A Trivial and Consistent Routing within Network Aggregation in Wireless Sensor Networks

C. Pravallika[1], T. Venkata Naga Jayudu[2]

[1]PG student, Department of CSE, Intell Engineering College, Anantapur, AP, India

[2]Assistant Professor, Department of CSE, Intell Engineering College, Anantapur, AP, India

Abstract: ***Wireless sensor networks (WSNs) will be increasingly deployed in different classes of applications for accurate monitoring. Due to the high density of nodes in these networks, it is likely that redundant data will be detected by nearby nodes when sensing an event. Since energy conservation is a key issue in WSNs, data fusion and aggregation should be exploited in order to save energy. In this case, redundant data can be aggregated at intermediate nodes reducing the size and number of exchanged messages and, thus, decreasing communication costs and energy consumption. In this work we propose a novel Data Routing for In-Network Aggregation, called DRINA, that has some key aspects such as a reduced number of messages for setting up a routing tree, maximized number of overlapping routes, high aggregation rate, and reliable data aggregation and transmission. The proposed DRINA algorithm was extensively compared to two other known solutions: the InFRA and SPT algorithms. Our results indicate clearly that the routing tree built by DRINA provides the best aggregation quality when compared to these other algorithms. The obtained results show that our proposed solution outperforms these solutions in different scenarios and in different key aspects required by WSNs.***

Keywords: In-Network Aggregation, Routing Protocol, Wireless Sensor Networks, Clusters, Hop Trees

1. Introduction

A Wireless Sensor Network (WSN) consists of spatially distributed autonomous devices that cooperatively sense physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants at different locations.WSN in its simplest form can be defined as a network of (possibly low-size and low-complex) devices denoted as nodes that can sense the environment and communication the information gathered from the monitored field through wireless links; the data is forwarded, possibly via multiple hops relaying, to a sink that can use it locally, or is connected to other networks (e.g., the Internet) through a gateway. Sensor nodes are energy-constrained devices and the energy consumption is generally associated with the amount of gathered data, since communication is often the most expensive activity in terms of energy. We shall consider two-tiered wireless sensor network with three main actors. Sensors are in charge of sensing data. The sink receives queries from users, contacts the inner network to get answers, and returns them to users. Storage nodes stores data from sensors and seek answers for queries from the sink.

A sensor network consists of multiple detection stations called sensor nodes, each of which is small, lightweight and portable. Every sensor node is equipped with a transducer, microcomputer, transceiver and power source. The storage nodes also brings many security challenges since the storage node stores the data received from sensors and serve as an important role for answering queries, they are more vulnerable to be compromised, mainly in a hostile environment. A compromised storage node causes threats to a sensor network.

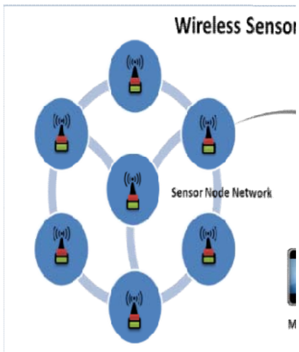


Figure 1: Architecture of wireless sensor network

2. Literature Survey

2.1 Routing techniques in wireless sensor networks: a survey

By J. Al-Karaki and A. Kamal (IEEE Transaction on Wireless Communications)

- Wireless sensor networks consist of small nodes with sensing, computation, and wireless communications capabilities. Many routing, power management, and data dissemination protocols have been specifically designed for WSNs where energy awareness is an essential design issue.
- Routing protocols in WSNs might differ depending on the application and network architecture. A survey of state-of-the-art routing techniques in WSNs.

3. Existing System

WSNs are data-driven networks that usually produce a large amount of information that needs to be routed, often in a multi hop fashion, toward a sink node, which works as a gateway to a monitoring center.

The solution to this problem was proposed by which is called as "InFRA and SPT Scheme". This scheme has two main drawbacks:

1. A possible strategy to optimize the routing task is to use the available processing capacity provided by the intermediate sensor nodes along the routing paths. This is known as data centric routing or in-network data aggregation.
2. Data aggregation is an effective technique for saving energy in WSNs. Due to the inherent redundancy in raw data gathered by the sensor nodes, in-networking aggregation can often be used to decrease the communication cost by eliminating redundancy and forwarding only smaller aggregated information.

Disadvantages:

- This will not do the maximization of data aggregation; and
- In this scheme, does not calculate the number hops and clustering.

4. Proposed System

The proposed scheme to Aggregation aware routing algorithms play an important role in event based WSN's the DRINA algorithm [1], a novel and reliable Data Aggregation Aware Routing Protocol for WSNs. In proposed DRINA algorithm was extensively compared to two other known routing algorithms [2], the InFRA and SPT, regarding scalability, communication costs, delivery efficiency, aggregation rate and aggregated data delivery rate.

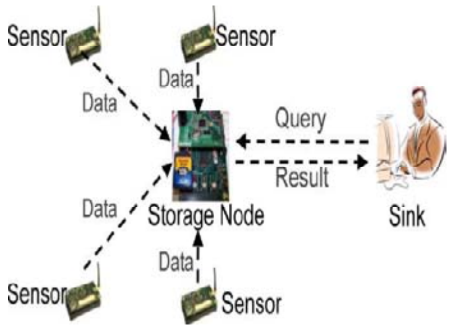


Figure 2: Two tiered sensor network

5. Implementation

1. DRINA: Data Routing for In-Network aggregation for WSN's.
2. Building the Hop Tree.
3. Cluster Formation.
4. Routing Formation and Hop Tree Updates.
5. Route Repair Mechanism.

5.1 DRINA: Data Routing for In-Network Aggregation for WSN's

The main goal of our proposed DRINA algorithm is to build a routing tree with the shortest paths that connect all source nodes to the sink while maximizing data aggregation [3], [4]. The proposed algorithm considers the following roles in the routing infrastructure creation:

- *Collaborator:* a node that detects an event and reports the gathered data to a coordinator node;
- *Coordinator:* a node that also detects an event and is responsible for gathering all the gathered data sent by collaborator nodes, aggregating them and sending the result toward the sink node;
- *Sink:* a node interested in receiving data from a set of coordinator and collaborator nodes;
- *Relay:* a node that forwards data toward the sink.

5.2. Building the Hop Tree

The DRINA algorithm can be divided into three phases. In this phase, the distance from the sink to each node is computed in hops. This phase is started by the sink node sending, by means of a flooding, the Hop Configuration Message (HCM) to all network nodes. The HCM message contains two fields: ID and HopToTree, where ID is node identifier that started or retransmitted the HCM message and HopToTree is the distance, in hops, by which an HCM message has passed.

Algorithm 1: Hop Tree Configuration Phase

1. Node sink sends a broadcast of HCM messages with the value of HopToTree = 1.
2. **For each u ∈ R do**
  If HopToTree (u) > HopToTree (HCM) *and*
  First Sending (u) **then**
  Next Hop ← IDHCM;
  HopToTree ← HopToTreeHCM + 1;
3. Node u updates the value of the ID field in the message HCM.
  IDHCM ← IDu;
4. Node u updates the value of the HopToTree field in the message HCM
  HopToTreeHCM ← HopToTree;
5. Node u sends a broadcast message of the HCM with the new values;
  FirstSendingu ← false;
6. **End**
7. **Else**
  Node u discards the received message HCM;
8. **End**
9. **End**

5.3. Cluster Formation

When an event is detected by one or more nodes, the leader election algorithm starts and sensing nodes will be running for leadership (group coordinator) [5]. For this election, all sensing nodes are eligible.

- If this is the first event, the leader node will be the one that is closest to the sink node. Otherwise, the leader will be the node that is closest to an already established route;
- In the case of a tie, i.e., two or more concurrent nodes have the same distance in hops to the sink (or to an established route); the node with the smallest ID maintains eligibility. Another possibility is to use the energy level as a tiebreak criterion.
- At the end of the election algorithm only one node in the group will be declared as the leader (Coordinator). The remaining nodes that detected the same event will be the Collaborators.

- The Coordinator gathers the information collected by the Collaborators and sends them to the sink.

Algorithm 2: Cluster formation and leader election

1. **Input:** S
   S - Set of nodes that detected the event.
2. **Output:** u
   A node of the set S is elected leader of the group.
3. **For each** u ∈ **S do**
   Role $_u$ ← coordinator;
   Node u *sends message MCC in broadcast*
   Announcement of event detection;
   Nu *is the set of neighbors of node* u ∈ **S**
4. Node u retransmits the MCC message received from node w;
   For each w ∈ N $_u$ do
   If HopToTree (u) > HopToTree (w) **then**
   Role $_u$ ← collaborator;
5. **End**
6. Node u retransmits the MCC message received from node w;
   Else if HopToTree (u) = HopToTree (w) ^
   ID (u) > ID (w) **then**
   Role $_u$ ← collaborator;
7. **End**
8. **Else**
   Node u discards the MCC message received from w;
9. **End**
10. **End**
11. **End**
12. **End**

5.4. Routing Formation and Hop Tree Updates

The elected group leader, starts establishing the new route for the event dissemination. The Coordinator sends a route establishment message to its Next Hop node. When the Next Hop node receives a route establishment message, it re-transmits the message to its Next Hop and starts the hop tree updating process. These steps are repeated until either the sink is reached or a node that is part of an already established route is found [6]. The routes are created by choosing the best neighbor at each hop.

Algorithm 3: Route establishment and hop tree update

1. Leader node v of the new event sends a message REM to its NextHop $_v$;
2. **Repeat**
3. u is the node that received the REM message that was sent by node v.
   If u = NextHop $_v$ **then**
   HopToTree $_u$ ← 0;
   Node u *is part of the new route built*
   Role u ← Relay;
   Node u sends the message REM to its NextHop $_u$;
   Node u broadcasts the message HCM with the value of HopToTree = 1; Nodes that receive the HCM message sent by node u.
4. **End**
5. **Until** Find out the sink node or a node belonging to the routing structure already established.
6. **Repeat**
7. sons u is the number of descendants of u
   If sons u > 1 **then**
8. Aggregates all data and sends it to the NextHop u;
   If Role u = Relay **then**
   Execute the mechanism
9. **End**
10. **End**
11. **Else**
    Send data to NextHop u;
12. **If** Role u = Relay **then**
    Execute the mechanism
13. **End**
14. **End**
15. **Until** The node has data to transmit/retransmit;

5.5. Route Repair Mechanism

The route created to send the data toward the sink node is unique and efficient since it maximizes the points of aggregation and, consequently, the information fusion. However, because this route is unique, any failure in one of its nodes will cause disruption, preventing the delivery of several gathered event data. Possible causes of failure include low energy, physical destruction, and communication blockage [7], [8].

- Some are based on periodic flooding mechanisms, and rooted at the sink, to repair broken paths and to discover new routes to forward traffic around faulty nodes.
- This mechanism is not satisfactory in terms of energy saving because it wastes a lot of energy with repairing messages.
- DRINA algorithm offers a piggybacked, ACK-based, route repair mechanism, which consists of two parts: failure detection at the NextHop node, and selection of a new NextHop.

6. Conclusion

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

In DRINA algorithm, a novel and reliable Data Aggregation Aware Routing Protocol for WSNs. Our proposed DRINA algorithm was extensively compared to two other known routing algorithms, the InFRA and SPT, regarding scalability, communication costs, delivery efficiency, aggregation rate and aggregated data delivery rate. By maximizing the aggregation points and offering a fault tolerant mechanism to improve delivery rate, the obtained results clearly show that DRINA outperformed the InFRA and SPT algorithms for all evaluated scenarios. In proposed algorithm has some key aspects required by WSNs aggregation aware routing algorithms such as a reduced number of messages for setting up a routing tree, maximized number of overlapping routes, high aggregation rate, and reliable data aggregation and transmission.

7. Future work

Spatial and temporal correlation of the aggregated data will also be taken into consideration as well as the construction of a routing tree that meets application needs. We also plan to modify DRINA algorithm to stochastically select nodes that will be part of the communication structure. The goal is to find a balance between the overhead and the quality of the routing tree. In addition, new strategies will be devised to control the waiting time for aggregator nodes based on two criteria: average distance of the event coordinators, and spatial and semantics event correlation.

References

[1] Leandro Villas, Azzedine Boukerche1, Heitor S. Ramos, "DRINA: A Lightweight and Reliable Routing Approach for in-Network Aggregation in Wireless Sensor Networks" *Wireless Sensors, 2012.*

[2] E. F. Nakamura, A. A. F. Loureiro, and A. C. Frery, "Information fusion for wireless sensor networks: Methods, models, and classifications," *ACM Computing Surveys*, vol. 39, no. 3, pp. 1/9–55, 2007.

[3] H. S. Abdel Salam and S. Olariu, "A lightweight skeleton construction algorithm for self-organizing sensor networks." in *ICC*. IEEE, 2009, pp. 1–5.

[4] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, "In-network aggregation techniques for wireless sensor networks: a survey," *Wireless Communications, IEEE*, vol. 14, no. 2, pp. 70–87, April 2007.

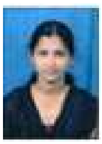
[5] A. Boukerche, *Algorithms and Protocols for Wireless Sensor Networks*. Wiley-IEEE Press, 2008.

[6] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 6–28, Dec. 2004

[7] F. Hu, X. Cao, and C. May, "Optimized scheduling for data aggregation in wireless sensor networks," in *ITCC '05: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 557–561

[8] L. A. Villas, A. Boukerche, R. B. Araujo, and A. A. Loureiro, "A reliable and data aggregation aware routing protocol for wireless sensor networks," in *Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, ser. MSWiM '09. New York, NY, USA: ACM, 2009, pp. 245–252.

Author profile



C. Pravallika received the B.Tech in Computer Science Engineering in 2010 from Moula Ali College Of Engineering and Technology, JNTUA University, Anantapur, Andhra Pradesh, India. Currently she is pursuing her M.Tech in Computer Science Engineering in Intell Engineering College, Anantapur, an Andhra Pradesh, and India.

T. Venkata Naga Jayudu, received his B.Tech degree in Computer Science and Information Technology from Jawaharlal Nehru Technological University, Hyderabad, India, in 2005; M.Tech in Computer Science from Jawaharlal Nehru Technological University, Anantapur, India, in 2011. He is Assistant Professor at Intell Engineering College, Anantapur, India. He guided several projects for both UG & PG students. His interesting research area is Mobile Ad-Hoc Networks, wireless sensor networks, Network Security and Operating Systems.