

# Chain-Based Protocols for Data Broadcasting and Gathering in the Sensor Networks \*

Kemei Du, Jie Wu and Dan Zhou

Department of Computer Science and Engineering

Florida Atlantic University

Boca Raton, FL 33431

{kdu, jie, dan}@cse.fau.edu

## Abstract

In wireless sensor networks, the sensor nodes gather information and send the information to a base station periodically. Some important messages need to be broadcasted to all nodes. Data gathering and broadcasting are important operations that consume significant amounts of battery power. Due to the limited battery life, energy efficiency is becoming a major challenging problem in these power-constrained networks. Chain-based protocols construct a transmission chain connecting all nodes to save energy dissipation of data transmission. In this paper, we first review several existing chain-based protocols. We then present our multiple-chain scheme which outperforms the existing ones in the sparse-node distribution case. Furthermore, we develop an energy-efficient chain construction algorithm which uses a sequence of insertions to add the least amount of energy consumption to the whole chain. It consumes less transmission power compared to the closest neighbor algorithm.

## 1 Introduction

Recent rapid advances in technology have made it possible to integrate microsensor, low-power signal processing, computation and low-cost wireless communication into a sensing device, such as WINS project at UCLA [11]. Networking inexpensive sensors to construct wireless sensor networks will revolutionize information gathering and processing in many situations. For example, large scale and robust sensors could be deployed in inhospitable physical environments to collect interesting information, such as temperature, pressure and noise levels. Networked sensors can track critical events via coordination in security systems. Hundreds to thousands of sensors can be embedded into

complex bridges to perform continuous monitoring of structural metamorphoses at a fraction of the cost of conventional wireline sensor systems. Wireless sensor networks improve sensing accuracy and are robust to the failure of individual sensors.

Sensor networks are related to wireless ad-hoc networks, which are self-organizing systems consisting of hosts that do not rely on the presence of any fixed network infrastructure. After deployment, the wireless sensors self-configure to gather acoustic, magnetic, or seismic information and send the information to a control center or a base station, where end-users can retrieve the data. Important messages (e.g., an intruder detected by a sensor) can be broadcasted to all nodes.

Sensor networks are characterized by sheer numbers of nodes, data-centric applications, constrained resources (bandwidth and energy) and dynamic configuration (e.g., node failure due to battery depletion). There are many technical challenges associated with sensor networks, such as self-organizing algorithms, energy-efficient routing protocols, data analysis/mining technology and network lifetime improvements [1, 6, 10, 12, 14]. The source of energy for a wireless sensor is most often an attached battery. The power in sensor nodes can be used up simply by computations and transmissions. Furthermore it is infeasible to replace thousands of nodes in hostile or remote regions. Therefore, conserving energy so as to prolong the network lifetime is becoming one of the key challenges for such power-constrained networks. Recent research has addressed this topic. Power-aware channel access, routing and broadcasting in ad-hoc networks are presented in [2, 8, 13]. Scalable coordination and data gathering in sensor networks are considered in [3, 4, 5].

In this paper, we concentrate on energy-efficient all-to-all broadcasting in sensor networks. We only consider sensor networks with stationary nodes. Transmission distance is a key factor of energy consumption for data communi-

\*The work was supported in part by a grant from Motorola Inc. and NSF grants CCR 9900646 and ANI 0073736.

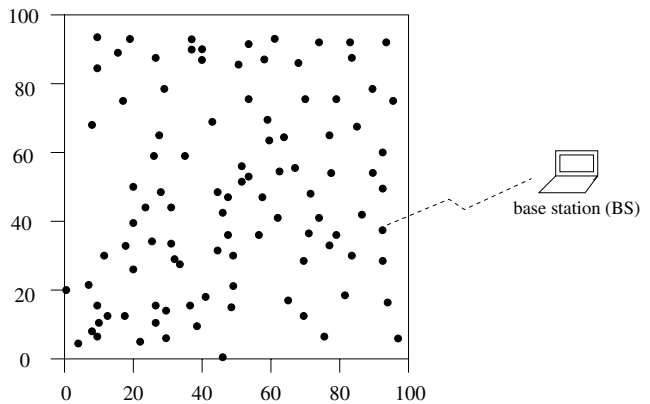
cation. Therefore our goal is to design effective protocols in which the sensor nodes are organized to transfer data energy-efficiently. Chain-based protocols reduce the distance of total data transmission, hence decrease the energy dissipation per round.

Lindsey and Raghavendra presented several chain-based protocols for sensor networks [9]. They investigated broadcast problems in sensor networks and adopted a chain-based approach for situation awareness systems, where networked sensors track critical events via coordination [8]. They proposed a linear-chain scheme for all-to-all broadcasting and data gathering. They also proposed a binary-combining scheme for data gathering which divides each communication round into levels in order to balance the energy dissipation and delay cost in sensor networks. For broadcasting, the linear-chain scheme starts data transmission with a packet at the beginning of a chain. Each node along the chain attaches its own data to this packet. Eventually, information of the whole network reaches the end of the chain. The same procedure runs in the reverse direction to complete all-to-all broadcasting. The linear-chain scheme can also be applied to gather data in sensor networks. To gather data, each node senses and transfers information along the chain to reach one particular node which will send data to a remote base station. Such a scheme is named as PEGASIS (Power-Efficient Gathering in Sensor Information Systems).

In this paper, we propose a multiple-chain scheme which divides the whole network into regions based on a center node. An independent linear chain ending at this particular center node is generated in each region. We apply the linear-chain scheme to each subchain to transfer data to the center node. The center node can either combine all the collected data or relay the separated data packets. The subchain is constructed through a sequence of insertions. Our multiple-chain scheme avoids the situation where two nodes located far away from each other become neighbors in the chain constructed for the whole network which is a problem that presented in the liner-chain scheme and the binary-chain scheme. Therefore, our scheme outperforms the existing chain-based protocols in sparse networks.

The main goal for chain-based protocols is to construct an energy-efficient chain. In this paper, we will compare our new chain-construction algorithm with that of Lindsey and Raghavendra. Their algorithm constructs a chain where each node communicates only with the closest neighbor [8]. Ours, proposed in this paper, constructs chains through *insertions*. One node is inserted at each round, which adds the least amount of energy consumption to the whole chain. Although ours is more complex, it consumes less energy and leads sensor networks to a longer life.

The remaining paper is organized as following: In Sec-



**Figure 1. A 100-node sensor network with a base station.**

tion 2, we introduce some preliminaries. We then discuss chain-based protocols and chain construction algorithms respectively in Sections 3 and 4. Section 5 presents and evaluates simulation results. Finally, in Section 6 we conclude the paper and present some ideas for future work.

## 2 Preliminaries

In this paper, we treat sensor networks as wireless ad-hoc networks with static nodes. We use the model presented by Heinzelman et al. [4]. A sensor network contains a large scale of sensor nodes that are immobile and energy constrained. Each node can adjust the area of coverage with its transmission using transmission power control. All live nodes sense data. Among them some nodes also need to relay data to a base station (BS). End users can access data through the BS. The BS is fixed and located far from the sensors. BS has sufficient power supply, therefore, it has no energy constraints. A 100-node sensor network is shown in Figure 1.

The source of power consumption can be classified into three types with regard to operations: *sensing* related, *communication* related and *computation* related [7, 12]. In a wireless sensor network, communication is the major consumer of energy. In this paper, our energy-efficient protocols concentrate on conserving communication power.

We use the first order radio model [4] to compute energy dissipation for communication. The unit energy consumption for transmitter or receiver electronics is defined as  $E_{elec} = 50$  nJ/bit, which a radio dissipates to run the transmitter or receiver circuits. The unit energy consumption for the transmit amplifier is defined as  $\epsilon_{amp} = 100$

pJ/bit/m<sup>2</sup>. The energy dissipation for sending data consists of two parts: transmitter electronics and transmitter amplifier. It depends on the data size and the transmission distance. The energy dissipation for receiving data only concerns the receiver electronics and data size. Thus, to transmit one-bit message at distance  $d$ , the radio consumes  $E_{elec} + \epsilon_{amp}d^n = 50 + 0.1d^n$  nJ and, to receive this message, the radio consumes  $E_{elec} = 50$  nJ, where  $n$  is a constant depending on the environment situation. For free space environment,  $n$  equals to 2, and for noisy urban environment,  $n$  equals to 4.

Based on the analysis of energy consumption, we need to minimize the transmission distance and data size to conserve energy. Depending on applications, different methods to reduce data size are applied. In all-to-all broadcasting, each node maintains the size of the header; it only concatenates its data to the data from other nodes. Because in broadcasting all the message are critical, no data fusion is allowed in this operation. Data gathering sensor networks collect useful data. The sensor nodes perform local data fusion as opposed to exchanging raw data over the air. Therefore, all the data packets in data gathering are the same size.

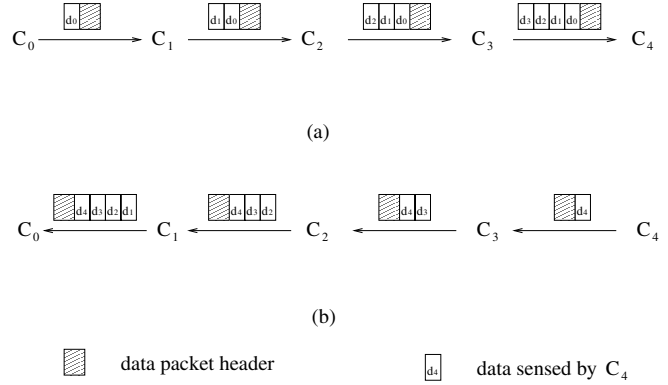
### 3 Chain-Based Protocols

Chain-based protocols construct a transmission chain connecting all nodes to save energy dissipation in each round of data collection. Several chain-based protocols are presented for sensor networks by Lindsey et al. [8, 9].

#### 3.1 Linear-Chain Scheme

The linear-chain scheme was proposed in [8] and applied to broadcasting in wireless sensor networks with urban model ( $d^4$  transmission energy where  $d$  is the distance). A linear chain connects all the sensor nodes within the network. Data is transmitted from one end of the chain to the other end. Each node attaches its own data to the received data to form a larger packet and sends it to the next node. When data finally reaches the end of the chain, the same procedure starts from that end back to the beginning node. To conserve energy, each node maintains the same size of the header. Thus all nodes transmit the same number of bits except for the end nodes, because the end nodes only transmit their own data. It is useful to distribute the energy usage among nodes to prevent nodes from being unwisely overused. Using this scheme, nodes die randomly. When a neighbor dies, a node simply skips the dead node and transmits data to the next living neighbor in the chain.

One example of linear-chain scheme is demonstrated in Figure 2. If each packet consists of a 40-byte header and 20-byte data, node  $C_0$  sends total 60 bytes (header +  $d_0$ ) to node  $C_1$ . After receiving this packet, node  $C_1$  keeps the



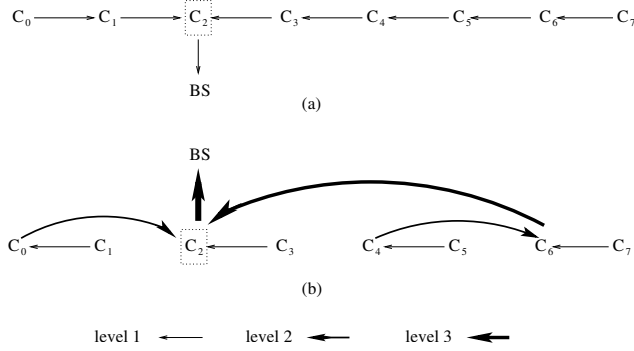
**Figure 2. Data transmission in the linear-chain scheme: (a) pass one. (b) pass two.**

same size of the header, attaches its sensing data ( $d_1$ ), and forwards 80 bytes to node  $C_2$ . The process continues until the message reaches the other end of the chain, which is  $C_4$  in this case. Once node  $C_4$  receives data, it has information from all other nodes. Now node  $C_4$  only sends its own 60 bytes packet (header +  $d_4$ ) to node  $C_3$ . Then the same procedure continues along this reverse direction until data transmission ends at node  $C_0$ . Eventually every node has the knowledge of the whole situation. In the broadcasting process, end nodes  $C_0$  and  $C_4$  send 60 bytes. Other nodes,  $C_1$ ,  $C_2$  and  $C_3$ , send 200 bytes each.

PEGASIS is another linear-chain scheme used for gathering data in sensor networks [9]. Information is sensed, fused and transferred by every node along the chain to reach one certain node (called *LEADER*) which will send data to the remote BS. Due to the data fusion, each packet has the same size. One example is shown in Figure 3(a), where  $C_2$  is the *LEADER*. Node  $C_0$  creates a packet, for example, 2000 bits, and transmits it to node  $C_1$ . After  $C_1$  receives this packet, it analyzes and compresses all the data including what it sensed to create another 2000-bit packet and sends it to node  $C_2$ , which is *LEADER*. The same procedure occurs from node  $C_7$  to  $C_6$ , then to node  $C_5$  until *LEADER*  $C_2$  collects all the information in the network. Then node  $C_2$  fuses all the information into one 2000-bit packet and transfers it to the BS to complete one round of data collection. Because transmission from a sensor to BS is expensive, each node acts as *LEADER* in turn to balance the workload among all the nodes. Thus next round, some node other than  $C_2$  will be the new *LEADER*.

#### 3.2 Binary-Combining Scheme

The delay cost from data sensing to data transmission to BS is high in PEGASIS. To balance the energy and delay

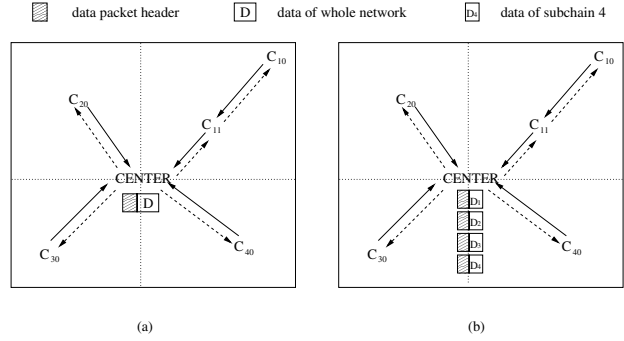


**Figure 3. (a) Data gathering in a PEGASIS. (b) Data gathering in a binary-combining scheme.**

cost, Lindsey et al. introduced  $Energy \times Delay$  metric to evaluate sensor network protocols [9]. They proposed a new chain-based approach for data gathering: binary combining scheme using CDMA. It performs parallel communication in about  $\log n$  units of delay ( $n$  is the number of nodes in the network) and spends a little more energy than PEGASIS. Chain-based binary approach divides each communication round into  $\log n$  levels. Each node transmits data to a close neighbor in a given level. Only those nodes that receive data can rise to the next level. Finally, one node, *LEADER*, sends data to BS. By then one transmission round completes. Similar to PEGASIS, each intermediate node performs data aggregation. Each round has only one *LEADER*, and live node takes turn to be *LEADER*. The scheme distributes workload evenly throughout the network. The binary-combining scheme for the same example as in Figure 3(a) is shown in Figure 3(b).

### 3.3 Multiple-Chain Scheme

Decreasing energy dissipation will prolong the lifetime of a network. Distance is a major factor in energy dissipation. We propose a multiple-chain scheme to decrease the total transmission distance for all-to-all broadcasting. Our main idea is to divide the whole sensing area into four regions centered at the node that is closest to the center of the sensing area. We construct a linear subchain in each region. The maximum distance between two neighboring nodes in a subchain is smaller than that in a single linear chain generated within the whole area. For example, we divide an  $l \times l$  area into four equal regions. The maximum neighboring distance in a subchain is  $\frac{\sqrt{2}}{2}l$ . However, it doubles to  $\sqrt{2}l$  in a single chain. Thus our scheme will minimize the total neighbor distance, especially for sparse node distribution. Node *CENTER* does become a bottleneck as it



**Figure 4. Multiple-chain scheme. (a) collecting approach. (b) relaying approach.**

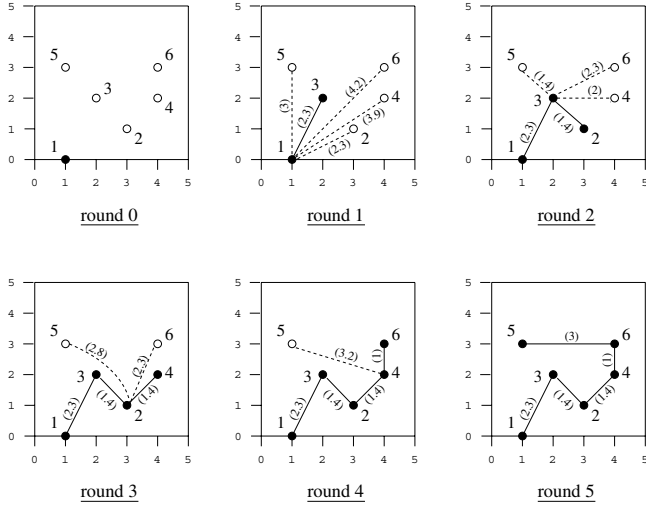
has to handle more receivings and transmissions than other nodes.

There exist two approaches to transfer data through the whole network, *Collecting* and *Relaying*.

- *Collecting* approach (see Figure 4(a)): Each subchain independently transfers and concatenates data from the end of chain to *CENTER*. After *CENTER* collects all data, it produces a new packet which contains information of the whole network. It then sends the packet to all four neighbors, which will relay the packet to the end of all subchains. However, every node except *CENTER* receives redundant data and, hence consumes more power than necessary.
- *Relaying* approach (see Figure 4(b)): The first part is the same as in *Collecting*. Each subchain independently transmits and accumulates data packet to *CENTER*. Then *CENTER* relays all received packets to all subchains. To avoid the redundancy, the neighbors of *CENTER* check each receiving packet. If a packet is the same as what a node just sent to *CENTER*, then it will discard the packet. If a packet is the first new packet the node receives, it will send this packet combined with its own sensing data along the chain. Each downstream node will perform the similar combination, create and send a larger packet to the next node, until transmission terminates at the end of subchain. If the packet does not belong to the proceeding two cases, the neighbor node will only relay this packet. No other data will be concatenated to the packet.

## 4 Chain-Construction Algorithms

One factor in conserving energy is to utilize chains efficiently, which we discussed in section 3. Another factor in



**Figure 5. Chain construction procedure for Algorithm 1. Legend for edges is  $d$ , where  $d$  is distance between corresponding nodes.**

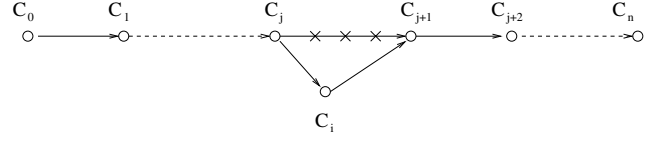
energy conservation is to construct a chain that minimizes the distance between the neighboring nodes. In this section we will first review the closest neighbor algorithm proposed by Lindsey et al. [8, 9] and then propose our chain-construction algorithm: minimum total energy algorithm.

#### 4.1 Algorithm 1: Closest Neighbor Algorithm

Energy-efficient chain construction is the main challenge in chain-based protocols. A greedy algorithm is described in [8, 9]. The operation starts with one node, the farthest node from BS. This node can also be other nodes, depending on the requirements of protocols. This node works as the head of the chain. The following procedure is broken into *rounds*, where in each round one non-chain node, the one that is closest to the chain head, is selected and appended to the chain as a new head. This procedure repeats until all nodes are in the chain. In this algorithm, the chain is constructed as a linked list, *CHAIN*.  $N$  is the set of nodes, and *HEAD* is the head node in the chain. Distance from node  $i$  to node  $j$  is denoted as  $d(i, j)$ .

**Algorithm 1:**

1. *HEAD*  $\leftarrow$  the farthest node from BS;  $N' \leftarrow N - \{HEAD\}$ ;  
  *CHAIN*  $\leftarrow \{HEAD\}$
2. **while** ( $N' \neq \emptyset$ )
3.    $key[i] \leftarrow d(HEAD, i)$ , for all  $i \in N'$
4.    $HEAD \leftarrow \text{EXTRACT-MIN}(N')$   
      //select a node with the minimum key
5.   Append(*CHAIN*, *HEAD*)  
      //append *HEAD* at the end of *CHAIN*,



**Figure 6. Algorithm 2. Node  $i$  is inserted between  $C_j$  and  $C_{j+1}$ .**

//as the new chain head

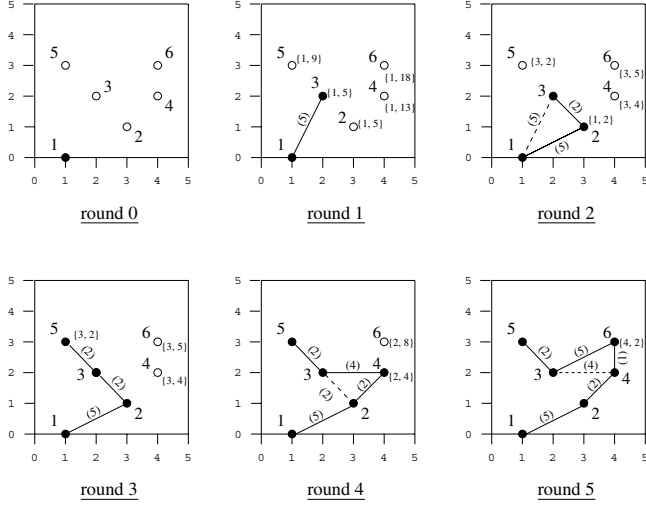
For broadcasting, this algorithm only needs slight modification. We can randomly pick one node as *HEAD* of the chain at the first step. A simple example is demonstrated in Figure 5. There exists a small sensor network with 6 hosts randomly distributed in a  $5 \times 5$  region, and BS locates at (10,10). At round 0 the farthest node from BS, node 1, is selected as *HEAD*. At round 1 the node that is closest to node 1 should be added. In this example node 3 is tied with node 2. So randomly we pick node 3 as the new chain *HEAD*. In the following rounds, node 2, 4, 6, 5 are appended to the chain in order. Finally, a chain 1-3-2-4-6-5 is constructed.

The complexity of Algorithm 1 is  $O(n^2)$ . This algorithm only ensures that the node will communicate with the closest neighbor. It cannot guarantee that the total transmission energy is minimum.

#### 4.2 Algorithm 2: Minimum Total Energy Algorithm

When a packet travels along a chain, the total energy dissipation of the network contains two parts: the transmission and receiving energy consumption at each node. Based on the first order radio model, the receiving cost only depends on packet size. The total energy used by all nodes in receiving a packet is  $(n - 1) \times E_{elec} \times S_d$ , where  $S_d$  is the packet size. The energy consumption of receiving is independent of the type of the chain used. The transmission energy depends on the distance between two nodes along a chain ( $\propto d^n$ ). To simplify the analysis we only consider the free space environment ( $n = 2$ ) here. The analysis for the urban model is similar. To conserve total energy consumption we need to build a chain with minimum  $\sum d^n$ , where  $d$  is the distance between two nodes along the chain.

The first stage in Algorithm 2 is the same as that in Algorithm 1, which is to find the farthest node from BS as one end of the chain. Then each round selects a *NEW* node which is not in the chain. The selection criteria is that the  $\sum d^2$  of the current chain with this *NEW* node increases to the minimum possible extent compared to the old chain.



**Figure 7. Chain construction procedure for Algorithm 2. Legend for nodes is  $(C_j, E')$ , where  $C_j$  is the node after which the current node should be inserted, and  $E'$  is the incremental energy after insertion which is based on  $d^2$ . Legend for edges is  $d^2$ , where  $d^2$  is the square of distance between nodes.**

There exist two cases depending on insertion position  $j$ . In this algorithm,  $C_j$  represents the node at location  $j$  in the chain. If  $C_j$  is already the end of the current chain, we simply append  $NEW$  to the chain, same as Algorithm 1. Otherwise we insert  $NEW$  between  $C_j$  and  $C_{j+1}$ . An insertion case is shown in Figure 6. The new transmission path from  $C_j$  to  $C_{j+1}$  through node  $C_i$  increases minimum energy cost to the chain compared to if other nodes or other insertion locations were selected.

**Algorithm 2:**

1.  $HEAD \leftarrow$  the farthest node from BS;  $N' \leftarrow N - \{HEAD\}$ ;  
    $CHAIN \leftarrow \{HEAD\}$
2. **while** ( $N' \neq \emptyset$ )
3.     **for each**  $i \in N'$
4.         **do**  $key[i, C_j] \leftarrow$   
            $\min\{d^2(C_j, i) + d^2(i, C_{j+1}) - d^2(C_j, C_{j+1})\},$   
           for all  $C_j$  in the  $CHAIN$
5.      $NEW \leftarrow \text{EXTRACT-MIN}(N')$   
           //select a node with the minimum key
6.      $\text{INSERT}(CHAIN, C_j, NEW)$   
           //insert  $NEW$  between  $C_j$  and  $C_{j+1}$   
           //in the  $CHAIN$

In Figure 7, we demonstrate the same example as in Figure 5 using Algorithm 2. Round 0 and round 1 are similar to that in Figure 5. Now the current chain starts at node 1 and connects to node 3. At round 2 the minimum incremental

**Table 1. Binary scheme for sensor networks with 50 nodes (R: Random, A1: Algorithm 1).**

Sensing Area	Base Station	First Node		Last Node	
		R	A1	R	A1
(10,10)	(5,110)	2006	2087	3634	3719
(20,20)	(10,120)	1724	1970	3319	3511
(50,50)	(25,150)	894	1421	2421	3040
(100,100)	(50,200)	329	626	1392	2116
(1000,1000)	(500,1100)	3	9	46	83
(50,50)	(25,100)	948	1611	3006	3757
(50,50)	(25,250)	716	1074	1567	1901
(50,50)	(25,550)	325	357	555	593

energy,  $\Delta(\Sigma d^2)$ , is 2 for both node 5 and node 2. Assume node 2 is selected and inserted between node 1 and node 3. The link between node 1 and node 3 is discarded. The similar procedure continues until the chain 1-2-4-6-3-5 links all nodes at round 5. Compared to the chain in Figure 5, the energy consumption ( $\Sigma d^2$ ) for this chain is 15 down from 19.

Although Algorithm 2, which has complexity  $O(n^3)$ , is more complex than Algorithm 1, the data transmission through the chain constructed in Algorithm 2 consumes less energy than that through the chain constructed in Algorithm 1. Therefore, chain protocol with Algorithm 2 conserves more power in the network than that with Algorithm 1.

## 5 Simulation

For our simulations we randomly generated sensor networks depending on predefined node number and area size. The initial energy of each node is within the range of (0.5J, 1.0J). The sensors are randomly distributed among the rectangular sensing area. All the nodes and BS are stationary.

Using this network configuration in the simulation of data gathering, we ran each protocol and tracked its progress in terms of the number of rounds of data transmission. We are interested in the number of rounds when first node and last node die. The more important one is the former. In sensor networks, a decrease in the number of available users can significantly degrade the network performance due to the fact that either some areas are not monitored accurately or that some data cannot be transferred within the network. The network will also be partitioned when alive nodes become less and less [15]. In this paper, we measure the network lifetime as the number of rounds when first node fault occurs. For each simulation, we create 30 random networks and average the results. We assume each transmission delivers 2000 bits of data due to data aggregation performed in each node.

**Table 2. Comparison of chain construction algorithms in binary scheme for sensor networks (A1: Algorithm 1, A2: Algorithm 2).**

Node #	Base Station	First Node		Last Node	
		A1	A2	A1	A2
25	(25,150)	1232	1253	2320	2394
50	(25,150)	1421	1592	3040	3022
100	(25,150)	1668	1912	3507	3602
200	(25,150)	1647	2127	4069	4111
50	(50,250)	1055	1114	1875	1900
50	(50,500)	423	425	697	700

**Table 3. Broadcasting in sensor networks with 50 nodes (urban model).**

Protocol	Rec.	Sending				
		(10,10)	(50,50)	(100,100)	(500,500)	(1K,1K)
1-linear	0.0211	0.0257	2.890	37.95	31047	384633
2-linear	0.0211	0.0216	0.292	3.78	2656	38896
1-collect	0.0235	0.0233	0.611	8.80	6322	84496
2-collect	0.0235	0.0228	0.320	3.84	2474	36467
1-relay	0.0224	0.0221	0.614	8.85	6368	84087
2-relay	0.0224	0.0215	0.310	3.68	2389	34763

First, we compare the performance of the chain constructed in Algorithm 1 (A1) with that of the randomly generated chain (R). In Table 1 we observe that in highly dense networks or with long distance BS, the results of two protocols are almost identical. In these situations establishing an energy-efficient chain is not the major issue of chain-based protocols. The reason is that distances between nodes are quite small in the dense case and any node has a chance of being selected in the construction process. The type of chain does not affect the simulation result. When BS is far from the sensing area, transferring data from *LEADER* to BS consumes the most power in each round, so that energy cost for data gathering along the chain is small and can be neglected. the

We conducted the simulation to compare two chain construction algorithms. The results are shown in Table 2. It shows that with moderate density and distance of BS, chain established using our scheme would live from 15% to 30% longer than Algorithm 1 before the first node dies. Our algorithm improves the lifetime of networks.

Using the same configuration, we can compare three chain schemes separately with two construction approaches for all-to-all broadcasting in sensor networks. For example, 1-linear in Table 3 denotes the linear-chain scheme that uses chain-construction Algorithm 1 to build the communication chain, 1-collect denotes the collecting approach in multiple-chain protocol using chain-construction Algorithm 1, and 1-relay denotes the relaying approach with the same chain-

**Table 4. Broadcasting in sensor networks within (500,500) (urban model).**

Protocol	Sending				
	10	50	100	200	500
1-linear	5331	31047	46070	88687	269393
2-linear	2431	2656	2201	2133	2037
1-collect	1586	6322	10901	21217	59657
2-collect	1509	2474	2670	2529	2435
1-relay	1751	6367	10765	20859	58904
2-relay	1655	2389	2478	2294	2185

construction algorithm. In this experiment, transmission to BS is not considered and there is no data aggregation. Each data packet consists of a 40-byte header and 20 bytes of information. To conserve energy, packets are combined with a single header. We measure energy cost of one round all-to-all broadcasting. Based on the data shown in Tables 3 and 4, we conclude that

- All cases have similar performance of receiving energy cost because each node needs to receive all the information. The more nodes the network has and the larger the area is, the smaller the percentage of the total energy cost is for receiving data. We observe that the energy consumption for receiving of the relaying approach is less than that of collecting. This result confirms our analysis that the relaying approach reduces the data redundancy.
- In the same chain-based protocol, transmission along the chain established by Algorithm 2 consumes less energy than that by Algorithm 1. For example, in a 50-node network, with Algorithm 2 energy consumption for linear-chain is only 10% of that with Algorithm 1. For multiple-chain with Algorithm 2 energy consumption is about 40% of that with Algorithm 1. We also observe that energy conservation is related to the size of networks. In Table 4, for a 100-node network, liner-chain by Algorithm 2 consumes 5% of the energy consumed by the chain constructed by Algorithm 1. And for 200- and 500-node networks, it is about 2.5% and 1%, respectively.
- The linear-chain scheme with Algorithm 2 consumes the least amount of energy in the dense case. Normally any chain scheme with Algorithm 2 has the opportunity to win depending on the density of the network. Increasing nodes, the best performance changes from multiple chain *Collecting* scheme to multiple chain *Relaying* scheme, finally the linear-chain scheme wins in the dense situation.

## 6 Conclusion

In this paper, we proposed a new chain protocol, multiple-chain protocol, for all-to-all broadcasting in wireless sensor networks. A new chain-construction algorithm, which establishes a chain with a minimum total energy, is also presented. We have evaluated different chain-based protocols through simulations: linear-chain, binary-combining-chain and newly proposed multiple-chain. Due to various network topologies and applications, it is impossible that one optimal approach exists which is suitable for all situations. Binary combining scheme works efficiently in sensor networks to balance transmission energy cost and latency. For all-to-all broadcasting, linear-chain consumes less energy per round in the dense case, while multiple-chain wins in relatively sparse networks. However, the bottleneck problem occurs in the multiple-chain scheme. Although the process of our chain construction algorithm is complex, the chain generated using our algorithm saves power in all-to-all broadcasting and prolongs lifetime in sensor networks compared to the shortest neighbor distance chain scheme. As part of future work, we will consider dynamic center for multiple-chain to reduce the bottleneck problem. We will explore the possibility to extend the multiple-chain scheme for data gathering. We will conduct more research works to develop other protocols for power-constrained networks, such as building clusters among networks other than the chain scheme.

## References

- [1] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan. Upper bounds on the lifetime of sensor networks. *Proceedings of IEEE ICC'2001*. Helsinki, Finland, Jun. 2001, pp.785-790.
- [2] J. H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. *Proceedings of IEEE INFOCOM2000*. Tel Aviv, Israel, March. 2000, pp.22-31.
- [3] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. *Proceedings of MOBICOM'99*. Aug. 1999, Seattle, WA, pp. 263-270.
- [4] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. Jan. 2000, pp.3005-3014.
- [5] W. R. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. *Proceedings of MOBI-COM'99*. Aug. 1999, Seattle, WA, pp. 174-185.
- [6] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. *Proceedings of ACM MobiCom'00*. Boston, MA, USA, Aug. 2000, pp.56-67.
- [7] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. Chen. *A Survey of Energy Efficient Network Protocols for Wireless Networks*. Kluwer Academic Publishers. 2001.
- [8] S. Lindsey and C.S. Raghavendra. Energy efficient broadcasting for situation awareness in ad hoc networks. *Proceedings of ICPP01*. Valencia, Spain, Sept. 2001, pp.149-155.
- [9] S. Lindsey, C.S. Raghavendra, and K. Sivalingam. Data gathering in sensor networks using the energy\*delay metric. *Proceedings of the IPDPS Workshop on Issues in Wireless Networks and Mobile Computing*. San Francisco, CA, Apr. 2001, pp.2001-2008.
- [10] A. Manjeshwar and D. P. Agrawal. TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. *Proceedings of the IPDPS Workshop on Issues in Wireless Networks and Mobile Computing*. San Francisco, CA, Apr. 2001, pp.2009-2015.
- [11] G. J. Pottie and W. J. Kaiser. Wireless integrated network sensors. *Communications of the ACM*. Vol.43, No.5, May. 2000, pp. 51-58.
- [12] K. Sohrahi, J. Gao, V. Ailawadhi, and G. J. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communication*. October 2000, pp.16-27.
- [13] I. Stojmenovic and X. Lin. Power-aware localized routing in wireless networks. *IEEE transactions on parallel and distributed systems*. Vol. 12, No. 11, Nov. 2001, pp. 1122-1133.
- [14] L. Subramanian and R. H. Katz. An architecture for building self-configurable systems. *Proceedings of the First Annual Workshop on Mobile and Ad Hoc Networking and Computing*. Boston, MA, Nov. 2000, pp.63-73.
- [15] C. K. Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE Communications Magazine*. June. 2001, pp. 138-147.