Journal of Computer Science 7 (9): 1400-1406, 2011
ISSN 1549-3636

Consistent and Proficient Algorithm for Data Gathering in Wireless Sensor Networks

P. Balamurugan and K. Duraiswamy
Department of Computer Science and Engineering,
K.S. Rangasamy College of Technology, Tiruchengode -637 215, India

Abstract: Problem statement: A wireless sensor network is usually deployed in a harsh geographical area to gather data that can be delivered to the remotely located base station. Sensor nodes have irreplaceable energy source, limited computational capability and limited memory. It is a challenge to maximize the use of energy of these sensor nodes to extend the network lifetime. **Approach:** This study proposes a Rank-Based Data Gathering Algorithm for wireless sensor networks. Sensor nodes were randomly distributed in a network field of different sizes. For every round of data communication, the algorithm steps were as follows: A set of sensor nodes or vertices were given a random rank between 0 and 1. A link formed between any two nodes if they were within each other's transmission range. If a sensor node had the highest rank among its neighbors, then it considered an associate node, else it falls into the leaf node. Next, the associate nodes form a complete graph among them and later form a Rooted Directed Tree (RDT) after an implementation of Kruskal's Minimum Spanning Tree algorithm and the Breadth First Search algorithm. Finally, a model that takes into account the energy when deciding what type of node a sensor was implemented. **Results and Conclusion:** After recording the simulation results, it is concluded that the RBDG yields a better outcome in terms of lifetime and delay per round for TDMA than other popular data gathering algorithms.

Keywords: Rank based data gathering, energy awareness, network lifetime, delay per round, breadth first search, wireless sensor networks, power-efficient

INTRODUCTION

Wireless Sensor Networks are a series of sensors randomly or evenly distributed across a vast area used to monitor disaster areas, terrorist attack areas, forest fires etc. The sensors are located at random locations and relay their information to a central base that is usually far from the region of sensor nodes. Sensors usually have a few basic properties that come along with them: one or more sensors, a radio transceiver for communication, a microcontroller for computation and decision making and a battery for energy.

Data gathering algorithms are usually measured by executing the algorithm several rounds. In each round, data from all the sensor nodes are gathered and then forwarded to the sink. Data gathering algorithms are categorized based on the type of communication structure they will be using, such as clusters, grid, chain, connected dominating sets and trees. Different types of clustering and grid algorithms have been proposed in the literature review. The well examined algorithms known as Low Energy Adaptive Clustering Hierarchy (LEACH) and Power-Efficient Gathering in Sensor Information Systems (PEGASIS) were just a couple of the several algorithms comprehended while reviewing literature.

METERIALS AND METHODS

In this study, a Rank Based Data Gathering algorithm for data collaboration in a wireless sensor network is proposed. The algorithm works as follows: A set of nodes are randomly distributed within the given sensor network. Each node has an energy level of 1 joule and is given a rank between 0 and 1. The associate nodes are decided if a node has the highest rank among it and its neighbors. If a node cannot become an associate node, then it is known as a leaf node of the neighbor with the highest rank. If a node wasn't considered an associate node in the first step or as a leaf node in the second, then the remaining node's rank values are increased and then the associate node is found and then that node will be included in the associate list The data aggregation tree is comprised of

Corresponding Author: P. Balamurugan, Department of Computer Science and Engineering,
K.S. Rangasamy College of Technology, Tiruchengode -637 215, India

leaf nodes- whose rank are lower than a neighbor node, associate nodes- whose rank is highest among neighbors and root nodes-node with the highest rank within the sensor network. During data communication, leaf nodes forward their data to their associate node. The associate nodes then send their data to an upstream associate node. The root node gathers all the data from the downstream associate nodes and then sends the data to the sink, where the data is then fused. Data communication is occurs in rounds and a RBDG is executed if energy still remains in a node within the network. We compare RBDG with LEACH and PEGASIS through simulations conducted for both CDMA and TDMA system.

Literature: Within wireless sensor networks, sensor nodes usually have a lot of energy problems for gathering data so data collaboration algorithms are much needed in extensive situations. Many approaches have been taken to solve this problem, but nearly all of them have unwanted drawbacks. One approach taken to prove direct transmission as a counterproductive method was minimum-transmission-energy. When a simulation of MTE was performed, it was conclusive that the last node dies sooner using direct transmission than in MTE, thus showing evidence that MTE is more energy efficient. (Siva *et al*., 2005) Some of the more commonly used data gathering algorithms are clustering algorithms. These protocols group each set of nodes and allows for the cluster head to communicate to the sink. Two algorithms that will be discussed from our literature will be LEACH and PEGASIS.

During LEACH, the set up phase clusters the nodes leaving one as the head cluster. While in the steady state, the sensor nodes collect the data and later transmit the data to the cluster heads and then the cluster heads transmit the data to the sink. Cluster heads are chosen randomly and achieve an approach. Although LEACH protocol reduces energy utilization by a factor of 8, energy is consumed is forming cluster. Further, in LEACH protocol, 5% of the nodes are the head nodes at the same time that also amounts to energy consumption (Cauligi and Raghvendra, 2002). The PEGASIS process is completely different; it applies the greedy approach by presenting the furthest node as the starting node. Next the node closest to the start node is added to the chain and the process continues until all nodes are added to the chain. In each round, a generator randomly selects a node as the leader node and informs the rest of the network (Viterbi, 1995). The leader node is responsible for aggregating the information to the sink node. The first algorithm using PEGASIS used a time division multiple access approach (PEGASIS-TDMA), whose drawback was the length of the delay as data moves from the closest node to the next until it reaches the leader (Lindsey, *et al*., 2001). PEGASIS was later innovated with code division multiple accesses, which used chain based binary scheme to minimize the delay incurred and reduce the energy metric. For every round of gathering, each low level node (based on hierarchy) will transmit data to a higher node and continue the process until the data is gathered at the highest level. Then the aggregated data will be transmitted to the sink. The amount of energy used by LEACH is fairly lower than that consumed by either of the PEGASIS's different implementations. Because of the fact that nodes are likely to move further away from each other, PEGASIS CDMA is prone to consuming more energy per round compared to PEGASIS-TDMA. This is because PEGASIS-CDMA requires nodes to communicate over long distances because of the binary tree hierarchy (Kumarawadu *et al*., 2008).

Rank based data gathering algorithm: This algorithm begins with a group of nodes that connect with one another if and only if the node is within its transmission range. First the nodes generate an identifier (some integer), in sequential order. The nodes are then assigned a unique rank by using a random generator that is based on the current system time in milliseconds[1].During each round of execution, a new rank is given to each node. The coordinates of the nodes are generated also using the same method except the time in milliseconds is multiplied by the XMAX and YMAX preset values. The next step in the simulation after the nodes are generated is placing an edge between nodes that are within each other's transmission range. Using Euclidean Distance formula the distance between two nodes is found out. If the distance was less than or equal to the preset transmission range, then an edge was placed between the first node and second node. After the edges were placed within the graph, an adjacency list was formed using a TreeMap data structure. Figure 1 represents a snapshot of a network topology of 16 sensor nodes (the identifier is a unique character label inside the circle) and their rank values (indicated outside the circle).

This provides us with a rank between 0 and 1

Detection of associate and leaf nodes: The assembling of the graph is now completed and the Rank Based Data Gathering Algorithm steps are to be followed to find the associate and leaf nodes:

Step 1: A node becomes an Associate Node if it has the highest rank among all its neighbors.

Step 2: For each node v that has not been selected as an Associate Node in Step 1,if there exists a neighboring node u that has been selected as an Associate Node in Step 1, then v becomes a leaf node for node u (Fig. 2).

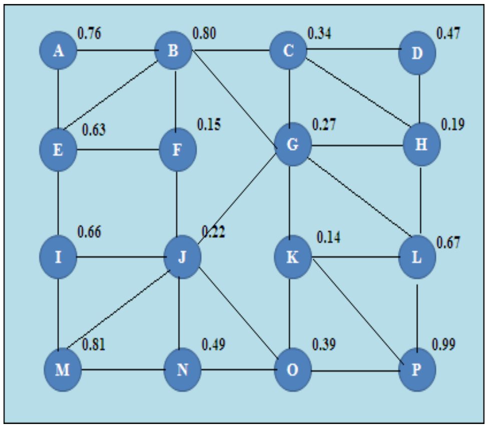
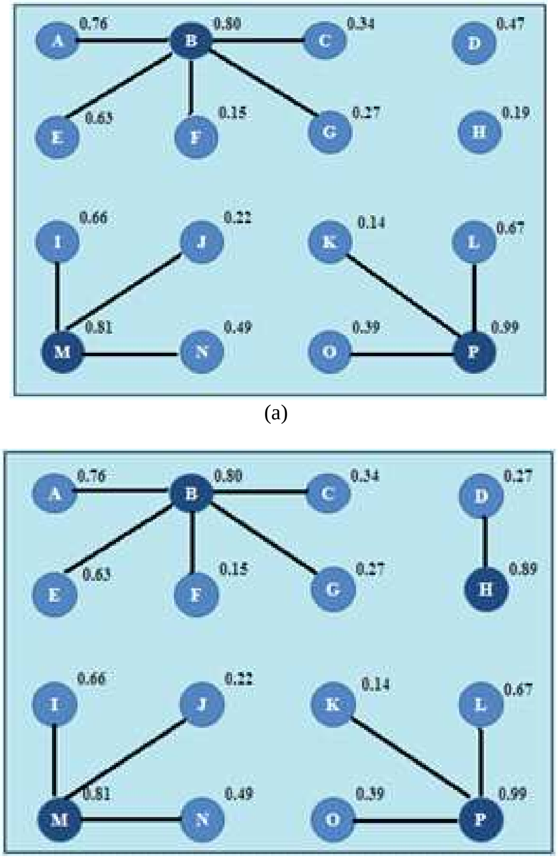


Fig. 1: Snapshot of the network topology graph

```
Input: snapshot of the network graph g = (v, e); where
V is the set of vertices and E is the set of edges
Auxiliary variables and initialization:
Node-list, adjacency-list of neighbors for each node,
Canbeassociate = true
Associate-node-list = φ leaf-node-list = Φ
Begin selection-associate and leaf nodes
For every vertex u ∊ node-list do
Rank (u) = random number assigned from 0-1
 End for
 For every vertex u ∊ node-list do
canbeassociate = true
For every vertex v ∊ adjacency-list (u) do
 If (rank (v)> rank (u)) then
 Canbeassociate = false
  Else
  Leaf-node-list= leaf-node-list U{v}
  End if
 End for
 if (canbeassociate = true) then
 Associate-Node-List = Associate-Node-List U {u}
  end if
 end for
 for every vertex u ∊ Node-List and u ∊Associate-
 Node-List
 and u ∊ leaf-node-list do
        for every vertex v ∊ Leaf-Node-List do
         if (rank (v)>rank (u)) do
               rank (u) = random number between max
 (Leaf-node-list) and 1
         end if
  end for
  Associate-node-list= associate-node-List U {u}
 end for
```

Fig. 2: Algorithm for the detection of associate nodes and leaf nodes



(a)

(b)

Fig. 3: Execution of RBDG algorithm

Step 3: If a node cannot be assigned as a leaf node for any Associate Node selected in Step 1, then the node's rank value is to be increased using random generator as an Associate Node and is added to the list of Associate Nodes (Fig. 3).

Step 4: A complete graph (link between every pair of nodes) is formed involving the Associate Nodes formed from Steps 1 and 3.

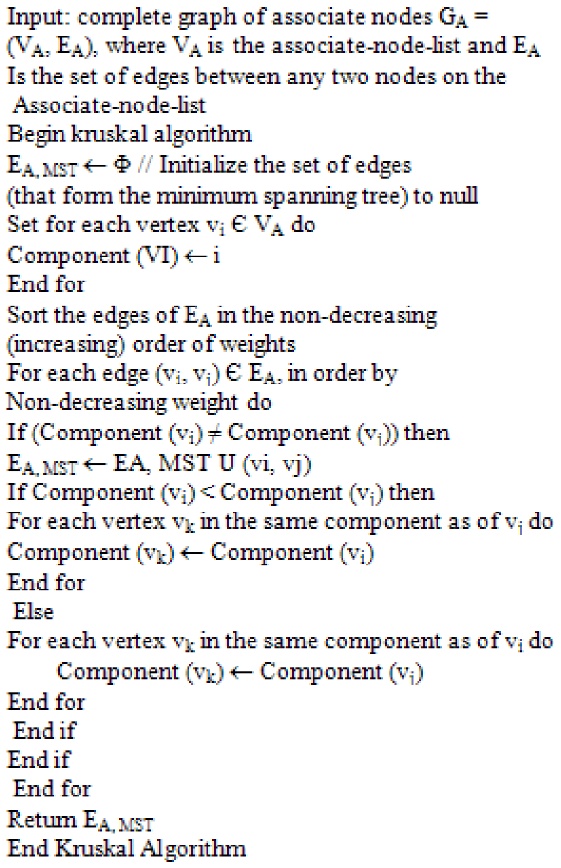
Step 5: Kruskal's algorithm (MST) is run on the complete graph formed in Step 4.

Step 6: The MST formed in Step 5 is transformed to a rooted directed DG tree with the root being the Associate Node with the largest available energy.

The associate nodes are generated if a node has the highest rank among its neighbors. For every node, *U*, it is assumed that it can be an associate node and collect all of its neighbors. For every neighbor node, *V*, it is verified whether or not its neighbor has a higher rank. If this is true, then the node, U, is no longer considered to be an associate node. If U can be considered as an associate node, then U is added to the associate node list.

The leaf nodes are generated if a node is adjacent to an associate node and it's rank must be lower than one of its adjacent nodes to be considered a leaf node. For every node, I, it collects its neighbor nodes and assume that it haven't yet discovered a leaf node. For all of I's neighbors the algorithm checks to see if the neighbor is in the associate list, if true, then it considers I as being a leaf node. Finally, it is placed those nodes without an associate node as a neighbor, into the associate list. it simply applies this method by adding a node to the list if it has not been placed in the leaf or associate node list.

Construction of the data gathering tree: After the simulator has accumulated all of the associate nodes, it forms a complete graph among the nodes. The construction of the complete graph has a few similarities to the construction of the original graph. Except this time the simulator doesn't worry about the transmission range when creating edges.



```
Input: complete graph of associate nodes $G_A$ =
($V_A$, $E_A$), where $V_A$ is the associate-node-list and $E_A$
Is the set of edges between any two nodes on the
 Associate-node-list
Begin kruskal algorithm
$E_{A,MST}$ ← Φ // Initialize the set of edges
(that form the minimum spanning tree) to null
Set for each vertex $v_i$ Є $V_A$ do
Component (VI) ← i
End for
Sort the edges of $E_A$ in the non-decreasing
(increasing) order of weights
For each edge ($v_i$, $v_j$) Є $E_A$, in order by
Non-decreasing weight do
If (Component ($v_i$) ≠ Component ($v_j$)) then
$E_{A,MST}$ ← EA, MST U (vi, vj)
If Component ($v_i$) < Component ($v_j$) then
For each vertex $v_k$ in the same component as of $v_j$ do
Component ($v_k$) ← Component ($v_i$)
End for
 Else
For each vertex $v_k$ in the same component as of $v_i$ do
      Component ($v_k$) ← Component ($v_j$)
End for
 End if
End if
 End for
Return $E_{A,MST}$
End Kruskal Algorithm
```

Fig. 4: Kruskal algorithm

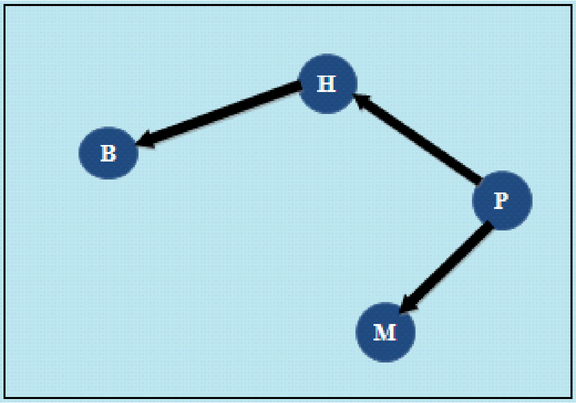


Fig. 5: The rooted directed tree

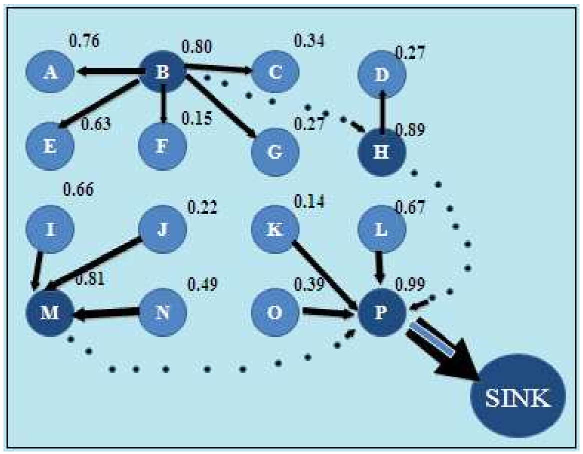


Fig. 6: Rank based data gathering tree

It automatically connects every node to another node in the associate node list. The other difference is to be kept a list of the edges and their corresponding weight, for help when forming a minimum spanning tree in the near future. Upon completion of the formation of the complete graph, a minimum spanning tree algorithm must be run in order form a tree. Although Prim's algorithm has low memory usage, Kruskal's (Fig. 4). can be faster in terms of computations in limited cases.

Following the formation of a spanning tree, next a breadth first search is run on the given tree. When executing the breadth first search a root node has to be found first, which will be known as the sensor node with the highest rank among the associate nodes, which is also the highest ranked node out of the entire network. This method is known as RDT (Rooted Directed Tree). As the BFS iterates, we keep track of a parent and it's downstream nodes (Fig. 5).

When the BFS is completed, now maintain a list is maintained an upstream node and it's downstream children, which includes it's leaf nodes from the decision phase and its downstream nodes from the MST. The RBDG is now completed when the newly birthed rooted directed tree is formed from the BFS. Now the rooted directed tree is applied to the entire network and forms the Rank Based Data Gather Tree, as shown below (Fig. 6).

RESULTS

Simulation Results: The simulation of the RBDG was carried out on a discrete-event simulator. This simulator has been used to successfully report simulation result for data gathering in sensor network. The size of the network is 100×100m. There are 100 sensor nodes that are randomly distributed throughout the network. Later, a network of just 60 nodes is also simulated. The sink node is located outside of the sensor network at the location (50, 300). Each node is assumed to be able to allow data communication between it and its downstream nodes if any. As the energy consumption model is implemented, a node takes into account the distance between it and the node it must stream data to, allowing for a more accurate data communication sequence. With this simulator, the execution of RBDG with a transmission range of 20-60m is carried out with increments of 5. We've conducted the simulations for both TDMA and CDMA systems. 100,000 trials of the RBDG within a CDMA and TDMA system and the same for LEACH and PEGASIS are simulated. Each node has been supplied with an initial energy of 1 Joule.

In a TDMA system, due the time slot variance, simultaneous communication among the clusters cannot occur. This also means that an upstream associate node cannot communicate with more than one of its downstream associate node. Before data communication occurs, each receiver advertises a distinct time slot for each of its senders. It is also assumed that each associate node receives data from its leaf nodes simultaneously before sending data to an upstream associate node. For the energy consumption model the first-order radio model will be used (Heinzelman, *et al*., 2000). The energy expended by a radio to run the transmitter or receiver circuitry is E_{elec} = 50 nJ bit $^{-1}$ and = 100pJ bit $^{-1}$ m2 for the amplifier. The radios are turned off when a node wants to avoid receiving unintended transmissions. The energy lost in transmitting a k-bit message over a distance d is given by: ETX (k, d) = E_{elec} * k + $_{amp}$*k* d^2. The energy lost in receiving a k-bit message is ERX (k) = E_{elec}* k.

Impact of the Transmission Range: when there is a network of at least 15 sensor nodes, then the sensor network is considered to be completely covered. When simulated with sensor nodes with high transmission range, it is noticed that the number of rounds before a node fail decreased drastically. This is due to the small amount of associate nodes and large amount of data aggregation done by the associate nodes. When simulated with a small transmission range, the network lifetime increases significantly in both data gathering tree. The small amount of leaf nodes per associate node is the cause of this. In sensor networks, the energy consumed for communication is much higher than that for sensing and computation (Zheng and Jamalipour). When the network contains a small amount of associate nodes, then the energy consumed for communication decreases for all associates except for the root associate node. Likewise with the rank and energy based data gathering tree, the network lifetime increases as the transmission range decreases. The rank and energy based version displayed astonishing results with the lifetime of the sensor network when compared to the original rank based protocol.

When simulated with a transmission range of 20 meters to 35 meters, the network lifetime was higher than other simulations. The network lifetime of the RBDG is lower for the simulations run with a transmission range of 40 meters and over. As we simulated a transmission range of 20-25 meters the average amount of leaf nodes decreased. The lack of leaf nodes also leads to the lack of energy consumed per round of data communication.

The height of the tree is dependent on the amount of leaf nodes that are selected during each round. When the transmission range was simulated at 20-30 meters, the height decreased slightly which is caused because they are more leaf nodes on a particular level. Based on simulation results, the rank based algorithm is best used when the transmission range of each node is lower than 35 meters.

Comparison of RBDG with PEGASIS and LEACH: The methodology used to select the associate nodes and leaf nodes equalizes the chance for a node to do data aggregation and forwarding packets to other sensor nodes. The results below, in Fig. 7, display a sensor network of 100 nodes, in a field size of 100m by 100m with a transmission range of 30 and simulated over 100,000 trials. This is one of the main differences between LEACH and PEGASIS and RBDG. The results greatly show how such a small difference on choosing leader nodes can reflect a network lifetime and delay of a tree.

The data reflects a difference of over just over 40% between RBDG and PEGASIS (Meghanathan, 2009). The data also shows how the strategy for LEACH is a failing one when compared to other data gathering algorithms.

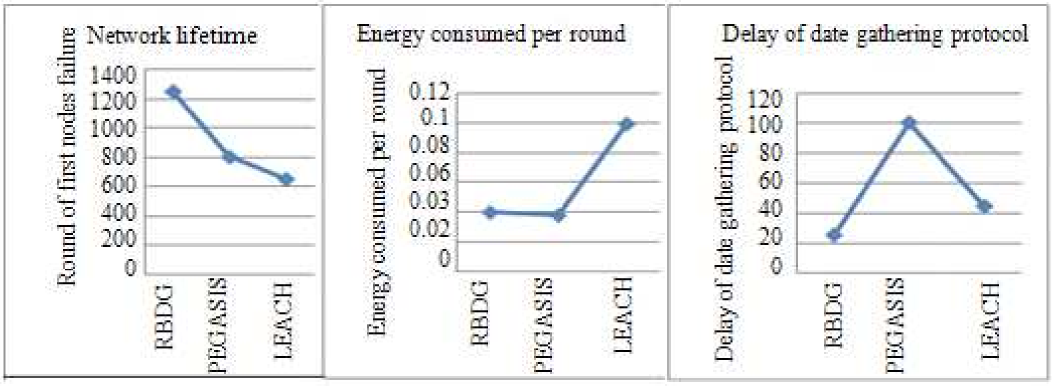


Fig. 7: Comparison of RBDG, PEGASIS and LEACH. Field size: 100m by 100m, Transmission range: 25 meters, simulation trials: 100,100

RBDG and PEGASIS-TDMA resemble an almost equal consumption of energy when compared to each other. With the given data it can also be said that a node, on average, has to gather information from only 3 or less nodes. This is because, PEGASIS-TDMA allows the downstream nodes to send to their upstream nodes and each node has only one upstream parent in the DG chain. Since PEGASIS-TDMA consumes less than 1% more energy than RBDG, we can see where the conclusion of 3 or less leaf nodes originates from (Meghanathan, 2009). PEGASIS-TDMA has a chain gathering strategy that has proven to be insufficient when compared to other DG algorithms. The delay is at the maximum (maximum delay = *n* number of nodes) point with the PEGASIS-TDMA algorithm. When comparing RBDG to LEACH, the data shows a difference of almost 50% with both algorithms (Meghanathan, 2009).Because of the delay in PEGASIS-TDMA and of the energy consumption in LEACH-TDMA, the figure shows both RBDG has a lower rate of energy consumption per round of each delay.

DISCUSSION

In this study it uses rang values to form a complete graph, then it forms a Rooted Directed Tree (RDT) after an implementation of Kruskal's Minimum Spanning Tree algorithm and the Breadth First Search algorithm. Finally it provides a better outcome in terms of lifetime and delay per round for TDMA.

CONCLUSION

The high-level contribution of this study is the development of a rank based data gathering algorithm. After over 100, 000 trials, it is observed that the network lifetime with RBDG is 3 and 2.2 times more than that incurred for LEACH and PEGASIS respectively. The delay per round of data gathering is significantly lower compared to that of PEGASIS and LEACH. The energy consumed per round of data gathering for both RBDG is less than half of that incurred with PEGASIS and LEACH. Compared with LEACH and PEGASIS, RBDG is fair with respect to the usage of the nodes and this reflects in the relatively larger value for the network lifetime, measured as the round of first node failure due to exhaustion of energy reserves. Overall, the rank-based data gathering algorithm and its energy entity can be a significant addition to the list of data gathering algorithms that can simultaneously maximize the network lifetime as well as minimize the delay per round of data gathering. In future this study has to concentrate on increasing the transmission range and energy consumption for efficient data gathering.

REFERENCES

Cauligi, S.L., S. Raghvendra, 2002. PGASIS. Power-Efficient Gathering in Sensor Information System. Proceeding of the IEEE Aerospace Conference, Mar. 9-16, IEEE Xplore Press, USA, pp: 31125-3113. DOI: 10.1109/AERO.2002.1035242

Heinzelman, W., A. Chandrakasan and H. Balakrishnan, 2000. Energy-efficient communication protocol for wireless microsensor networks. Proceedings of the 33rd Hawaii International Conference on System Sciences, (HICSS), IEEE Computer Society Washington, DC, USA, pp: 8020-8020.

Jung, S.M., Y.-J. Han, T.-M. Chung, 2007. The concentric clustering scheme for efficient energy consumption in the pegasis. Proceeding of the 9th International Conference on Advanced Communication Technology, Feb. 12-14, IEEE Xplore Press, Gangwon-Do, pp: 260-265. DOI: 10.1109/ICACT.2007.358351

Kumarawadu, P., D.J. Dechene, M. Luccini and A. Sauer, 2008. Algorithms for node clustering in wireless sensor networks. Proceeding of the 4th International Conference on Information and Automation for Sustainability, Dec. 12-14, IEEE Xplore Press, USA, pp: 295-300. DOI: 10.1109/ICIAFS.2008.4783999

Lee, J.-E., K. Kim,2008. Diamond-shaped routing method for reliable data transmission in wireless sensor networks. Proceeding of the IEEE International Symposium on Parallel and Distributed Processing with Applications, Dec. 10-12, IEEE Xplore Press, Sydney, NSW, pp:799-801. DOI: 10.1109/ISPA.2008.111

Lindsey, S., C. Raghavendra and K.M. Sivalingam, 2001. Data gathering in sensor networks using the energy*delay metric. Proceedings of the 15th International Parallel and Distributed Processing Symposium, April 23-27, IEEE Xplore Press, USA, pp: 2001-2008. DOI: 10.1109/IPDPS.2001.925196

Lindsey, S., C. Raghavendra and K.M. Sivalingam, 2002. Data gathering algorithms in sensor networks using energy metrics. IEEE Trans. Parallel and Distri. Sys., 13: 924-935. DOI: 10.1109/TPDS.2002.1036066

Meghanathan, N., 2009. Grid Block Energy Based Data Gathering Algorithm for Lower Energy*Delay and Longer Lifetime in Wireless Sensor Networks. Pub Zone Scientific Publication.

Meghanathan, N.,2 010. An algorithm to determine energy-aware maximal leaf nodes data gathering tree for wireless sensor networks. J. Theoretic. Applied Inf. Tech., 15: 96-107.

Muruganathan, S.D.; Ma, D.C.F.; Bhasin, R.I.; Fapojuwo, A.O., 2005. A centralized energy-efficient routing protocol for wireless sensor networks. Proceedings of IEEE Communications Magazine, March 2005, IEEE Xplore Press, Canada , pp: 8-13. DOI: 10.1109/MCOM.2005.1404592

Sivalingam, K.M., J. Chen, P. Agrawal and M. Srivastava, 2000. Design and analysis of low-power access protocols for wireless and mobile ATM networks. ACM/Baltzer Wireless Networks. 6: 73-87. DOI: 10.1023/A:1019152506607

Tabassum, N., Q.E.K. Mamun and Y. Urano, 2006. COSEN. A chain oriented sensor network for efficient data collection. Proceeding of the 3rd International Conference on Information Technology: New Generations, April 10-12, IEEE Xplore Press, Las Vegas, NV, pp: 262-267. DOI: 10.1109/ITNG.2006.44

Viterbi, A.J.,1995. CDMA: Principles of Spread Spectrum Communication. 1st Edn., Addison-Wesley Pub. Co., Reading, Mass, pp: 245. ISBN: 0201633744.

Zheng, J. and A. Jamalipour, 2009. Wireless Sensor Networks: A Networking Perspective. 1st Edn., Wiley-IEEE, Piscataway, N.J., pp: 489. ISBN: 9780470167632 0470167637