International Journal of Information Technology and Knowledge Management
July-December 2008, Volume 1, No. 2, pp. 449-462

DESIGN ISSUES ON TREE BASED AGGREGATION ALGORITHMS IN WIRELESS SENSOR NETWORKS

A. Sivagami, K. Pavai, D. Sridharan & S. A. V. Satya Murty

Abstract

Wireless Sensor Networks (WSN) consists of tiny sensor nodes which are having limited CPU, memory, battery and communication capabilities. WSN differs from conventional wireless networks in several ways such as sensor nodes have severe energy constraints produces redundant low-rate data traffic, and many-to-one flows. The end-to-end routing schemes that have been proposed in the literature for mobile ad-hoc networks are not appropriate under these constraints. Hence, it is essential to have data-centric technologies that perform in-network aggregation which gives energy-efficient dissemination. We focus on data aggregation problems in energy constrained sensor networks. The main goal of data aggregation algorithms is to gather and aggregate data in an energy efficient manner so that the network lifetime can be increased. In this paper we present an elaborate survey on different data aggregation algorithms based tree architecture and compare them in terms of lifetime, latency and data accuracy. Also we present the different network issues such as reliability and security while performing aggregation.

Keywords: Wireless Sensor Networks, Data aggregation, Tree based algorithms.

1. Introduction

Wireless sensor networks (WSNs) consist of a large numbers of inexpensive sensor nodes which are deployed densely in harsh and inaccessible terrains like gigantic mountains, valleys, oceans, deep forest etc [1]. These nodes have capabilities for sensing the parameter like temperature, humidity, vibration, etc., processing the sensed data and transmitting it to the base station either by direct link or by multihop communication. Though these nodes have no fixed topology, they can form a multihop self-organized network by sending beacons through wireless channels and configure themselves in to adhoc wireless network. Sensor networks can be used in plenty of applications like battle field and enemy terrain surveillance and reconnaissance in military, patient health monitoring in medical application, environment and habitat monitoring application, building infrastructure monitoring, building a smart and automated home and office, search and rescue operations during emergency conditions like earthquake, flood, tsunami, etc.[1] [2] In this paper, the various issues while aggregating the information, the various techniques for aggregation and their strength and weakness in terms of energy, data freshness and data accuracy are discussed briefly.

Sensor nodes are battery driven and hence operate on an extremely frugal energy budget. It is impracticable to replace the battery for the network with thousands of physically embedded nodes. The network lifetime can be maximized by incorporating energy-awareness into every stage of wireless sensor network design and operation. [3] In a mote, the battery power is utilized by the computing sub system (MCU), sensing subsystems, and communication sub system. The microcontroller (MCU) is responsible for control of the sensors, execution of communication protocols, and signal processing on the gathered sensor data. The sensor node radio is responsible for wireless communication with neighboring nodes and the outside world. Sensor transducers translate the physical phenomena to electrical signals. There are several sources of power consumptions in the sensor including 1. sampling 2. signal conditioning and 3. analog to digital conversion. All these sub systems consume power. From the data sheet of many commercially available motes, it is understood that the power used by the microcontroller and the sensing sub systems is less compared to that used by communication unit. In order to increase the life time of the network, it is good to design an algorithm that reduces the number of transmissions. The data aggregation algorithm can reduce the number of transmission by allowing the aggregator node to transmit only the required data, not the redundant information.

This paper is organized as follows: section 2 discuss about the data aggregation, aggregation operators and the method of aggregation. Section 3 describes the parameters considered for analyzing the performance of the aggregation algorithms and section 4 discuss about network architecture, various tree based protocols in detail. Finally section 5 describes the network issues like reliability and security during aggregation.

2. Data Gathering & Data Aggregation

Data gathering is defined as the systematic collection of sensed data at predefined time interval from the multiple sensor nodes and transmitted to the base station for further processing. Since sensor nodes are energy constrained, it is inefficient for all the sensors to transmit the data directly to the base station due to the following reasons. Data generated from neighboring sensors is often redundant and highly correlated. Hence these sensors report the same data about an event and hence it is not needed to transmit the multiple copies of the same information. In addition, the amount of data generated in large sensor networks is usually enormous for the base station to process. Hence, we need methods for combining data into high quality information at the sensors or intermediate nodes which can reduce the number of packets transmitted to the base station resulting in conservation of energy and bandwidth. This can be accomplished by data aggregation. *Data aggregation* is defined as the process of aggregating the data from multiple sensors to eliminate redundant transmission and provide fused information to the base station. Data aggregation usually involves the fusion of data from multiple sensors at intermediate nodes and transmission of the aggregated data to the base station (sink).

2.1. Data Aggregation Methods

In energy constrained wireless sensor nodes, the Energy efficiency can be achieved by using some of the in-network aggregation techniques. While forwarding the data, the information from various sources are combined along its path to the sink and this process is called In-network aggregation. The data aggregation operator is simple SUM, AVERAGE, MAXIMUM, MINIMUM and COUNT, etc to more complicated data aggregation methods like, MEDIAN, Wavelet Histogram.[1] The energy saving depends on the type of aggregation operator employed. For example, if the MAX operator is used for aggregation and the results in a single packet of same size as that of individual sensor readings. If the aggregation ratio is n: 1, then the energy saving will be n-fold. Suppose, the concatenate operator is used for aggregation, i.e, the individual sensor readings are appended by the aggregator node, the energy saving takes place only on medium access. The processing of data takes place in side the network, hence the aggregation process suppress the transmission of unwanted information. This increases the lifetime of the sensor nodes and hence the sensor network. The benefits of in-network aggregation is also extended to the sink by receiving less amount of useful information from the sensor sources and hence the sink can perform less processing and filtering to get useful information from these data by consuming less resources. Even though, aggregation reduces the energy consumption, it increases the delay of delivering the packet to the sink. This is due to the fact that each aggregator has to wait for a predefined time interval to collect data from its children. This leads to the delay in the delivery of the data and hence the sink may not get the fresh data. Hence there is a tradeoff between energy saving, data accuracy and freshness, i.e., the longer a node waits, the more readings it is likely to receive and therefore, the more accurate the information it sends out. On the other hand, waiting too long may result in stale data. Furthermore, if a node waits too long, it may interfere with the next "data wave".

By considering the above factors, there are three different methods of data aggregations namely, *Periodic simple*, *periodic per-hop*, and *periodic per-hop adjusted* are proposed. *Periodic simple* aggregation works by having each node wait a pre-defined period of time, aggregate all data items received, and send out a single packet containing the result. In *Periodic per-hop,* the aggregator nodes send out the fused packet as soon as they received packets from all their children or till the clock times out, the time out is set to data generation rate. *Periodic per hop adjusted* uses the same basic principle of *periodic per-hop* but schedules a node's timeout based on its position in the distribution tree. [4]

Ignacio Solis and Katia Obraczka *proposed the cascading timeouts* aggregation mechanism which falls on *periodic per-hop adjusted* category, in which the nodes timeout depends on the single hop delay. The performance of this is compared with the other methods in terms of energy, data accuracy and data freshness. They showed that the performance of the algorithm depends on the time out value of the aggregator. When compared to other existing *periodic per-hop adjusted* algorithms, cascading time out reduces the traffic by 6 times while maintaining the data accuracy and freshness. Also it presents other benefits such as not requiring clock synchronization among nodes and minimizing the timeout-scheduling overhead. It is a simple aggregation algorithm, minimum control overhead, no clock synchronization and independent of routing algorithms.

3. Performance Measures

Network lifetime, data accuracy, and latency are some of the important performance measures of data aggregation algorithms. The definitions of these measures are highly dependent on the desired application.

Network lifetime*:* Network lifetime is defined as the number of data aggregation rounds till the specified percentages of the total nodes dies and the percentage depends on the application. In some applications, simultaneous working of the all the sensor nodes is crucial hence the lifetime of the network is the number of rounds until the first node dies. The data aggregation methods ensure the uniform draining out power from all the sensor nodes which improves the energy efficiency of the nodes and enhances the lifetime of the entire network.

Latency*:* Latency is defined as the delay involved in data transmission, routing and data aggregation. It can be measured as the time delay between the data packets received at the sink and the data generated at the source node. It is also called as data freshness.

Data accuracy*:* The definition of data accuracy depends on the specific application for which the sensor network is designed. For instance, in a target localization problem, the estimate of target location at the sink determines the data accuracy. In general it is a measure of ratio of total number of readings received at the sink to the total number of readings generated.

Communication Overhead: It measures the communication complexity of the in-network aggregation algorithms. The control packets are transmitted between nodes to maintain the network. This packet will not relay any data to the sink and hence these are considered as overhead. The amount of overhead should be kept minimum, since these packets drains energy from the battery.

The performance of the data aggregation algorithms should be analyzed based on the above metrics. These metrics are interdependent on each other. Improving the energy efficiency is achieved by fusing more packets. This gives more accurate data but increases the delay.

Ramesh Rajagopalan *et al.* discussed in their survey paper about the various data aggregation algorithms and compared their performance in detail. According to them, the existing data aggregation algorithms are classified under the following categories: Network Architecture based, Network flow based, and QoS based. [5] In this paper we concentrate more on the aggregation schemes based on the architecture and we give brief research work going on tree based network.

4. Data Aggregation Based On the Network Architecture

The architecture of the network will have a great impact on the performance of the data aggregation. The network may have either a flat or hierarchical architecture.

4.1. Flat Networks

In the flat networks, all the nodes are having similar capabilities and responsibilities and plays the similar role. They are not having any fixed topology, the route to the sink from a data source is established by the routing protocol. Classic routing protocols which are based on the shortest path are not suitable for data aggregation paradigm. To promote data aggregation, the packet should be routed based on its content and the next hop candidate should be selected based on the most suitable aggregation points, data types, the priority of information, etc [6]. This type of routing is classified as data centric routing and it is interlinked with the data aggregation mechanism. The Sensor Protocol for Information via Negotiation (SPIN), DIRECTED DIFFUSUION and PEGASIS are the few examples in this group and are discussed in [5].

4.2. Hierarchical Networks

Flat network architecture will not be suitable if the size of the network is large. It is either Cluster based or Tree based. Hierarchical networks contain heterogeneous nodes, which can be either an end device/child or a cluster head/parent. In cluster based networks, the cluster head is selected based on the residual energy in it, the node with more energy is selected as cluster head. The end nodes are not transmitting the data directly to the sink, instead it transmits it to the cluster head which will be behaving as aggregator. The cluster heads will aggregate the data coming from its children and forward it to the sink as shown in the figure 1. The brief description about the following algorithms is given in [5]. *Low Energy Adaptive Clustering Hierarchy* (LEACH), *Hybrid Energy Efficient Distributed Clustering Approach* (HEED) and *clustered diffusion with dynamic data aggregation* (CLUDDA) fall in the category.

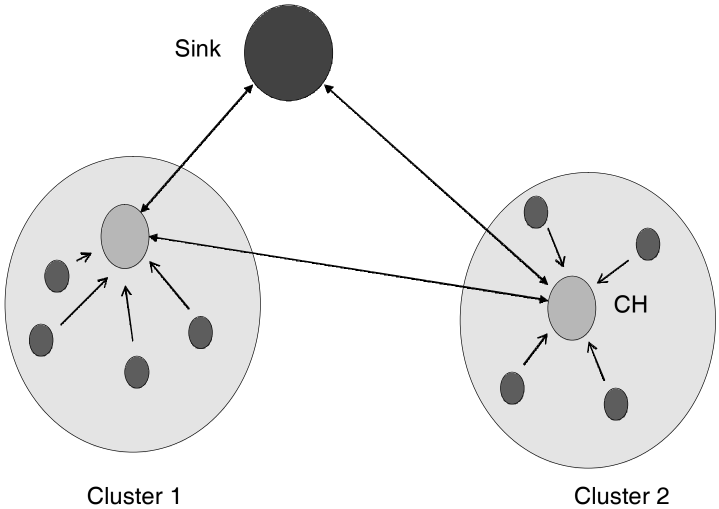

Figure 1: Cluster Routing

In the tree based networks, the sensor nodes are organized into a tree and the data aggregation takes place at the intermediate parent nodes along the tree as shown in figure 2.

In this figure 2, the nodes 4, 5, and 7 are the leave nodes which send the raw data to its parent and the nodes 6, 2 and 3 performs the role of parent which are responsible for data aggregation from its children and forwarding the aggregated information to the root. The main challenge in this type of data aggregation is the construction of efficient tree. In a network graph $G = (V, E)$ where V is the set of nodes and E is the set of edges that connect nodes which can communicate directly. Let $S_1, S_2 \ldots, S_k \in S$ be data sources and D be a sink node. For optimal aggregation, a minimal cost tree connecting nodes in S and node D with minimal number of edges should be found. This is the Steiner Tree

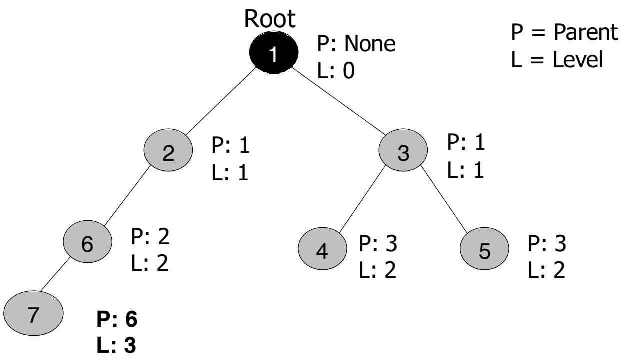

Figure 2: Data Aggregation Tree

problem, which is an NP-hard variant of the minimum spanning tree problem, some suboptimal aggregations are proposed in [7] In *Center at Nearest Source (CNS)*, the source which is nearest the sink acts as the aggregator, all other sources send their data directly to this source which then sends the aggregated information to the sink. The *Shortest Paths Tree (SPT*) allows each source to send its information to the sink along the shortest path and the overlapping paths are combined to perform aggregation. The *Greedy Incremental Tree (GIT)* builds the aggregation tree by allowing the source which is nearest to the sink to send its data via shortest path. Then the next source nearest to the tree is allowed to join the tree and the entire tree is constructed. In the following paragraphs, we discuss about the various tree-based protocols available.

4.2.1. TAG : A Tiny Aggregation Service for Adhoc Sensor Networks

It is the very first algorithm proposed by Madden et al in 2004 and is more efficient in terms of energy. The tree is constructed by the root node which sends the broadcast message with level 0 and its sensor Id. All nodes hearing the message increase the level field, attach their id and rebroadcast it again. They select the source of the message as their parent. The process continues down the tree. It can be used as periodic monitoring or query driven.

In periodic monitoring mode, the root sends the query. The child nodes send their current aggregated value and rebroadcast the query to the next level. Now the root receives the information from the first level child nodes. This process continues until, the root receives the packets from the last level.

The TAG offers a lot of advantages- saves energy, minimizes the number of messages transfer, use of epoch allows nodes to sleep during idle time thereby saving energy. [8]

4.2.2. TiNA: Temporal Coherency Aware in-Network Aggregation

The approach is to send the data only when there is a significant change in the data value in the adjacent readings over time. The concept of epoch as in TAG is also used here for synchronizing the receipts of the packets from the child nodes and sending the aggregate. A data value can be ignored if the variation from the previous value is within the specified range called filters. This needs higher memory requirements at each node because they need to store the intermediate results of the child nodes, (partial aggregation). The advantage is the significant reduction in number of messages over TAG [9].

4.2.3. DQEB - Dynamic query-tree Energy Balancing Protocol

The DQEB [10] protocol is an energy balanced protocol by dynamically modifying the tree structure based on the energy left at nodes. In this approach the nodes are organized into clusters with cluster heads. Each node is assigned a weight which goes on increasing with decreasing lifetime or energy. As the energy decreases, it is wiser to move the node down the tree i.e. turn it in to a leaf node so that the tree does not get disconnected. The energy cost depends on the number of leaf and non leaf nodes and the energy remaining at the node. Whenever the weight of a non leaf node goes down a threshold, the coordinator node asks all its child for alternating parents, then using a greedy approach it selects the alternating parents for all its children and itself becomes a leaf node. Since the nodes with less energy has become the leaf node it will live a little longer as now it only has to send its data. This increases the life time.

4.2.4. *Adaptive Application-Independent Data Aggregation in WSN (AIDA)*

As the name suggested, the aggregation decision is independent of the application and it also ensures the timely delivery of the packets. It resides between the Routing and MAC layers of the network stack as in fig. 3 and hence it doesn't require any modification in the existing network and medium access protocols. [11]

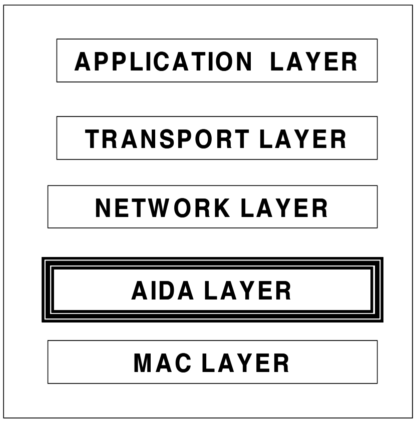

Figure 3: Architecture of AIDA

It adaptively adjusts its aggregation strategies according to the traffic conditions and the sensor network requirements. There are four aggregation strategies supported in this framework. It buffers the packets from the network layer, and the network units are aggregated by using any of the following methods and schedules this aggregated packets to MAC layer for transmission. No Aggregation, where packets are not aggregated, Fixed Aggregated in which fixed numbers of packets are aggregated, On-demand scheme, in which the aggregation takes place until the channel is available for transmission and *Dynamic Feed back loop* combines the fixed scheme and the on-demand scheme.

Tian He *et al.* simulated this frame work using GloMoSim simulator and verified the results with the testbed with Berekely MICA motes. The end-to-end delay, energy consumption, MAC control packet overhead, Degree of Aggregation and AIDA overhead per packet delivered are considered as the performance metrics. This shows that the end-to-end delay of dynamic aggregation scheme is 80% less than that of non aggregation scheme under heavy traffic load condition. Also, the energy consumption is reduced by 50% with reduced overheads

4.2.5. Load Balanced Tree Protocol (LBTP)

The main purpose of LBTP is to gather periodical data from all sensors to a sink. In LBTP, the non leaf nodes have similar amount of children and the tree structure changes when the energy of the non leaf node is lower than the predefined threshold. [12] It first forms the Breath First search (BFS) tree and the sink adjusts this tree to load balancing tree.

Each node broadcasts the HELLO message periodically. The HELLO message contains the packet type, broadcasting node id, remaining energy, parent id, and set of possible parent candidates. The node which receives the HELLO message will update its neighbour table. To balance the tree, after receiving the TREQ packet, each node has to wait for the predefined time to receive any HELLO message. After updating the NT by HELLO message, it will send the Tree Reply (TREP) packet to its parent, so that the tree can be adjusted properly.

The performance of the algorithm in terms of the amount of data received vs. number of sensors and the lifetime of the sensors vs. number of sensors is compared with that of DQEB, BFT and SLBTP. They claim that LBTP can receive more data when the density of the network is very high since the lifetime of the nodes is extended.

4.2.6. Heuristic Algorithms for Real-time Data Aggregation

In real time applications, the packet should be delivered to the sink in prescribed time bound. The packet arriving after the specified time limit is worthless. There are some applications like real time control system applications, which require this time limitation very firmly. This stringent requirement of time is called hard real time requirement. Some applications like live video streaming doesn't require this type rigid time requirements and are called soft real time. In soft real time applications, some packets arrive after the time bound will not collapse the entire system but the quality will be poor. Many researchers are trying to realize the soft real time data aggregation. In order to conserve the energy, the WSN nodes go to sleep state very often. By having such sleeping nodes, it is very difficult to achieve the hard real time bounds. The time constraints of a WSN application can be satisfied by constructing the tree such a way that the number of hops traveled by a packets should be minimum (HOP constraint H) and also each aggregator should have certain number of children (DEGREE constraint D). The end-to-end delay can be decreased by having less HOP count, which reduces the number of hops traveled by a packet and less DEGREE count, which decreases the waiting time of the aggregator to receive packets from its children. To have less number of hops, more children are added to each aggregator, which increases the degree count. Hence there is a tradeoff between H and D.

Hongju Cheng Q. *et al*. suggested the three different tree construction algorithms for real time data gathering in which the packet should be transmitted with in specified time bound. They proposed three heuristics algorithms to build a MST with hop and degree constraints Node-First Heuristic (NFH), Tree-First Heuristic (TFH), and Hop-Bounded Heuristic (HBH) [13]. Simulation results reveal that they are all suitable to solve the real-time data aggregation problem and the performance of these algorithms is tested in terms of total energy cost against the increase in transmission range. It shows that, NFH is the performing better compare to other two algorithms since the energy cost is close to that of non constraint MST.

4.2.7. Correlation Aware Data Aggregation Tree

Due to the resource constraint nature of the sensor nodes, they are deployed densely in the field, wherein the phenomena of interest to be monitored. The readings reported by these sensors are highly correlated and termed as spatial correlation. The data aggregation trees discussed above are not exploiting the correlation between the sensor nodes. In the following paragraphs, the data aggregation tree, Semantic/Spatial correlation-aware tree (SCT) that exploits the correlation strategies can be elaborated

In SCT, the entire network is divided into concentric rings and these rings are further divided into sectors. A node from each sector can have an aggregator depends on the residual energy and its location. The nodes, which are in a sector are closely packed and will report the data which are highly correlated. Hence this structure exploits the spatial correlation. [14]

The given network with n no of nodes which are spread over the radius R is divided into m concentric rings and each ring is divided into sector that contains n_0 nodes. During the initial setup period, the sink broadcasts the packet that contains the information about the sink location, number of nodes in the network, radius of the network, number of concentric rings m, and the number of nodes in each sector n_0. By knowing the sink location, each node in the network will calculate the ring number i in which it resides and also calculates the number of sectors in that ring i as shown in the figure. 4

Once the network is divided into circles and sectors, the node which acts as an aggregator for each sector must be selected. The node which is close to the geometric

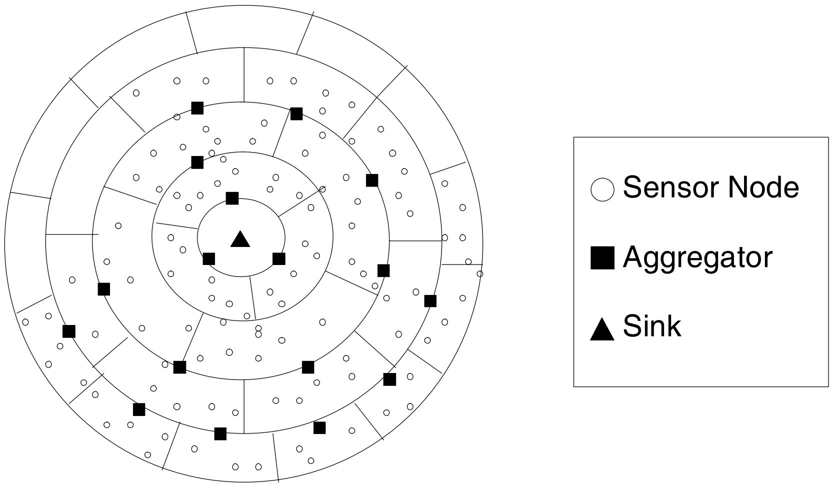

Figure 4: SCT Structure

centre of the lower arc of the sector is selected as aggregator node for that sector. When a node in a sector has a data to send, it calculates the geometric centre and send the data to the node nearest to this centre. If a node receives the first packet, it takes the roll of data aggregator and broadcasts the message to its entire neighbor, declaring that it is the aggregator for that sector. Hearing this message all other nodes will route their packets to this node.

The role of the aggregator is shifted after few rounds of query propagation to achieve load balancing. Also, the location of the ring is shifted slightly and the sector is also shifted by some offset value. SCT ensures the reliable delivery of packets both during query as well as data delivery phase. During the query transmission phase, a node which is closer to the ideal location broadcasts itself as aggregator if the current aggregator fails. If the aggregator fails during the data collection phase, it is notified to the source node and hence the source node will retransmits the packet. This retransmission will initiate the selection of new aggregator. The performance of the tree is compared to that of correlation unaware structures and they show that the SCT perform better in terms of message cost and latency. The advantages of SCT are it does not require any centralized coordination, it requires very low overhead to maintain the tree and the tree can be constructed instantaneously.

5. NETWORKING ISSUES

Due to the adhoc nature of the wireless networks and the dynamic channel conditions, the wireless channels are unreliable and prone to error. While doing aggregation, the packets contain more information and hence the packet loss is not acceptable. Hence the packet delivery should be reliable over unreliable wireless medium. This raises the reliability issues in WSN. Also the packets should be transmitted to the sink in secured manner. The hackers should not modify the content or they should not send the wrong information to the sink which will mislead the sink. Hence the security issues are also very important while aggregating the information. In this section we will discuss the reliability and security issues in details.

5.1. Reliability

The loss of aggregated packets in WSN causes more energy loss since lot of resources is already invested to transmit the sensor readings from various sensors and retransmission of the lost packets requires more energy. This leads to more energy wastage in the sensor nodes which is undesirable. Holger Karl, Marc Lobbers investigated the strategies to adaptively employ different link-level error control mechanisms (FES and ARQ) depending on how precise the information in the message packet. [15] Lost messages in the child – parent link will not create much loss but the lost of aggregated message leads to lot of information loss as well as lot of energy has been invested would have been lost.

5.2. Security

The two fundamental issues to be considered in wireless communication are data confidentiality and data Integrity. Data Confidentiality deals with the protection against the eavesdropping of the transmitted data by the intruders. Sensor nodes are deployed in the unattended area and are easy for the intruders to get or change the information physically and it can be made as malicious node. Also the communication medium is wireless, the information can be taped or modified easily. It necessitates the use of encryption methods to transmit the data. The data integrity ensures the consistency of the received data. It is easy for the intruder to spread the erroneous information or modify the transmitted information in the network. A proper Message Authentication Algorithm (MAC) schemes can verify the integrity of the message.

These security issues are more critical while aggregating the data. The aggregated packets are having more information and hence this should not be hacked or modified by the intruder. The source node or the aggregator node may become malicious node. A compromised node can modify, forge or discard messages. If the source node is compromised, it may send the wrong reading to the aggregator which results in corrupted aggregation at the aggregator. If the aggregator is compromised, it can either send the wrong aggregated result to the sink or it can use the wrong aggregator operator. Both of them made the sink difficult to estimate the original readings from the altered aggregated readings.

In summary an adversary can damage the data confidentiality by the following attacks: 1) eavesdropping the messages in the wireless channel; 2) compromising a node and obtaining all keys stored in it; 3) using the compromised node's keys to deduce the keys employed elsewhere in the network; 4) using the compromised node's keys to inject unauthorized malicious sensor nodes in the network. The adversary can also spoil the data integrity by the following attacks: 1) injecting arbitrary chosen malicious data into the compromised sensing nodes 2) modifying, forging, or discarding messages in the compromised aggregator nodes. [16]

The secure aggregation should be designed to address these issues as well it should take care of the sensor nodes constraints. The size of the encrypted message should be less, the execution time and memory foot print of the security algorithm should be minimal. Also, the secrete key distribution algorithms should be secure and efficient.

5. Conclusion

Normally, the tree based structure will take time and energy for their construction and their maintenance. Also, if a non leaf node breaks, the particular sub section of the tree is disconnected from the network. It necessitates the implementation of reliability methods to ensure the delivery of the packets. Due to the dense deployment nature of the sensor networks, it is necessary to take care of the spatial and temporal correlations into account while designing data aggregation algorithms. Also, appropriate security algorithm should be used to ensure the confidentiality and integrity of the message. In addition to all the above limitations, it uses a simple routing method reduces the complexity of the routing algorithms.

References

[1] I. Akyldiz, W. Su, Y. Sankarasubramanian and E. Cayirci, "A Survey on Sensor Networks", *IEEE Communication Magazine*, **40**, (8), (Aug. 2002), 103–14.

[2] Yang Yu, Viktor K. Prasanna, Bhaskar Krishnamachari "Information Processing and Routing in Wireless Sensor Networks" *World Scientific*, (2006).

[3] V. Raghnathan, C. Schurgers, S. Park and M. B. Srivastava, "Energy Aware Wireless Sensor Networks", *IEEE Signal Processing Magazine*, **19**, (2), (March 2002), 40–50.

[4] I. Solis and K Obraczka "In-Network Aggregation Trade-offs for Data Collection in Wireless Sensor Networks", *International Journal of Sensor Networks*, **1**, (3/4), (2006), 200–212.

[5] R Rajagopalan and P K. Varshney, "Data Aggregation Techniques in Sensor Networks: A Survey", *Journal on IEEE Communications, Surveys and Tutorials*, **8**, (4), (2006).

[6] Fasolo E., Rossi M., Widmer J. and Zorzi M, "In-network Aggregation Techniques for Wireless Sensor Networks: A Survey", *IEEE Wireless Communications*, **14**, (2), (2007), 70–87.

[7] B. Krishnamachari, D. Estrin, and S. Wicker , "Modeling Data-centric Routing in Wireless Sensor Networks", *Computer Engineering Technical Report CENG 02-14* , (2004).

[8] S. Madden, M. Franklin, J. Hellerstein and W. Hong, "TAG: a Tiny AGgregation Service for Adhoc Sensor Networks", OSDI December 2002.

[9] A. Sharaf, J. Beaver, A. Labrinidis, K. Chrysanthis, "Balancing Energy Efficiency and qulity of Aggregate Data in Sensor Networks", *VLDB Journal*, **13**, (4), (Dec. 2004), 384-403.

[10] H. Yang, F. Ye, B. Sikdar, "A Dynamic Query-tree Energy Balancing Protocol for Sensor Networks", *Wireless Communications and Networking Conference*, WCNC, **3**, (2004), 1715– 1720.

[11] Tian He, Brian M. Blum, John A. Stankovic and Tarek Abdelzaher, "AIDA: Adaptive Application-independent Data Aggregation in wireless Sensor Networks", *ACM Transactions on Embedded Computing Systems*, **3**, (2), (May 2004), 426–457.

[12] Tzung-Shi Chen, Hua-Wen Tsai, Chih-Ping Chu, " Gathering-Load-Balanced Tree Protocol for Wireless Sensor Networks", *IEEE Conference on Sensor Networks, Ubiquitous, and Trust-Worthy Computing*, (2006).

[13] Hongju Cheng, Qin Liu, Xiaohua Jia, "Heuristic Algorithms for Real-time Data Aggregation in Wireless Sensor Networks", *International Conference on Wireless and Mobile Computing*, (2006).

[14] Yujie Zhu, Ramanuja Vedantham , Seung-Jong Park, Raghupathy Sivakumar, "A Scalable Correlation Aware Aggregation Strategy for Wireless Sensor Networks" *Science Direct - Information Fusion*, **9**, (3), (2008), 364–369.

[15] Holger Karl, Marc Lobbers and Tim Nieberg , "A Data Aggregation Framework for Wireless Sensor Networks" , TKN Technical Report Series.

[16] Yingpeng Sang, Hong Shen, Yasushi Inoguchi, Yasuo Tan and Naixue Xiong, " Secure Data Aggregation in Wireless Sensor Networks: A Survey", Seventh International Conference on *Parallel and Distributed Computing, Applications and Technologies*, (2006).

A. Sivagami, K. Pavai & D. Sridharan
Department of ECE
CEG Campus
Anna University
Chennai, INDIA
E-mail: *siva_psg69@yahoo.co.in; pavai_me@yahoo.com; sridhar@annauniv.edu*

S. A. V. Satya Murty
Networking Section
Computer Division
IGCAR
Kalpakkam, INDIA
E-mail: *satya@igcar.gov.in*