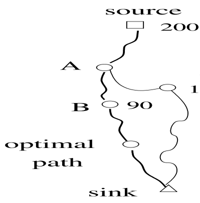


Figure 2: Forwarding along the minimum-energy path

remaining cost budget as $200 - 110 = 90$. Now B will forward REP since $90 = OL_B$. However, C will not forward REP since the remaining budget is not sufficient to reach the sink, i.e., $90 < OL_C$, which means it is not on the optimal path of the source.

This way, we can effectively enable only the nodes along the minimum cost path to forward REP without keeping any path states, because a node knows if it is on the optimal path by calculation. Furthermore, there is no need for any IDs since a sender simply broadcasts the message in its one-hop neighborhood, and each receiver decides whether to further forward this message or not.

3.3 A Backoff-Based Cost Field Establishment Algorithm

Now we present our back off algorithm to set up the cost field.

3.3.1 A naive solution: flooding

A straightforward solution to set up the cost field would be through flooding. Initially, each node sets its cost to ∞. After the sink broadcasts an ADV (advertisement) message containing its own cost (0 initially), the message propagates throughout the network. Upon hearing an ADV message from node M, node N has a path with cost $L_M + C_{N,M}$, where L_M is node M's cost, and $C_{N,M}$ is the cost from N to M. Node N then compares its current cost L_N and $L_M + C_{N,M}$. If the new cost is smaller, it sets L_N to $L_M + C_{N,M}$ and broadcasts an ADV message with its new cost. Whenever a node receives an ADV message leading to a smaller cost, it resets its cost and broadcasts a new ADV message. Eventually, every node may calculate the optimal cost to the sink through flooding.

The flooding based design suffers excessive advertisement messages. A node may receive many advertisement messages consecutively, and each of which leads to a smaller cost. Thus a node could advertise many times. For example, in a network of 150×150 m^2 area with 1500 nodes and a transmission range of 10 meters, the total number of advertisement messages can go as high as 77365 (the average number of broadcasts for each node is about 51.5), as we observed from simulations. Moreover, each advertisement at a node induces further updates and advertisements for next hops. The farther a node is to the sink, the more advertisement messages it rebroadcasts. In the same example, although nodes close to the sink rebroadcast only a few times, nodes "far" from the sink advertise over 150 times! Clearly, flooding can easily stress the network and is not scalable.

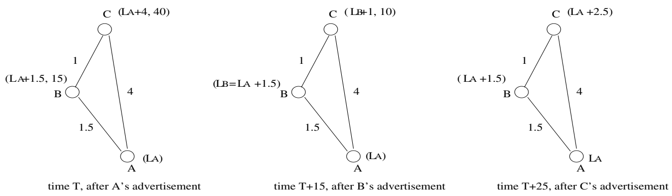


Figure 3: An illustrative example for backoff-based optimal cost field establishment

3.3.2 Main idea of our algorithm

The reason that a node broadcasts more than once is that it broadcasts immediately after obtaining a lower cost, no matter whether the cost is optimal or not. If we can defer the broadcast at the node to the time after it has heard the message leading to the minimum cost, the node may broadcast only once, carrying its optimal cost. Thus how long the node defers its broadcast becomes critical. Our back off algorithm sets the total deferral time to be proportional to the optimal cost at a node.

To illustrate the idea, we use the example of Figure 3 to explain how our backoff-based approach works; the link cost is shown in the figure.

- At time T, node A broadcasts an ADV message and the message is heard by neighbors B and C. Assume the minimum cost at A is L_A. The cost for B and C are ∞. After B receives the ADV from A, B sets its cost as $L_A + 1.5$ where 1.5 is the link cost between A and B, and B sets a backoff timer that expires after $\gamma \cdot 1.5 = 15$ where γ is a constant and we let $\gamma = 10$. Similarly, C sets its cost as $L_A + 4$ and sets a backoff timer $\gamma \cdot 4 = 40$. We notice that if flooding were used, both B and C will broadcast immediately since they have got some costs less than ∞.

- At $T + 15$, B's backoff timer expires, B *finalizes* its minimum cost as $L_B = L_A + 1.5$, and broadcasts an ADV message containing L_B. When C hears it, since $L_C = L_A + 4 > L_B + 1 = L_A + 2.5$, C updates its cost to $L_B + 1$, and *resets* its backoff timer to be $\gamma \cdot 1 = 10$. Notice that the previously set timer has not expired by this time. Had flooding been used, C would advertise a second message at this time. When A receives ADV from B, it *discards* it since $L_B > L_A$.

- Finally, at $T + 25$, C's timer expires, and C finalizes its cost as $L_C = L_B + 1 = L_A + 2.5$, and broadcasts a message with its minimum cost.

We observe from this example that each node broadcasts once and only once with the optimal cost and suppresses other non-optimal advertisement messages. The intuition behind is that each node backs off proportionally to its minimum cost. We will formally prove this result in Section 2.3.4.

```
Event: node N receives an ADV message
from node M
if($L_N > L_M + C_{N,M}$) {
$L_N = L_M + C_{N,M}$
reset timer to expire after $\gamma \cdot C_{N,M}$
}

Event: N's backoff timer expires
broadcast an ADV message containing $L_N$

Variables:
$L_N$, $L_M$: costs of node N, M to reach the sink
$C_{N,M}$: cost between node N and M
$\gamma$: backoff timer coefficient
```

3.3.3 Algorithm details

The following pseudo code specifies the operations at node N upon hearing an ADV message. It should be noted that a node may still hear multiple ADV messages, each of which may induce a lower cost. But instead of broadcasting immediately, the node only updates its cost and waits until its backoff timer expires. By the time a node broadcasts, the above algorithm ensures that it has obtained the minimum cost.

3.3.4 Analytical properties

Our proposed algorithm has the following properties:

(a) Each node only propagates optimal information to its neighbors, and discards all redundant or non-optimal messages.

(b) The approach establishes the minimum cost field with only one message broadcast at each node.

The following propositions formalize the properties.

Proposition 3.1 *Given a sink, and an additive path cost function J, the backoff-based algorithm always sets up the optimal cost field with one message (containing optimal cost) broadcast at each node.*

Proof We prove our results by induction.

Let us consider all the nodes $N_0, N_1, N_2, ..., N_k$ on N_k's optimal cost path, where N_0 is the sink. Denote the minimum cost for them as 0, OL_1, OL_2, ..., OL_k, respectively. Without any loss of generality, we assume $0 \leq OL_1 \leq OL_2 \leq ... \leq OL_k$, and $OL_i = OL_{i-1} + C_{N_{i-1},N_i}$, where $1 \leq i \leq k$. We want to prove that each of these nodes broadcasts only once with its minimum cost, and in ascending order of minimum costs.

Define the total backoff time T_i for node N_i as the time from sink N_0 broadcasting an ADV message to N_i broadcasting its ADV message, where $0 \leq i \leq k$. We need to prove $0 \leq T_1 \leq T_2 \leq ... \leq T_k$. We use induction for proof. It is easy to see that $0 \leq T_1$ since $T_1 = \gamma \cdot C_{N_1,N_0} \geq 0$. Thus N_1 broadcasts with its minimum cost OL_1. N_1 broadcasts only once since messages from any other node would lead to higher cost than OL_1.

Suppose we have proved for node N_j. Now If $T_{j+1} < T_j$, node N_{j+1} must have heard an ADV message from another node M_i before N_j broadcasts its ADV message. From the pseudo code, we can see that

$$T_{j+1} = \gamma \cdot C_{N_{j+1},M_i} + T_{M_i},$$

where T_{M_i} is the total back off time of M_i. Apply the above equations recursively, we can have

$$T_{j+1} = \gamma \cdot (C_{N_{j+1},M_i} + C_{M_i,M_{i-1}} + ... + C_{M_1,N_0})$$

where each of M_i, $M_{i-1}, ..., M_1, N_0$ gets its cost from the previous node. Similarly,

$$T_j = \gamma \cdot (C_{N_0,N_1} + C_{N_1,N_2} + ... + C_{N_{j-1},N_j})$$

Since $T_{j+1} < T_j$,

$$\gamma \cdot (C_{N_{j+1},M_i} + C_{M_i,M_{i-1}} + ... + C_{M_1,N_0}) < \gamma \cdot (C_{N_0,N_1} + C_{N_1,N_2} + ... + C_{N_{j-1},N_j}),$$

then node N_{j+1} has another path $N_{j+1}, M_i, M_{i-1}, ..., M_1, N_0$ with an even smaller cost than the optimal one $OL_j + C_{N_j,N_{j+1}}$. This is contradictory to our assumption. So $T_{j+1} \geq T_j$.

Note that N_{j+1} cannot have a $L_{j+1} < OL_{j+1}$ after it broadcasts the ADV message. Otherwise, using similar arguments, we can show that it must have a path with even less cost than its optimal path, which violates the assumption. Therefore, N_{j+1} broadcasts once and only once, and with its minimum cost. ☐

Note in the above proof, we do not impose any constraint on the form of the cost. It can be hop number, energy, etc. Hop number cost is easy to apply. When energy is used as the cost, the sender includes its broadcasting power in the ADV message. When a receiver hears the message, it can infer the minimum energy needed to reach the sender by measuring the signal strength and employ certain signal attenuation model.

3.3.5 Practical issues

In the above design, we have ignored several practical issues, and we address them here.

Nonnegligible delays If the transmission, propagation and processing delays are non-zero, these factors may alter the ordered broadcasting along the optimal path $\{N_0, N_1, N_2, ..., N_k\}$. However, if we set γ large enough, the impact could be minimal. In Section 4, we will show through simulations that if γ is set to be the same order of magnitude as the delay factors, few nodes broadcast more than once, and the algorithm still finds the minimum costs for all nodes.

Channel errors If ADV messages are lost due to channel errors, the above algorithm may not establish the minimum cost field in one pass. But among all the successfully delivered advertisements, it can still find the best possible cost field. Letting the sink broadcast more rounds can improve the field. We will see in the simulation section that the impact of channel error is quite limited.

Node failures Node failures before the field setup phase is “bypassed” as if there were less nodes in the network. The cost values are optimal among all remaining alive nodes. Node failures after the cost field setup phase can make the cost field not optimal. This issue can be addressed by refreshing the cost field, either in a time-driven (i.e., periodic) or event-driven (i.e., delivery quality at the sink changes dramatically) manner. To overcome occasional node failures without refreshing the whole cost field, we could slightly increase the cost budget at the source to beyond OL_{source}. Thus the message can go along multiple paths and is not subject to node failures along a single path. Further investigations can be found in a techical report [].

4 Performance Evaluation

4.1 Metrics and Simulation Settings

Our simulator is written in PARSEC [10], because of its capability to handle a large number of nodes efficiently. We use a simulation setting of 1500 randomly-scattered sensors in a $150 \times 150\,\mathrm{m}^2$ area. The origin point (0,0) is at the upper left corner, the sink is at (1350, 1350) – the lower right corner. Each node has a transmission range of 10 meters. We use the energy as cost in the simulations. The minimum energy needed to reach another node (that is d($d \leq 10$) meters away) is d^2 units, i.e., signals attenuate inversely proportional to the square of distance.

We use three metrics to evaluate the algorithm: the number of advertisement messages; the field set up time, which is the time from sink broadcasting the ADV message to the last node broadcasting its ADV message; and the cost a node obtains as channel error increases.

4.2 Message Overhead in the Cost Field Setup

Optimal cost To verify the correctness of the algorithm, we compare the cost values obtained by our algorithm with the one obtained by flooding. We run one simulation for each on the same network topology. There is a $10ms$ delay at the nodes, and we set the backoff time coefficient in the algorithm γ to $10ms$. We then compare the minimum costs of the corresponding same node in two cases. It turns out that for all nodes, the minimum cost obtained using our backoff algorithm is exactly the same as the one using flooding. This demonstrates the correctness of our algorithm.

Overhead Figure 4 shows the number of advertisement messages a node broadcasts for both our design and flooding. Nodes are ordered in increasing minimum cost (i.e., from nodes “closer” to sink to “farther” ones). From the figure, we observe that node costs vary from 0 to around 800 units. In our backoff algorithm, no matter how far a node is to the sink, its broadcasts never exceed three times. In fact, only 42 nodes broadcast twice, and 2 nodes broadcast three times, and the other 97% nodes broadcast exactly once. In contrast, In the flooding approach, the farther a node is to the sink, the more messages it broadcasts. Some nodes even broadcast over 140 times! The average number of broadcasts for each node is about 50. Therefore, our algorithm finds the same optimal cost for all nodes while requiring only one broadcast for nearly all nodes if we set γ properly.

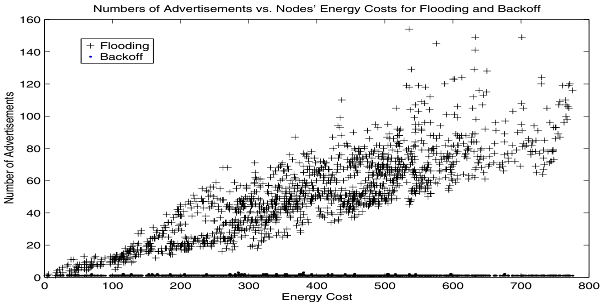


Figure 4: advertisement numbers of flooding and backoff

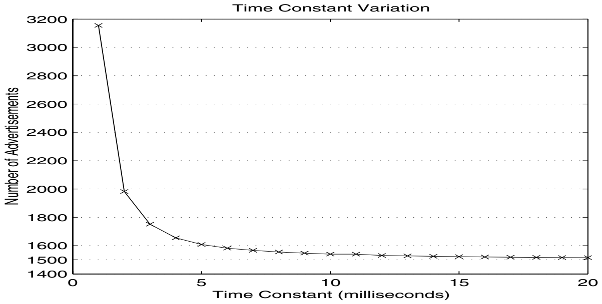


Figure 5: total advertisement number vs. backoff timer coefficient

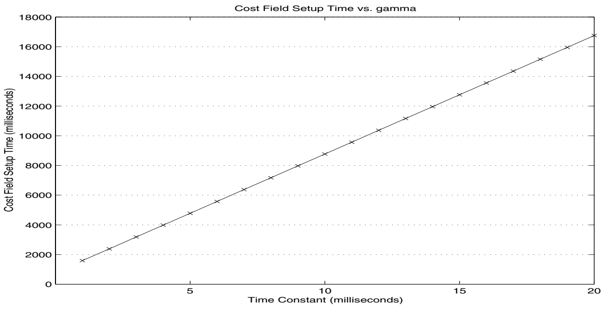


Figure 6: cost field setup time vs. backoff timer coefficient

4.3 Impact of Backoff Timer Coefficient

We now investigate how γ value affects advertisement message overhead and field setup time. When γ is not large enough, the accumulative processing, transmission, propagation delay factors along a path could alter the ordered broadcasts of nodes. Then a node may broadcast more than once.

In Figure 5, γ varies from 1 to 20ms at 1ms steps. It can be seen that if γ is low (say 1ms) compared to the 10ms delay,

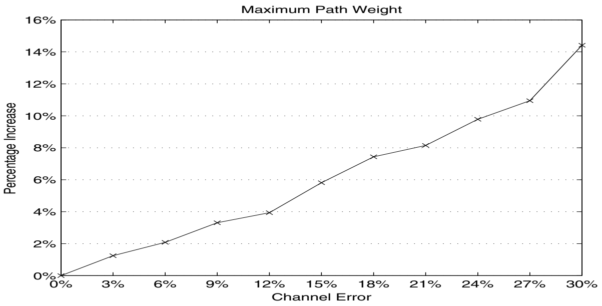


Figure 7: maximum cost increase

the total number of advertisements is over 3000. This implies that each node advertises twice on average. But note that it is already an order of magnitude lower than the flooding case, which requires each node advertise 50 times on average. As γ increases, the number drops sharply and at $\gamma = 10$, only a few nodes have more than one advertisements. Further increase in γ almost eliminates multiple broadcasts for every node.

Besides, we can observe that the field setup time is proportional to γ. Large γ values lead to linear increasing field setup time (Figure 6).

4.4 Impact of Channel Error

We gauge the impact of channel error on our algorithm by tracking the average increase in the field's largest cost (i.e., the node "farthest" to the sink) in Figure 7. γ is still 10ms, which is the same as the delay factors at a node.

For an error-free environment, our algorithm generates the same optimal cost value. As error rates increase, the maximum cost increases slightly. A moderate 10% error rate leads to less than 4% in cost increase. Even for extremely high error rate of 30%, the increase is less than 15%. This increase remains gradual and nearly linear for most of the tested error rates. Therefore, we can see that, the impact of channel error on the cost is quite limited.

5 Conclusions

In this paper, we study the problem of how to deliver messages from any source to an interested client user along the minimum-cost path in a large sensor network with an unconstrained number of nodes. We achieve optimal path forwarding by taking the cost field based approach. To this end, we devise a novel backoff-based cost field setup algorithm that finds the optimal costs of all nodes to the sink with only one message at each node. Once the field is established, the message, carrying dynamic cost information, flows along the minimum cost path in the cost field. Each intermediate node forwards the message only if it finds itself on the optimal path for this message based on the message's cost states. Our design does not require an intermediate node to maintain explicit "forwarding path" states. Our approach requires constant time and space complexities at each node, and scales to large network size. The correctness and effectiveness of our proposal have been confirmed by both simulations and analysis.

References

[1] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *ACM MOBICOM'00*, 2000.

[2] J. Broch, D. A. Maltz, D. Johnson, Y. Hu, J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *MOBICOM'98*, 1998.

[3] P.F. Tsuchiya, "The Landmark Hierarchy: A New Hierarchy for Routing in Very Large Networks," ACM Computer Communications Review, vol. 18, no. 4, pp.35-42,1988

[4] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-hoc Wireless Networks," *Mobile Computing*, Kluwer Academic Publishers, 1996.

[5] J. Kulik, W. Rabiner, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," *ACM MOBICOM'99*, 1999.

[6] C. Perkins, "Ad-Hoc On Demand Distance Vector Routing (AODV)," Internet-Draft, November 1997.

[7] K. Sohrabi and G. Pottie, "Performance of A Novel Self-Organization Protocol For Wireless Ad-Hoc Sensor Networks", *IEEE Vehicular Technology Conference*, 1999.

[8] G. Pei, M. Gerla and X. Hong, "LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility," *MobiHOC'00,* 2000.

[9] R. Kravets and P. Krishnan, "Power Management Techniques for Mobile Communication," *MOBICOM'98*, 1998.

[10] PARSEC simulation language, *http://pcl.cs.ucla.edu/projects/parsec/* .

[11] J. E. Wieselthier, G. Nguyen, A. Ephremides, "On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks," *IEEE INFOCOM'00*, 2000.

[12] J. Chang L. Tassiulas, "Energy Conserving Routing in Wireless Ad-Hoc Networks," *IEEE INFOCOM'00*, 2000.

[13] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, "Energy-Efficient Communication Protocol for Wirelesss Mircrosensor Networks," *International Conference on System Sciences*, 2000.

[14] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister, "System Architecture Directions for Networked Sensors," *Architectural Support for Programming Languages and Operating Systems*, 2000.