CHAPTER 6

STEINER TREE BASED ENERGY EFFICIENT DATA AGGREGATION ALGORITHM IN WIRELESS SENSOR NETWORKS

6.1 Introduction

The sensor has randomly deployed in the real world environment to collect the sensed information for further actions. Sometime detected information contains the more redundancy because of the particular event may sense by more near sensors. For this redundant information, sensor spent 50% of their energy for the transmission. Therefore, we avoid such transmission on the network and to prolong the network lifetime by the help of data aggregation (Hu Yanhua and Zhang Xincai 2016). Another cause of data redundancy is to occupy the more bandwidth on the networks and consume more energy for the operation. A remedy for the problem is data aggregation a sensor perform the data aggregation algorithm that, aggregate the data and forward to next level(Xiaolan Tang, Hua Xie, Wenlong Chen, Jianwei Niu and Shuhang Wang 2017). The data aggregation has performed based on the nature of network structure, either flat network or hierarchical network.

In Figure 6.1 shows sensor network, a nodes has randomly deployed and sensed data are forward towards sink node t. The node u and v has aggregate their data of region A and B, which has indicated by arrow lines. Once both region data is aggregate at u node then forwarded fused data to node t. For the assumption, transmission cost Co is identical to all the nodes and fusion cost is qo has calculated based on total no. of incoming data. The total aggregated data at the point v has
calculated by \( (w(u)+w(v))(1-\sigma_{uv}) \) where \( w(u) \) denote data aggregation at \( u \), \( w(v) \) denote data aggregation at \( v \), \( \sigma_{uv} \) represents the data reduction ratio owing to aggregation.

![Figure 6.1. Illustration of aggregation benefit](image)

From the above model, overall energy spent on the route \( v \) to \( t \) has based on the intermediate hops between them. Suppose there should be \( N \) hops then \( Nc_0(w(u)+w(v)) (1-\sigma_{uv})+q_0(w(u)+w(v)) \). The data fusion has not took in the part then total energy is calculated based on \( Nc_0(w(u)+w(v)) \). To reduce the energy consumption of the network, \( v \) should not perform data fusion as long as \( \sigma_{uv} < \frac{q_0}{Nc_0} \) (Hong Luo et al 2005).

### 6.2 Constructing Aggregation Tree

Several application use the tree structure as a routing structure for their networks. For avoid the huge energy spent to maintain route table. The aggregation algorithms used for two general purpose which is one base station on the entire network and another is tree structure minimize the communication overheads on the
networks. That problem has been proven in NP-hard (Yunquan Gao, Xiaoyong Li, Jirui Li and Yali Gao 2018). The aggregation delay and network lifecycle is depends on their depth and its degree of node. Figure 6.2 shows an original aggregation tree.

![Figure 6.2. Original aggregation tree](image)

In sensor networks, energy consumption is high during data transmission operation compare to CPU operation. One of the issue is bottleneck problem will arise due to the more children for that nodes on the networks. So data aggregation tree must be carefully designed which maintained the balanced degree. Overall design of the data aggregation is organize as hierarchical tree structure and root as a base station. Every non-leaf nodes are performed data aggregation and fusing the data for forwarding to the higher level. From the process communication overhead was largely minimized because of data has fused at each level towards the base station (Yi Yang, Xinran Wang, Sencun Zhu and Guohong Cao 2006).
6.3 Proposed Scheme

The algorithm named Minimum Energy Steiner Tree (MEST), which construct the data aggregation tree for minimizing energy consumption during data transmission. The model has connected Graph $G=(V,E)$, $s(v) \in \mathbb{Z}^+$ and zero weights associated with all node $v \in V$, $V$ denote nodes on the networks and $E$ denote edges on the networks and sink node represent by $r \in V$. The routing tree has formed for the network $G$ with sink node. The directed tree $T=(V_T,E_T)$ along root node $r$, let $V_T=V$ and a directed edge $(u,v)\in E_T$ only if an undirected edge $\{u,v\}\in E$. Node $u$ send the data to node $v$ only if $u,v$ belongs to directed edge $E_T$. The data will reach to root node $r$ based on aggregation ratio performed at hop-by-hop node. The model Graph $G=(V,E)$, $s(v)\in \mathbb{Z}^+$ and zero weights associated with all node, aggregation ratio $q \in \mathbb{Z}^+$ and $T_x$, $R_x$ are belongs to $\mathbb{R}^+$ which measure the energy costs of transmit & receive the data. The $C \in \mathbb{R}^+$, the Minimum Energy Steiner Tree (MEST) request for a routing tree $T=(V_T,E_T)$ with root $r$ and $V_T=V$, to ensure that the total energy cost not higher than $C$ for the transmission data by each node. Sometime longer routing will takes higher energy cost for their transmission. The MEST problem has NP-complete, finding the shortest path for routing to opt the algorithm called Fusion Routing Tree (FRT).

The FRT algorithm consists of two stages namely Parent & Sibling Selection and Dynamic Routing Tree Construction.

6.3.1 Parent & Sibling Selection

In this process each node has determine the role on their networks either parent and sibling nodes and its level. The Node identification message (NIM)
broadcast on the networks which contains destination node identifier (dest_id), sender node identifier (src_id) and level.

Algorithm

Step 1: Sink node transmit NIM to root node with level initial value -1.

Step 2: Root node spread the NIM message with src_id=r and level=-1.

Step 3: repeat the step 4 to 6 until each node set the level and identify parent or sibling node on the networks.

Step 4: Recipient node check NIM.level = -1 then

   Step 4.1: node_i.level = NIM.level + 1

   Step 4.2: SPi = NIM.src_id

   Step 4.3: broadcast NIM to the networks

Step 5: Recipient node check node_i.level-1 = NIM.level then SPi = src_id

Step 6: Recipient node check node_i.level = NIM.level then SSi = src_id

Step 7: Stop
The Parent Set \( (SP_i) \) and Sibling Set \( (SS_i) \) for each node \( i \) has defined as follow:

\[
SP_i = \{ v_i \mid v_j \text{ neighbor of } v_i \text{ also } l_j = l_i - 1 \} \quad (6.1)
\]

\[
SS_i = \{ v_i \mid v_j \text{ neighbor of } v_i \text{ also } l_j = l_i \} \quad (6.2)
\]

In Figure 6.3 shows the flow of NIM by arrow indication and corresponding nodes are set the values of the sets \( SP_i \) and \( SS_i \).

### 6.3.2 Dynamic Routing Tree Construction

Dynamic routing tree construction begin from root node to disseminate the query message on the networks. The query msg contain the information are dst_id, src_id, min_dist, query. The following algorithm shows the dynamic routing tree construction process. Figure 4 and 5 shows the example of how dynamic routing tree constructed by user query.
Algorithm:

Step 1: Sink node pass the query Q to root.

Step 2: Root node disseminate src_id and min_dist = 0

Step 3: Node i receives query Q message

Step 4: check src_id of query Q message belongs to SP_i

Step 4.1: assign \( MD_{SP,i,Q} \) with src_id, min_dist

Step 4.2: compare the \( MD_{SP,i,Q} = SP_i \) and node_i is candidate node of Q then set \( MD_{i,Q} = 0 \)

Step 4.3: compare the \( MD_{SP,i,Q} = SP_i \) and node_i is not candidate node of Q then assign \( MD_{i,Q} = \min ( MD_{SP,i,Q} ) +1 \)

step 4.4: update query parent node with MinDistId(\( MD_{SP,i,Q} \)) and src_id with own node address then broadcast the message

Step 5: check src_id of query Q belongs to SS_i and min_dist value = 0 then assign min_dist = 0 for sibling node src_id

Step 6: check src_id of query Q belongs to SS_i and min_dist value != 0 then assign min_dist = Q.min_dist – 1

Step 7: Repeat step 4 to 6 for every node decides their SP node.

Step 8: Finally the node send its data to node which has minimum distance id.
Figure 6.4. Query Dissemination

Data forwarding using the routing tree the leaf node 6 sent the packet to superior node 7. In node 7 prepare one single packet by the data aggregation of node 6 & 7 and forward into superior node 3. Once again the same process happens at node 1 to aggregate the packet from node 3 and node 4 delivered to root node r. Similarly, all the branches of tree data are aggregate and forward into the superior node to reach root tree. In Figure 6.4, the query dissemination help to choose the min_dist value, the values have stated on the lines among sibling. From the example, node 4 has min_dist value 0 with sibling node 3, whereas node 3 has the min_dist value 1 with sibling node 4.

Figure 6.5. Dynamic Routing Tree Construction
6.3.3 Data Gathering in FRT

Based on the query Q has been sent by the sink node, all sensor node sends their data. Data forwarding initiated from the bottom node which satisfies that query message passed by sink node then forward to either parent or sibling node on the constructed tree. Data aggregation process complete before data forwarding to other nodes.

Each sensor nodes which has the data wish to send, and it has two opportunities for transmission to the next level. The node will choose the min_dist value as the next node in the upper level. Another choice of next node forward is by random selection among parent and sibling nodes when both have same min_dist value. The node always prepare the parent node when both have the identical value.

Figure 6.6. Shows data gathering process at the different level on the network graph. In Phase-1, node 6 wishes to send data to the next level then it chooses the min_dist value as sibling node 5. In Phase-2, node 5 want to forward the aggregated data to the next level. There are four neighbor nodes 2,3,4 & 6 has connected with node 5,
and min_dist value of node 2, 4 & 6 has same then node 5 prepare the parent node-2 among them.

6.4 Simulation And Results

6.4.1 Simulation study

This work is simulated over the Network Simulator 2.32. The performance is evaluated by comparing with related schemes QBRT, EEDAT and AggreTree. The results has simulated over different simulation topologies. The Table-6.1 shows the parameter which has used in the simulation.

Table 6.1. Initial Parameter Settings For The Simulation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area of deployment</td>
<td>150m ×150m - 1200m×1200 m</td>
</tr>
<tr>
<td>Number of nodes</td>
<td>100 -500</td>
</tr>
<tr>
<td>Length of the packet</td>
<td>500 bits</td>
</tr>
<tr>
<td>Initial energy of the sensor nodes</td>
<td>2 J</td>
</tr>
<tr>
<td>Transmission range</td>
<td>5m</td>
</tr>
<tr>
<td>Simulation time</td>
<td>500-2500 sec</td>
</tr>
<tr>
<td>Energy consumption on circuit</td>
<td>50 nJ/bit</td>
</tr>
<tr>
<td>Base station location</td>
<td>(0,0)</td>
</tr>
</tbody>
</table>

6.4.2 Algorithms used for Comparison

The protocols QBRT: Query-Based Routing Tree (In Chul Song, Yohan Roh, Dongjoon Hyun and Myoung Ho Kim 2006), EEDAT: Energy-Efficient Data Aggregation Tree (Md. Sajidul Islam, Imtiaz Bin Rahim and Mosarrat Jahan 2014) and aggregate: Aggregation Tree-Based Data Aggregation Algorithm in Wireless
Sensor Networks (Hu Yanhua and Zhang Xincai 2016) has taken to compare the performance with the proposed scheme.

Query-Based Routing Tree (QBRT) is constructed for each continuous query with the goal of increasing the amount of in-network processing. First, the MD-tree is defined for a given continuous query, using a measure called the minimum distance. The minimum distance of a node indicates how far the node is from some other node that generates a message. Next, the construction of MD-trees is described for sensor networks. MD-trees has constructed by the messages generated from sensor nodes.

Energy-Efficient Data Aggregation Tree (EEDAT) (Dhananjayan Gayathri and Subbiah Janakiraman 2016) reduces the cost of data transmission by data aggregation with intermediate nodes. The intermediate nodes have merged the data packet and reduce the hop distance of traveling data packet on the network which helps to reduce the transmission cost. For data forwarding, each node has selected the node in the upper-level tree by the aggregation ratio. The intermediate nodes have selected based on its remaining energy level.

AggreTree minimizes the energy consumption by maximal weighted independent set. The data aggregation tree has formed by link collision matrix, and that communication has been establishing through the link set. From the collision matrix, an approximate maximal weighted independent set has constructed through each interval of the communication link. The communication links are carefully selected for a low weighted delay and increase the lifetime of the network.
6.4.3 Results and Discussion

The performance of MEST has studied with respect to throughput, aggregation latency, total energy consumption, communication overhead, aggregation accuracy and number of alive nodes (Dhananjayan Gayathri and Subbiah Janakiraman 2016).

Throughput: from the given period the total number of bits successfully delivered to the destination — the excellent performance of the protocol by providing higher throughput.

Aggregation Latency: Latency due to the performing the data aggregation operation where data has come from nearest and farthest sources.

Total Energy Consumption: It measures the node spend their energy for their operation and transmit/receive a packet on the networks.

Communication Overhead: Sensor nodes have exchanged packet for establishing the communication with other nodes on the network for sending data packets to the destination. The length of traveling the data packet to increase communication overhead on the network.

Aggregation Accuracy: It is the ratio between the total number of packet received at the base station to the total number of packets generated by each node. An ideal state, all the scheme perform full accuracy on the data aggregation. Sometimes data could be lost due to collision and noisy communication.
Figure 6.7 Throughput efficiency

Figure 6.7 shows the throughput efficiency of the proposed mechanism MEST* proposed with the other protocols. The throughput of MEST is 1.4kbps for 400 rounds, but other protocols have the value from 0.7 to 1.1kbps. Because of MEST routing structure use the FRT method for a fusion decision. The throughput of MEST is 3kbps for a maximum of 1000 rounds, but the other protocols QBRT, EEDAT and AggreTree have throughput 1.5kbps, 1.95kbps and 2.15kbps respectively.

Figure 6.8 shows the effect of latency caused during data aggregation in MEST. The longer latency occurred in the schemes QBRT and EEDAT is due to the control and communication overhead which occurs while establishing data aggregation. In contrast the proposed scheme MEST ensures shorter latency as the time consumed for exchanging control overhead is very much reduced. From the node identification message, each node in the network to know their level, parent and sibling. It has been observed through simulation the aggregation latency caused by
proposed MEST when there are 200 nodes in the network is 110s. But the related schemes EEDAT and AggreTree achieve latency of 190s and 150s due to the exchange of control messages in achieving the aggregation process.

![Figure 6.8 Aggregation Latency](image)

Energy consumption on the networks has observed with other schemes. In EEDAT and QBRT schemes are close to each other in total energy consumption. In Figure 6.9, the result shows that the proposed scheme which consumes half of the energy with other schemes. When increasing the network scale, the proposed MEST aggregation mechanism perform the data aggregation and reduce the number of transmissions.
Figure 6.9 Total Energy Consumption

The communication overhead of the proposed MEST has compared with other schemes. In Figure 10. shows that with different network sizes, the node has transfer average number of bytes. The communication overhead may vary from network structure. The nodes have randomly deployed in the area with different network structure for every simulation. To observe the communication overhead when to increase the network size, the MEST has still sustained the cost of communication at some point, but other schemes have fluctuation. It has concluded that the aggregation approach does not add much overhead than the no-aggregation approach.
In Figure 6.11, the measure of aggregation accuracy with different simulation time to other schemes. The MEST has given the excellent result of data aggregation accuracy compare with AggreTree and others with less period of simulation time. In AggreTree scheme which has taken outlier data has to be included for data
aggregation, so it took more time spent on the process. There is no consideration of outlier data for aggregation in the proposed MEST scheme.

In Figure 6.12, shows measure the alive nodes during the various simulation time. From the observation, AggreTree and MEST are the most similar result of network lifetime given in the initial and less than 100s. After the few times, MEST withstands the node alive is higher than others schemes. In every 50s time interval MEST the alive node ratio which is 5-10 % great than other protocol. For the reason that MEST reduce their energy consumption on their query dissemination and dynamic routing tree construction process.

Figure 6.12 Number of Alive Nodes Comparison
Table 6.2 Comparative analysis of MEST with the existing schemes

<table>
<thead>
<tr>
<th>Parameters</th>
<th>QBRT</th>
<th>EEDAT</th>
<th>AggreTree</th>
<th>MEST (Proposed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes alive during the simulation time of 200s</td>
<td>15</td>
<td>20</td>
<td>20</td>
<td>45</td>
</tr>
<tr>
<td>Aggregation accuracy during the simulation time of 80 seconds (%)</td>
<td>40</td>
<td>45</td>
<td>60</td>
<td>75</td>
</tr>
<tr>
<td>Average byte sent per node for a network of size 200 nodes (bytes)</td>
<td>45</td>
<td>50</td>
<td>65</td>
<td>75</td>
</tr>
<tr>
<td>Total energy consumption for a network of size 250 nodes (J)</td>
<td>345</td>
<td>295</td>
<td>240</td>
<td>195</td>
</tr>
<tr>
<td>Aggregation latency for a network of size 300 nodes (s)</td>
<td>350</td>
<td>290</td>
<td>220</td>
<td>180</td>
</tr>
<tr>
<td>Throughput for a maximum of 1000 rounds (kbps)</td>
<td>1.5</td>
<td>1.95</td>
<td>2.15</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 6.2 shows the comparison of MEST with the existing protocols for different parameters. The parameters may be number of alive nodes, aggregation accuracy, average byte sent per node, total network energy consumption, aggregation latency and throughput.

6.5 Summary Of Contributions

This paper presents an algorithm named Minimum Energy Steiner Tree (MEST) that help to minimize the energy utilization and best shortest route for data delivery in a wireless sensor network. The proposed work has divided into two categories one is Fusion routing Tree and dynamic routing. The FRT algorithm consists of two stages namely Parent & Sibling Selection and Dynamic Routing Tree Construction. A Parent and Sibling Set for each sensor node have identified in the Parent & Sibling Selection process. After this process, all the nodes get to know their levels, parent and sibling nodes of the networks. In dynamic routing tree
construction, a query message floods from the root node to entire networks. The data transmission begins from the last node to the root node. During the transmission, partial aggregation completed and forwarded into higher nodes.

The proposed scheme has simulated and their result compared with similar schemes. The following metrics have taken for comparison to other schemes such as throughput, reducing the aggregation latency, increasing data aggregation accuracy, minimizing the total network energy consumption, increasing the average byte sent and maximizing the network lifetime. Each result the proposed scheme has to outperform other schemes.