An Efficient Grid-based Data Aggregation Scheme for Wireless Sensor Networks

Neng-Chung Wang, Yung-Kuei Chiang, Chih-Hung Hsieh

Department of Computer Science and Information Engineering, National United University, Taiwan
{ncwang, ykchiang}@nuu.edu.tw, x06231@gmail.com

Abstract

Energy efficiency is a very important issue in wireless sensor networks (WSNs). Sensor nodes are battery-powered, thus the energy of them is limited. Data transmission is energy-intensive for sensor nodes, particularly over long distance. In this paper, an efficient Grid-Based Data Aggregation Scheme (GBDAS) for WSNs is proposed. In GBDAS, the grid structure is constructed by dividing the sensor network into a grid of cells, so that most of sensor nodes merely transmit the sensed data within the local cell. Only a small part of them need disseminate data towards the BS, greatly decreasing the energy load of sensor nodes. Besides, the position of each dissemination node is selected in rotation so that the energy depletion of sensor nodes is equally distributed. Consequently, the lifetime of network is extended.

Keywords: Base station, Cell head, Data aggregation, Grid-based, Wireless sensor networks

1 Introduction

Wireless sensor networks (WSNs) are extensively applied in many areas such as environmental monitoring, earthquake detection, and intelligent transport monitoring. They are also wildly utilized for military purposes such as object detection, target tracking, and security surveillance [1-4]. WSNs usually comprise of a great many small, low-priced, and power-limited sensor nodes which can gather, process, and transmit data via wireless channel [5].

Since the energy of sensor nodes is constrained, energy efficiency is one of the core issues in WSNs. It is energy-intensive for sensor nodes to transmit data, particularly over long distance. Because the base station (BS) is placed far away, the energy will be used up very soon for sensor nodes to transmit their data to the BS. Therefore, it is crucial for an approach to decrease the data transmissions to the BS as many as possible.

Many improved approaches have been presented to try reducing the data transmissions. For example, some researches use only a few of sensor nodes to transmit data to the BS instead. Besides, the data aggregation is performed by dissemination nodes not only to diminish redundancy and data transmissions but conserve energy as well. Secure data aggregation protocols is one of the most important issues in WSNs. Some studies focus on secure dataaggregation schemes to improve the security of data aggregation in WSNs [6-7].

In this paper, an efficient Grid-Based Data Aggregation Scheme (GBDAS) for WSNs is proposed. Initially, GBDAS equally divides the sensor network into a grid of cells to construct a grid infrastructure. In each cell, the node with most residual energy is chosen as a head (hereafter called a cell head). The rest nodes of the cell are called ordinary nodes. A chain is further formed by connecting all cell heads together using the greedy algorithm. In each cell, the ordinary nodes periodically transmit the sensed data to the cell head, and then merely enter into sleep mode in terms of the GAF protocol [8]. After receiving the data, the cell head aggregates with its own and disseminates towards the chain leader which is the cell head with most residual energy. Then, the chain leader transmits the gathered data to the BS. Since only the cell heads participate in the data dissemination, greatly decreasing the number of data transmissions.

The remaining of this paper is organized as follows. In Section 2, some related work is briefly reviewed. Section 3 mainly describes the proposed scheme. The simulation results are presented in Section 4. Finally, Section 5 concludes this paper.

2 Related Work

Recently, lots of researches focused on energy efficient for extending the network lifetime in WSNs. We review some of relevant protocols that is classified into three fields: cluster-based, chain-based, and grid-based.

In the cluster-based approach, LEACH [9] is a traditional cluster-based protocol, in which the sensor nodes organize themselves into clusters, each with a

*Corresponding Author: Neng-Chung Wang; E-mail: ncwang@nuu.edu.tw
DOI: 10.3966/160792642018121907022

special node called cluster head (CH). LEACH performs data aggregation to diminish the data transmissions. In each cluster, all non CH nodes transmit their data to the CH, which aggregates with its own and then transmits it to the remote BS. Since the CH consume more energy than a non CH node, all sensor nodes in each cluster take turns to be the CH to uniformly share the energy load. LEACH divides the operation into rounds, each having two phases. In a set-up phase, the clusters are first organized. In a steady-state phase, the sensed data are transmitted from the nodes to the CH and then directly towards the BS. HCTE [10] is another cluster-based routing protocol, combining new CH selection and routing algorithms. The CH selection algorithm is performed in two separate stages with two CHs, initial CH and second CH, in each cluster. The initial CH is responsible for creating the cluster, collecting data from the other cluster members, and transmitting the collected data to the second CH, which in turn disseminates the collected data to the BS through multi-hops.

In the chain-based approach, PEGASIS [11] is the earliest chain-based protocol that constructs the chain by connecting all sensor nodes using a greedy algorithm. Each node is responsible for sensing data, receiving data from one neighbor, fusing them with its own and then transmitting them to another neighbor on the chain. For collecting data in each round, a sensor node is designated as a leader on the chain. Collected data travels recursively from one node along the chain and finally the leader transmits them to the BS. EECB [12] further improves PEGASIS by evenly distributing the energy depletion among the sensor nodes with a distance threshold. EECB chooses the node with lowest cost as the leader to eventually send the gathered data to the BS.

In the grid-based approach, TTDD [13] is a per-source grid-based protocol that enables efficient data transmission from multiple sources to multiple mobile sinks. In TTDD, the dissemination node at each grid point is necessary to acquire the data forwarding information. After receiving the query from the sink, the immediate dissemination node in turn forwards it towards the source. TTDD builds a grid structure, thus only dissemination nodes need to set up the data forwarding information. GMEAR [14] partitions the network area into non-overlapping zones with equal size in a grid-by-grid manner. The sink is located at the center of the sensing field. Sensor nodes are randomly and uniformly deployed around the sink. Routing follows grid to grid and then tries to follow the diagonal path if the source node is within some defined range. But, if the source node is on diagonal, then routing follows the original diagonal path. In each grid, the node with more energy has high probability to relay route discovering requests to neighboring grids, and propagate data to a neighboring grid within some defined range. In this way, the data is finally transmitted to the sink.

In the following, we describe the differences between the proposed GBDAS and PEGASIS. PEGASIS is a chain-based data aggregation scheme, In PEGASIS, the chain by connecting all sensor nodes using a greedy algorithm. Data transmission is conducted along the constructed chain. The proposed GBDAS scheme is a hierarchical data aggregation scheme that contains two tiers data transmission. The sensor field is divided into cells of grid. The low tier data transmission is the intra-cell transmission from source nodes to cell head in each cell that use direct transmission, and then the data is fused in the cell head. The high tier data transmission is the inter-cell transmission from cell heads to BS that use chain-based routing (like PEGASIS) for data aggregation of cell heads. A framework of the proposed GBDAS is shown in Figure 1. In order to understand easily the differences for various protocols, we present a comparison between our scheme and related approaches in Table 1.

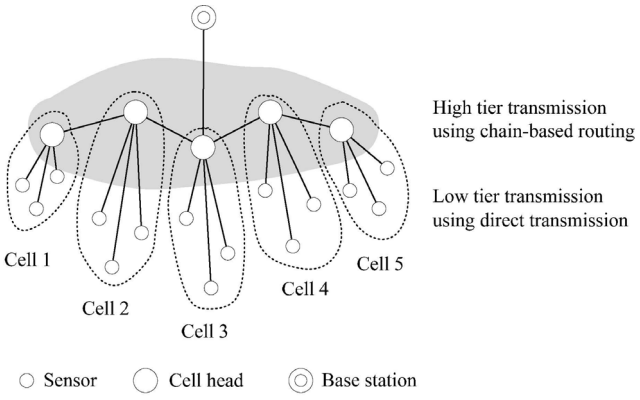


Figure 1. A framework of the proposed GBDAS

Table 1. A comparison between the proposed GBDAS and related approaches

	Classified field	Hierarchical architecture	Energy Load Distributed	Energy efficient
Direct transmission	No	No	No	No
LEACH	Cluster-based	Yes	Low	Low
HCTE	Cluster-based	Yes	Medium	Medium
PEGASIS	Chain-based	No	Low	Medium
EECB	Chain-based	No	Medium	Medium
TTDD	Grid-based	No	No	Medium
GMEAR	Grid-based	Yes	Medium	Medium
Proposed GBDAS	Grid-based	Yes	High	High

3 The Proposed Scheme

We first make assumptions about the network model as follows. Each sensor node should be aware of its residual energy and its geographic position via some positioning techniques such as Recursive Position

Estimation (RPE) and Global Positioning System (GPS) [15-16]. In our applications, only a fraction of sensor nodes are equipped with GPS under cost considerations, and those with such capability are called beacons. Therefore the other nodes can apply the RPE position algorithm to estimate their positions with the help of beacon nodes. All sensor nodes stay immovable after they were deployed.

Like most network model in WSNs, all sensor nodes are assumed to be immovable and position-aware in GBDAS. Lots of researches have studied the issue of sensor node deployment [17-18]. The sensor node deployment is application-specific. However, an appropriate node deployment approach can lower the complexity of problems in WSNs such as routing, data fusion, communication, etc. Moreover, it can prolong the network's lifetime by minimizing energy depletion. In our proposed scheme, we randomly deployed sensor nodes with uniform density to simulate some real environments, especially when a *priori* knowledge of the terrain is not available. Therefore, we applied grid infrastructure for our proposed scheme in terms of the following three aspects. First, each sensor node can easily determine which cell it locates in. Second, grid infrastructure can lead proper sensor node distribution. Third, the data transmission between cell heads along the chain is simple. The result of this research might be theoretically better to evenly deploy the sensor nodes though.

3.1 Grid Construction

Initially, the sensor network is equally divided into a 2-Dimension virtual grid of $M \times N$ cells. A sensor node can identify its cell ID $Cell[i, j]$ using its geographic position (x, y) as follows:

$$i = \left\lfloor \frac{x - x_0}{\alpha} \right\rfloor, j = \left\lfloor \frac{y - y_0}{\alpha} \right\rfloor \quad \textbf{(1)}$$

where $\lfloor k \rfloor$ is a floor function of k, α is the cell size, and (x_0, y_0) is the position of the virtual origin determined at the network setup stage such that $0 \le i \le M\text{-}1$ and $0 \le j \le N\text{-}1$. For simplicity, it is assumed that all cell IDs are non-negative.

Algorithm *GridConstruction* as shown below was implemented for grid construction in our simulation.

Algorithm 1. GridConstruction (*node_list*)

Purpose: Give a list of sensor nodes with the geographic position to create a grid of cells with their members

```
{
  // Initialize each cell without any member node
  for (i = 0; i ≤ M-1; i++)
    for (j = 0; j ≤ N-1; j++)
      Cell[i, j] ← NULL;
  ptr ← node_list;
  while (ptr != NULL) {
    i ← ⌊(x - x0)/α⌋;
    j ← ⌊(y - y0)/α⌋;
    // Add this node to the list of Cell[i, j]
    AddList(ptr.node, Cell[i, j]);
    ptr ← ptr.next;
  }
  return Cell[M, N];
}
```

3.2 Cell Head Election

For electing the cell head, an energy threshold E_t for each cell is first computed as Eq. (2):

$$E_t = \frac{1}{m} \sum_{i=1}^{m} E_i \quad \textbf{(2)}$$

where m is the number of sensor nodes in the cell. Nodes with residual energy not less than E_t are qualified for being the candidate of the cell head. A qualified sensor node s will then join the competition by broadcasting a *Head_election* message carrying its position after delaying for a certain time period $f(s)$, which is calculated as follows:

$$f(s) = K / RE(s) \quad \textbf{(3)}$$

where K is a constant determined at the network setup stage and $RE(s)$ is the residual energy of sensor node s. Thus, any sensor node in the same cell receives *Head_election*, it will just yield because of its less residual energy. If multiple broadcasting messages occur, the node with smallest position will win. For simplicity, position $L_1(x_1, y_1)$ is smaller than position $L_2(x_2, y_2)$ if and only if x_1 is smaller than x_2. Finally, the winner will broadcast a *Head_confirm* message carrying its position. The ordinary nodes update their current cell head with this new position. Accordingly, the cell head in the cell is determined. Algorithm *CellHeadElection* as shown below was implemented for cell head election in our simulation.

Since the energy load of a cell head is extremely heavy, it will be used up very soon. When the residual energy is lower than the threshold E_t, the cell head will broadcast an announcement for a new cell head election. All qualified ordinary nodes in the cell will participate in the contest applying the Algorithm *CellHeadEelection* as stated above.

Algorithm 2. CellHeadElection(*Cell*[*i*, *j*])

Purpose: Give a list of member nodes of a cell *Cell*[*i*, *j*] to elect the cell head

```
{
   SUM = 0;
   for (i = 1; i <= m; i++) {
      SUM = SUM + E[i];
   }
   Et = SUM / m;   // Compute the energy threshold
   ptr ← Cell[i, j];
   t_min ← ∞;
   while (ptr != NULL) {
      s ← ptr.node;
      if (RE(s) < ET)
         continue;
      // Compute the time period for sensor node s
      t(s) = K / RE(s);
      if ((t(s) < t_min) or ((t(s) == t_min) and
(Loc(s) < Loc(CellHead[i, j])))) {
         CellHead[i, j] ← s;
         t_min ← t(s);
      }
      ptr ← ptr.next;
   }
   return CellHead[i, j];
}
```

3.3 Chain Formation

In GBDAS, the main idea of forming a chain is to connect all cell heads together. In chain formation phase, it begins with most distant cell head away from the BS applying the greedy algorithm. The most distant cell head sends its nearest downlink cell head a *Chain_formation* message with its own cell ID. Then, the downlink cell head caches the ID and replies with a *Chain_replying* message with its own cell ID to the uplink cell head. The uplink cell head also caches the ID in the *Chain_replying* message. Each cell head caches both IDs of its uplink and downlink cell heads. The chain is finally formed by connecting the cell heads together, as shown in Figure 2. Algorithm *ChainFormation* as shown below was implemented for chain formation in our simulation.

Figure 2 illustrates the chain formation process, starting from cell head h_0 to cell head h_1, then to cell head h_2, etc., and finally to cell head h_{15}. The cell head caches both IDs of its uplink and downlink except both ends. As illustrated in Figure 2, the chain starts from h_0 without uplink cell head and ends at h_{15} without downlink cell head. The cell heads of the grid are finally connected together to form a chain. Among the cell heads, the one with most residual energy will be designated as the chain leader by the BS in each data gathering round. After receiving the data from its two neighboring cell heads (if there are two neighbors), the chain leader aggregates with its own, and finally transmits to the BS.

Algorithm 3. ChainFormation (*head_list*)

Purpose: Give a list of cell heads to create a chain of the cell heads

```
{
   // Find the farthest cell head away from the BS
   current ← MaxDist(BS, head_list);
   AddChain(current, Chain);
   RemoveList(current, head_list);
   while (head_list is not empty) {
      // Find the nearest cell head away from the current
      // head
      nearest ← MinDist(current, head_list);
      AddChain(nearest, Chain);
      current ← nearest;
      RemoveList(current, head_list);
   }
    return Chain;
}
```

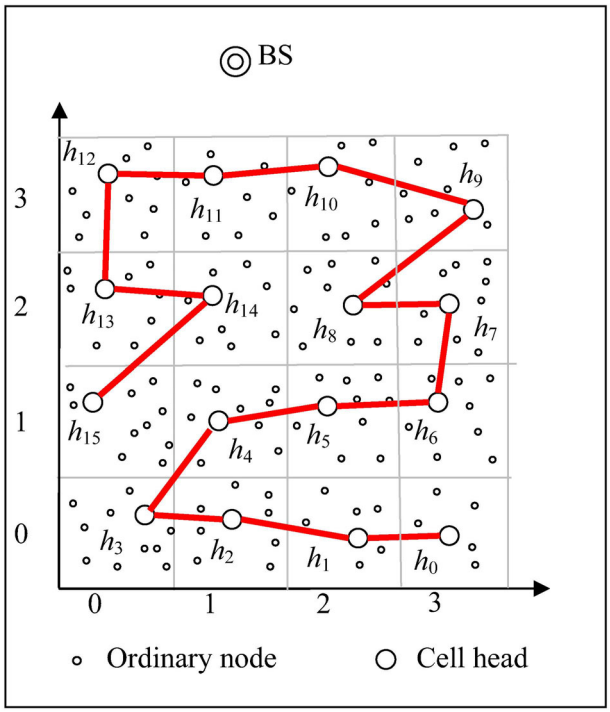


Figure 2. Chain formation by the greedy algorithm

3.4 Data Transmission

The activity of data gathering is separated into rounds. Each round the BS send a request to the chain leader, which will send two tokens: t_1 and t_2, to its two neighbors respectively. Token t_1 will be transmitted recursively along the chain clockwise. Conversely, token t_2 will be transmitted recursively along the chain counterclockwise. Each token travels from head to head along the chain in its own direction. Upon getting the token, the end cell head forwards its aggregated data to its uplink head in reverse direction. After receiving the aggregated data, the uplink head also aggregates with its own and then disseminates towards the chain leader, which waits for the data transmitted

from both neighbors and then aggregates with its own. At last, the leader transmits only one message to the BS.

For example as illustrated in Figure 3, the BS chooses h_{10} as the chain leader. After receiving the request, h_{10} sends two tokens t_1 and t_2 to its two neighbors, h_9 and h_{11} respectively. Token t_1 moves from h_9 along the chain clockwise to the end node h_0. After receiving t_1, h_0 transmits its aggregated data to its uplink head h_1. Likewise, h_1 aggregates its own data with the data received from h_0 and then recursively transmits to h_2, and eventually to h_{10}. The chain leader h_{10} waits for the data transmitted from both neighbors h_9 and h_{11}, and then aggregates with its own. Finally, h_{10} transmits only one message to the BS.

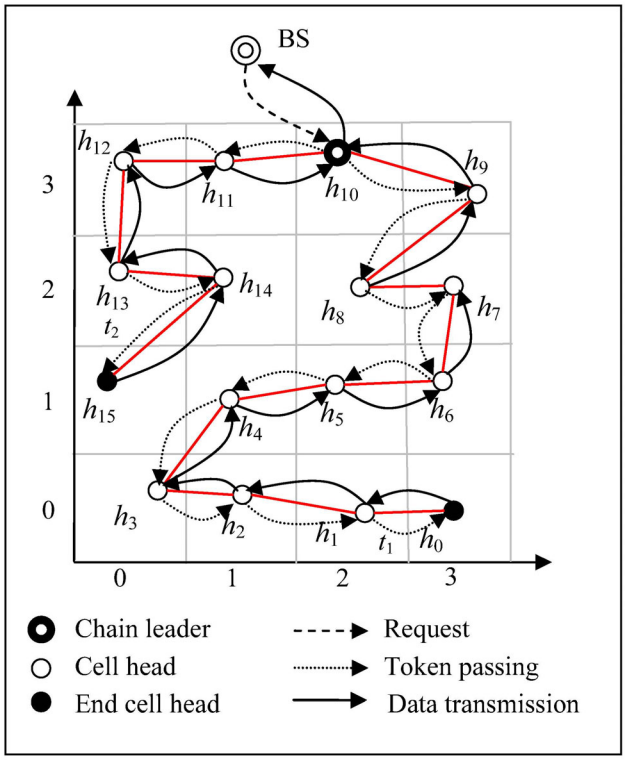


Figure 3. An example of data transmission

3.5 Discussion

In GBDAS, the cell head or the chain leader is determined as the one with most residual energy only, without considering the factor of distance. Since it is energy-intensive for a sensor node to transmit data, particularly over a long distance, the possibility of it to be the cell head or the chain leader will naturally decrease next turn. Therefore, the energy load as the cell head or the chain leader will be evenly distributed.

There are three algorithms in GBDAS, *GridConstruction*, *CellHeadElection*, and *ChainFormation*. Algorithm *GridConstruction* partitions the network into a grid of $M \times N$ cells, which contains sensor nodes. By using Eq. (1), each sensor node decides the cell it locates in. The computation complexity of Algorithm *GridConstruction* is $O(n)$, where n is the number of sensor nodes in the network. The computation complexity of Algorithm *CellHeadElection* is $O(m)$, where m is the number of sensor nodes in a cell. Assume that the sensor nodes are uniformly deployed in the network, then we have $m \approx \frac{n}{M \times N}$. Algorithm *ChainFormation* further connects all the cell heads together to form the chain. Therefore, its complexity is $O(M \times N)$. As to other traditional approaches [11-12] using the greedy method, the complexity of them is $O(n^2)$, where n is much greater than $M \times N$.

Although wireless sensor nodes have minimum storage capabilities, they generally just need a small amount of program memory and data storage. Modern microcontrollers contain up to 128 KB of on-chip program storage, which can be utilized as both program memory and data storage [19]. In WSNs, memory requirement is application dependent. Storing or updating data on the device's memory requires energy. Therefore, data is only cached long enough so that it can be processed and then sent out to reduce the energy consumption.

In WSNs, an outage occurs if the destination cannot successfully receive the sender's message. The outage is detected by timeout. If the sender never receives acknowledgement from the receiver for a certain time, the outage is detected. During the cell head election phase, each sensor node competes for being the cell head. If the outage occurs, the sender just fails the competition. During the chain formation phase or the data transmission phase, if the outage occurs, the sender simply retransmits. However, it is probable for a sender to send a message to a void cell, where all nodes are not longer operative or there are no nodes at all. If the outage occurs for the first time, the sender just retransmits as usual. If the outage occurs again, the sender will find a roundabout path by detouring the cell.

4 Simulation Results

For the sake of consistency, the first order radio model is applied as by PEGASIS to evaluate the energy depletion of all sensor nodes. In terms of this model, the radio uses E_{elec} = 50 nJoule/bit to operate the transmitter circuitry or the receiver circuitry. A transmission amplifier for a sender node further uses $E_{amp}d^2$ to transmit messages, where E_{amp} = 100 pJoule/bit/m^2 and d is the distance between nodes. Thus, the cost to transmit a k-bit message away a distance d is:

$$E_{Tx}(k, d) = E_{elec} \times k + E_{amp} \times k \times d^2 \quad \textbf{(4)}$$

and the cost to receive this message is:

$$E_{Rx}(k) = E_{elec} \times k \quad \textbf{(5)}$$

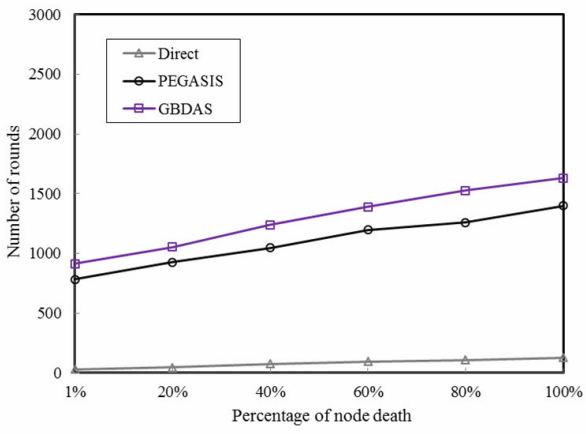
Since either transmitting or receiving data is a high-cost operation, many schemes perform data aggregation to reduce the number of data transmissions

and receptions. Lots of data aggregation techniques have been discussed in Fasolo *et al.*'s survey research [20]. GBDAS performs data aggregation by two phases in each data gathering round. The scenario is that physical quantity of temperature is gathered by each sensor node, storing in a 2000-bit message. For the first phase, each ordinary node sends the sensed data to its cell head. After receiving all the data, the cell head aggregates with its own by generating a single message of the same length. For the second phase, each cell head receives the message from its downlink head, it also aggregates with its own to generate a single message of the same length and then transmits to its uplink head towards the chain leader. In GBDAS, all sensed data of a cell are aggregated into a single message, only two messages from both ends are transmitted simultaneously on the chain, greatly decreasing the data transmissions and thereby enhancing the network's lifetime.

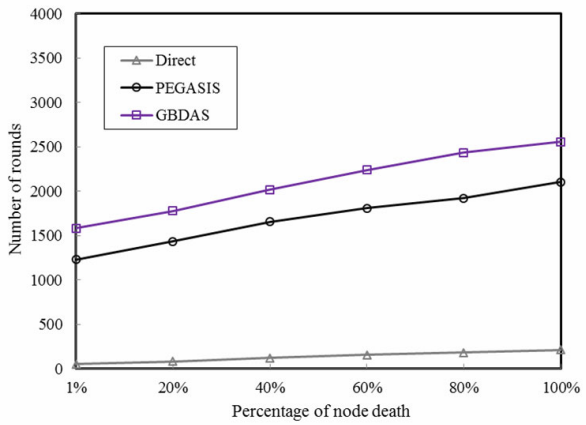
To evaluate the performance of GBDAS, we simulated GBDAS, Direct protocol, and PEGASIS [11]. In Direct protocol, all sensor nodes just use one hop to directly transmit their data to the BS. We developed a simulator based on MATLAB for performance evaluation. The sensor nodes were deployed uniformly in a 100 m×100 m field. The packet length was 2000 bits. The number of sensor nodes was 300. The BS is located at (50, 100). The data aggregation energy was 5 nJoule/bit/message.

4.1 Impact of Initial Energy

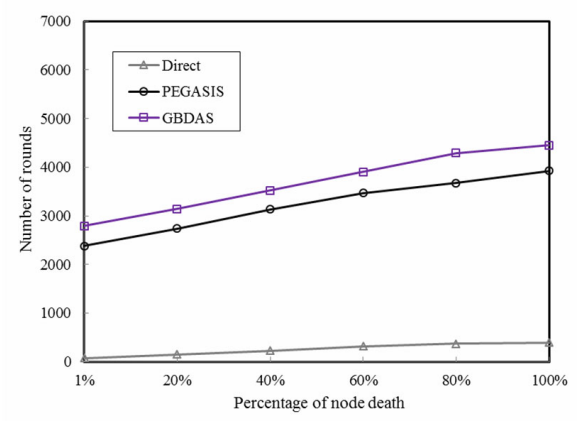
First, we study the impact of initial energy on GBDAS, Direct, and PEGASIS with three different initial energy: 0.25 J, 0.5 J, 1.0 J, respectively. In GBDAS, 10 × 10 cells and 300 sensor nodes were applied. The simulations were conducted to examine the number of rounds when 1%, 20%, 40%, 60%, 80%, and 100% nodes died using Direct, PEGASIS, and GBDAS. As shown in Figure 4(a) to Figure 4(c), GBDAS is better than Direct and PEGASIS. The number of rounds grows quite uniformly using both GBDAS and PEGASIS. However, the gap between the growth rates of both approaches is getting wider after about 60% nodes die. The reason is that the distances between nodes become farther using PEGASIS so that the energy is exhausted quickly. GBDAS averagely has nearly 1.2 times more rounds compared with PEGASIS. As can be expected, the more initial energy per node, the more rounds for each approach.



(a) initial energy = 0.25 J



(b) initial energy = 0.5 J



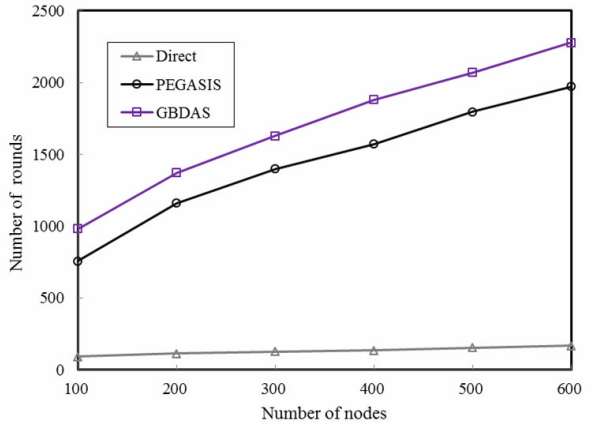
(c) initial energy = 1.0 J

Figure 4. Impact of initial energy

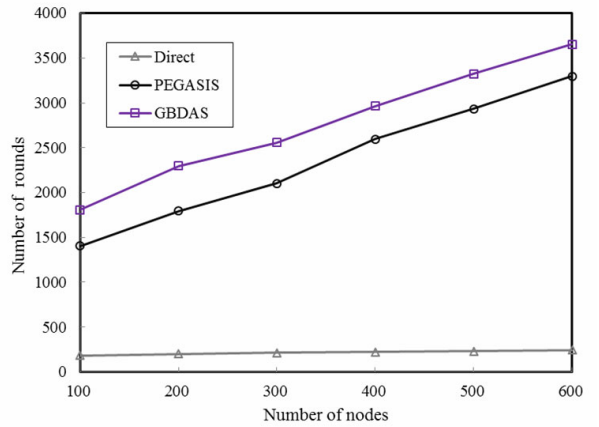
4.2 Network Lifetime

To explore the network lifetime on GBDAS, we used 6 × 6, 8 × 8, and 10 × 10 cells, respectively. In the simulations, 300 sensor nodes with the initial energy of 0.5 J were used. A sensor node is regarded as inoperative if its residual energy is not enough to receive or transmit data. As shown in Figure 5(a) to Figure 5(c), GBDAS runs the network longer than the other two approaches. A grid of 10 × 10 cells is also most suitable for GBDAS among the three different

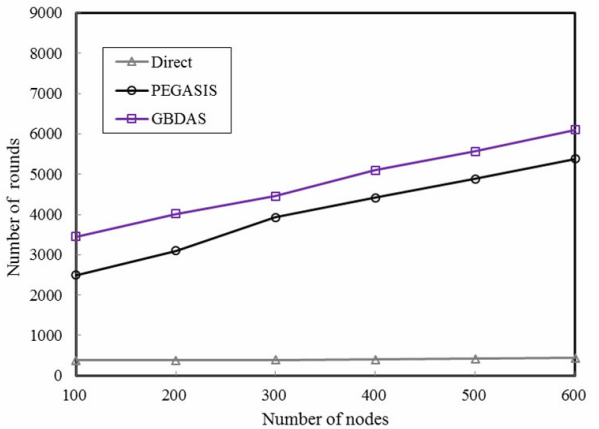
partitions. In GBDAS, each ordinary sensor node only needs to locally transmit the sensed data within the cell, comparing with the other two approaches. Only the cell heads participate in forwarding the aggregated data towards the chain leader. Moreover, all cell heads and the chain leader are selected in rotation, so the energy load of the cell head and the chain leader is evenly shared. Consequently, each node's lifetime is extended and the entire sensor network's lifetime is also prolonged.



(a) 6 × 6 cells



(b) 8 × 8 cells



(c) 10 × 10 cells

Figure 5. Number of alive nodes over time

4.3 Total Consumed Energy

To evaluate the performance of GBDAS, we examine the change of the total consumed energy, including communication, computation, and sensing energy depletion. In the simulations, 300 nodes with initial energy of 0.5 J were used in each approach. Therefore, the total energy which can be consumed is 150 J for each approach. In GBDAS, a grid of 10 × 10 cells is applied. Figure 6 illustrates that the total consumed energy increases as the number of rounds increases; however, among the three approaches, the total consumed energy of GBDAS is exhausted most slowly. The total consumed energy is also least among the three approaches before 100 rounds of data gathering. This is because that the algorithm *GridConstruction* only needs to execute once during the network setup time. The algorithm *CellHeadElection* uses an energy threshold to trigger the execution, without election each round. The algorithm *ChainFormation* also executes once during the chain formation phase. Afterward the chain is maintained by the cell heads by simply caching the IDs of their own uplink and downlink cell heads. Besides, the other favorable policies also help so that the number of rounds can be over 2,500.

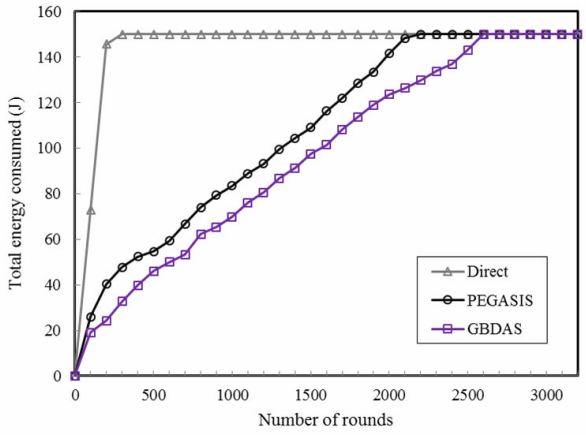


Figure 6. Total consumed energy over time

5 Conclusion

In this paper, we described an efficient grid-based data aggregation scheme for WSNs. GBDAS first constructs the grid infrastructure once and then connects the cell heads to form the chain using a greedy algorithm. In GBDAS, all sensor nodes are immobile and position-aware so that GBDAS can efficiently constructs and maintains the grid infrastructure and chain connection. Data transmission of all ordinary sensor nodes is just allowed within local cell. Only the cell heads are necessary to aggregate data and disseminate towards the chain leader. The position of the cell heads and the chain leader is selected in terms of the residual energy level to

distribute the energy load among the sensor nodes. Consequently, the individual sensor node's lifetime is extended, thereby the entire sensor network's lifetime is enhanced. Our analysis and simulations have verified GBDAS is energy efficient.

Acknowledgments

This work was supported by the National Science Council of Republic of China under grants NSC-104-2221-E-239-009 and NSC-105-2221-E-239-019.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A Survey on Sensor Networks, *IEEE Communication Magazine*, Vol. 40, No. 8, pp. 102-114, August, 2002.

[2] J. N. Al-Karaki, A. E. Kamal, Routing Techniques in Wireless Sensor Networks: A Survey, *IEEE Wireless Communications*, Vol. 11, No. 6, pp. 6-28, December, 2004.

[3] Q. Jiang, D. Manivannan, Routing Protocols for Sensor Networks, *Proceedings of IEEE Consumer Communications and Networking Conference*, Las Vegas, NV, 2004, pp. 93-98.

[4] R. S. Bisht, S. Mishra, Network Structure and Topology based Routing Techniques in Wireless Sensor Network– A Survey, *International Journal of Computer Applications*, Vol. 74, No. 14, pp. 5-10, July, 2013.

[5] S. D. Muruganathan, D. C. F. Ma, R. I. Bhasin, A. O. Fapojuwo, A Centralized Energy-Efficient Routing Protocol for Wireless Sensor Networks, *IEEE Communication Magazine*, Vol. 43, No. 3, pp. S8-13, March, 2005.

[6] F.-Y. Lei, C. Fu, X. Li, J. Chen, Secure Data Aggregation Solution Based on Dynamic Multiple Cluster Key Management Model, *Journal of Internet Technology*, Vol. 12, No. 3, pp. 465-475, May, 2011.

[7] K. Hou, B. Wang, Z. Wang, C. Zhu, A Review of Recent Developments about Secure Data Aggregation in Wireless Sensor Networks, *Journal of Internet Technology*, Vol. 17, No. 7, pp. 1511-1519, December, 2016.

[8] Y. Xu, J. Heidemannn, D. Estrin, Geographyinformed Energy Conservation for Ad Hoc Routing, *Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Rome, Italy, 2001, pp. 70-84.

[9] W. B. Heinzelman, A. P. Chandrakasan, H. Balakrishnan, An Application-Specific Protocol Architecture for Wireless Microsensor Networks, *IEEE Transactions on Wireless Communications*, Vol. 1, No. 4, pp. 660-670, October, 2002.

[10] Nasrin Azizi, Jaber Karimpour, Farid Seifi, HCTE: Hierarchical Clustering Based Routing Algorithm with Applying the Two Cluster heads in Each Cluster for Energy Balancing in WSN, *International Journal of Computer Science Issues*, Vol. 9, No. 1, pp. 57-61, January, 2012.

[11] S. Lindsey, C. S. Raghavendra, PEGASIS: Power-Efficient Gathering in Sensor Information Systems, *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, 2002, pp. 1125-1130.

[12] Y. C. Yu, Y. C. Song, An Energy-Efficient Chain-Based Routing Protocol in Wireless Sensor Network, *Proceedings of the International Conference on Computer Application and System Modeling*, Taiyuan, China, 2010, pp. 486-489.

[13] F. Ye, H. Luo, J. Cheng, S. Lu, L. Zhang, A Two-Tier Data Dissemination in Large-Scale Wireless Sensor Networks, *Proceedings of the Eighth Annual ACM/IEEE International Conference on Mobile Computing and Networks*, Atlanta, GA, 2002, pp. 148-159.

[14] D. Sharma, K. Sharma, Grid Based Multipath Energy Aware Routing Protocol for WSNs, *International Journal of Computer Trends and Technology*, Vol. 4, No. 4, pp. 729-733, April, 2013.

[15] J. Albowitz, A. Chen, L. Shang, Recursive Position Estimation in Sensor Networks, *Proceedings of the Ninth International Conference on Network Protocols*, Riverside, CA, 2001, pp. 35-41.

[16] E. D. Kaplan, *Understanding GPS: Principles and Applications*, Artech Hourse, 1996.

[17] Z. Li, L. Lei, Sensor Node Deployment in Wireless Sensor Networks Based on Improved Particle Swarm Optimization, *Proceedings of the IEEE International Conference on Applied Superconductivity and Electromagnetic Devices*, Chengdu, China, 2009, pp. 215-217.

[18] H. Zhang, C. Liu, A Review on Node Deployment of Wireless Sensor Network, *International Journal of Computer Science Issues*, Vol. 9, No. 3, pp. 378-383, November, 2012.

[19] J. L. Hill, *System Architecture for Wireless Sensor Networks*, Ph.D. Thesis, University of California, Berkeley, CA, 2003.

[20] E. Fasolo, M. Rossi, J. Widmer, M. Zorzi, In-Network Aggregation Techniques for Wireless Sensor Networks: A Survey, *IEEE Wireless Communications*, Vol. 14, No. 2, pp. 70-87, April, 2007.

Biographies

Neng-Chung Wang received the Ph.D. degrees in Computer Science and Information Engineering from National Cheng Kung University, Taiwan, in June 2002. Currently, he is a Full Professor at the Department of Computer Science and Information Engineering, National United University, Taiwan. His research interests include wireless networks and Internet of Things.

Yung-Kuei Chiang received the M.S. degree in computer science from University of Maryland, Baltimore County, in 1989. Currently, he is an Assistant Professor in the Department of Computer Science and Information Engineering, National United University, Taiwan. His research interests are mobile computing and wireless networks.

Chih-Hung Hsieh received the B.S. degree in Computer Science and Information Engineering from Ming Chuan University, Taiwan, in June 2012, and the M.S. degree in Computer Science and Information Engineering from National United University, Taiwan, in June 2014. His research interests include mobile computing and wireless networks.