Chapter 5

Extending Network Lifetime by Time-Constrained Data Aggregation in WSN

5.1 Introduction

Recent advances in technology have made wireless sensors compact and inexpensive. Networks formed from such sensors are known as wireless sensor networks and are used in a wide range of applications such as environmental surveillance, military operation, and other domains. The wireless sensor network consists of groups of nodes, which captures and transmits the data to the base station. The base station has continuous power supply, while the nodes are battery-powered. If a sensor node runs out of power, its coverage is lost. The network lifetime of a wireless sensor network is determined by the time duration before the first node fails in the network [81]. Therefore, it is very important to manage the sensor nodes in an energy-efficient way to extend the lifetime of the sensor network [82].

To increase the network lifetime, the number of packet transmission between the sensor node and the sink must be decreased. Data aggregation is a technique used to combine the information from the sensor nodes surrounding the event and send the information to the end point, which thereby reduces congestion [83]. Wireless sensor networks offer different methods of data gathering in distributed system architectures and dynamic access via wireless connectivity.

5.2 Objectives

- In the tree topology data is aggregated at a required accuracy which is the ratio of actual received number of sensor flows to the expected number of flows transmitted back to the sink.
- The data freshness should also be considered, because reports from sensor nodes that arrive at an aggregating node may have to be held there for some period of time before being reported, so that additional reports have to reach the aggregator from lower nodes.
• To use an intelligent timer and some high-level knowledge of the network to implement an efficient aggregation timing control protocol.

5.3 System Model

Consider a sensor network with \( n \) nodes (1,2,3,…..n) and one base station BS. The nodes monitor the environment and periodically report to the BS. Time is divided into time stamps, and each sensor node generates one B-bit message per time stamp. The messages from all the sensors are collected periodically and aggregated at different levels and sent to the base station for processing. The nodes are powered by batteries and each sensor node has a battery with finite, non-replenishable energy \( E(i) \). The base station is connected to an unlimited power supply \( E(0) = 1 \).

We make the following assumptions about our system: We assume that the sensor nodes and the base station form a connected graph, i.e., there is a path from any node to any other node in the graph. This can be achieved by setting the transmission power level to be above the critical threshold [88][89], which ensures that the network is connected with probability one as the number of nodes in the network goes to infinity. For simplicity, we do not consider dynamic adjustment of the transmission power levels, and assume that all nodes transmit at the same fixed power level. We adopt a simple data aggregation model[117-119]. We assume that an intermediate sensor can aggregate multiple incoming B-bit messages into a single outgoing message of size B bits.

5.4 Problem Definition

Consider a connected graph \( G \) with \( N \) nodes (\( v1, v2...vn \)) powered by batteries with non-replenishable energy \( E(i) \) and a base station \( v0 \) connected to an unlimited power supply with energy \( E(0) \). The nodes monitor the environment and periodically report to the base station. Each sensor node generates one B-bit message per time stamp. The message from all sensors are collected at each time stamp and aggregated at the intermediate sensor into a single outgoing message of size B-bit and send to the base station. The amount of time required to send or receive one bit of data is \( \alpha_s \) and \( \alpha_r \).
Consider Fig 5.1(a) where node 4 has to aggregate the data collected by the children node 1, 2 and 3 and forward that to node 5. In Fig 5.1(b) four columns are displayed where the first three represents the queues which store the packets coming from node 1, 2 and 3. The last column stores the aggregated packets of 1, 2 and 3. The vertical axis denotes the time when the data packet is collected. Data collected at the same time usually contain the information about the same event. In this paper we consider data aggregation is done only on data collected at the same time. Suppose a packet coming from node 1 at time stamp 5 has no packets at that time stamp from node 2 and 3 the aggregation node just forwards packet from node 1.

If the tree T has a lifetime $L(T)$. Data aggregation is required to maximize the network lifetime $(A)$ $\max L(T)$ such that $T \in A(G)$, Where $A(G)$ is the set of data gathering trees in $G$. 
Let $C(T,i)$ be the number of children for node $vi$ in $T$. $B$ is the energy required by $vi$ to aggregate the data received from all children. During each time stamp node $vi$ receives $B$-bit message from each child. The energy consumption of node $vi$ for each time stamp after aggregating the received message and transmitting $B$-bit message to its parent is $\alpha r B C(T,1) + \alpha t B$. Its lifetime is

$$L(T,i) = Ei/\alpha r B C(T,1) + \alpha t B \quad (5.1)$$

In Fig 5.1(c) node 2 postpones all its data collection by one slot, similarly with node 3. Data aggregation cannot be done, since all the packets are collected at different time stamp. As a result, the number of packets in queue $4$ is the summation of queue $1,2$ and $3$’s packets. To keep the network stable and to prevent the queue of node $4$ from overflow, it has to transmit faster than the aggregate rate of node $1,2$ and $3$. This example reveals that as the transmission rate of an aggregation node increases the energy consumption increases and in turn the network lifetime decreases.

Fig 5.2(a), 5.2(b), 5.2(c): An example of data availability constraint
Consider Fig 5.2(a) where node 0, 1 and 2 works as source nodes. Node 1 and 2 are directly connected to the aggregation node 4, whereas node 0 depends on the intermediate node 3 to transfer the data to aggregator 4. Suppose at a particular time \( t_1 \), as shown in Fig 5.2(b), nodes 1 and 2 has delivered some data to 4, whereas there is a delay in the arrival of data from node 0.

1. At that time if node 4 has to wait till \( t_2 \) after receiving the data from node 0 aggregation in done, as shown in Fig 2(c) the delay increases
2. At time \( t_1 \) if node 4 delivers packets, it has to do the same job again after receiving packet from node 0, which results in the increase of network traffic and decrease network lifetime.

### 5.5 Solution and Implementation

Data aggregation aims to combine responses from multiple sensors into a single message. By reducing the number of message transmission in the network, the energy consumption can be reduced and the network lifetime is increased. In practice this is complicated by the fact that not every node has a response ready at exactly the same time as in Fig 5.1.

**A: Time Approximation Algorithm for Data Aggregation at Intermediate Nodes**

Let each aggregation node estimate and report to the base station, the number of children for the aggregation node \( v_n \), the minimum time required to receive B-bit message from one child. The maximum time required to receive B-bit message from at least \( v_n/2 \) nodes. The base station optimizes time allocation for the aggregation node to extend network lifetime.

**Optimal Time Allocation Algorithm**

Input: Aggregation node \( v_a \) with the number of children \( v_1, v_2, v_3 \)

Output: B-bit message from the aggregation node.

1. for time = min to max
2. B-bit message = aggregated result of any two children of node \( v_a \)
3. \( \text{min} = \text{min} + 1 \)
4. end for
5. Send B-bit message generated by the aggregator to its parent.

**B: Time Approximation Algorithm for Data Aggregation Tree**

In an unbalanced network as in Fig 5.2 the response time will vary depending on the difference in tree levels of the responding nodes.

1. The leaf nodes respond as soon as the event occurs and send the sensed data to the aggregator.

2. The aggregator wait for a specific time before aggregating the response received and send the result to the parent.

3. The process continues till the sink receives a single B-bit message.

4. This also allows the sink to recalculate a more appropriate time for the next query if necessary.

**5.6 Performance Evaluation**

![Unbalanced Tree](image)

*Fig 5.3 Unbalanced Tree*
5.6.1 Performance Evaluation – with respect to Time Consumption

In this section we consider an unbalanced aggregation tree Fig 5.3 Let t be a time slot required by the node to send the data to its parent node and t be the time taken by the aggregate node to aggregate the values once it has received the data from all the children node.

Fig 5.4 Data Aggregation Time at Intermediate Nodes

In Fig 5.4 the tree has 9 source nodes sense the data and forward it to the sink node S, through the aggregation nodes. Let t1 be the time at which node 2, 5, 7, 10, 12, 13, 14, 15 and 16 sense data. At time t4 data is aggregated at nodes 4. At t5 data is aggregated at nodes 3. At t6 data is aggregated at node 1. Data is aggregated at node 0 at time t8 and at t9 is forwarded to the sink S. The data generated at time stamp t1 reaches the source only at time stamp t10.

5.6.2 Performance Evaluation – with Respect to Energy Consumption

A sensor node spends maximum energy in data communication both for transmission and reception.

Receiving energy consumption

\[ E_{rx} = (P_{Lo} + P_{Rx}) t_{rx} \]

\[ P_{Lo} \]: power consumption of the circuitry

73
\( P_{RX} \), power consumption of active components, e.g., decoder, \( t_{rx} \), time it takes to receive a packet

Transmitting energy consumption

\[
E_{tx} = (P_{LO} + P_{PA}) t_{tx}
\]  
(5.2)

\( P_{LO} \), power consumption of the circuitry

\( P_{PA} \), power consumption of power amplifier

\[
P_{PA} = \frac{1}{n} P_{out}
\]  
(5.3)

\( n \), power efficiency of power amplifier, \( P_{out} \) desired RF output power level

Assuming a sensor node is only operating in transmit and receive modes with the following assumptions:

Energy to run circuitry:

\[
E_{elec} = 50 \text{ nJ/bit}
\]  
(5.4)

Energy for radio transmission:

\[
e_{amp} = 100 \text{ pJ/bit/m}^2
\]  
(5.5)

Energy for sending \( k \) bits over distance \( D \)

\[
E_{Tx}(k,D) = E_{elec} * k + e_{amp} * k * D^2
\]  
(5.6)

\[
E_{Rx}(k,D) = E_{elec} * k
\]  
(5.7)

If 1 Mbit of information is transferred from the source to the sink and the broadcast radius of each node is 5 meters

The energy consumed is

\[
E_{Tx}(k,D) = E_{elec} * k + e_{amp} * k * D^2
\]  
(5.8)

\[
E_{Tx} = 50 \text{ nJ/bit} * 10^6 + 100 \text{ pJ/bit/m}^2 * 10^6 * 5^2
\]

\[
= 0.05J + 0.0025 J
\]

\[
= 0.0525 J
\]  
(5.9)

\[
E_{Rx}(k,D) = E_{elec} * k
\]

\[
E_{Rx} = 0.05 J
\]  
(5.10)
\[ E_{\text{pair}} = E_{\text{Tx}} + E_{\text{Rx}} = 0.1025 \text{J} \]
\[ E_{T} = 17 \times E_{\text{pair}} = 17 \times 0.1025 \text{J} = 1.7425 \text{ J} \]  

\[ (5.11) \]

### 5.6.3 Performance Evaluation – using Proposed Method for Time Consumption

In Fig 5.5 the proposed method for time consumption is shown, if an aggregation node has \( n \) children it should receive data from at least \( n/2 \) nodes to aggregate the data and send the aggregated data towards the sink. By applying the proposed method for the same tree at time \( t6 \) aggregation is done at node 0, as we have already received the data from 2 children node out of 3, and data is transferred to the sink. By using the proposed method the data generated at time stamp \( t1 \) reaches the source at time stamp \( t7 \). Fig 5.6 gives a clear comparison of time taken for data aggregation with respect to time at intermediate nodes using the simple method and the proposed method.

![Data Aggregation Time at Intermediate Nodes using Proposed Method](image)
5.6.4 Performance Evaluation – using Proposed Method with Respect to Energy Consumption

If 1 Mbit of information is transferred from the source to the sink and the broadcast radius of each node is 5 meters

The energy consumed is

\[ E_{\text{Tx}}(k,D) = E_{\text{elec}} * k + e_{\text{amp}} * k * D^2 \]
\[ E_{\text{Tx}} = 50 \text{nJ/bit} * 10^6 + 100 \text{pJ/bit/m}^2 * 10^6 * 5^2 \]
\[ = 0.05J + 0.0025 J = 0.0525 J \quad (5.12) \]

\[ E_{\text{Rx}}(k,D) = E_{\text{elec}} * k \]
\[ E_{\text{Rx}} = 0.05 J \]
\[ E_{\text{pair}} = E_{\text{Tx}} + E_{\text{Rx}} = 0.1025J \]
\[ E_T \text{ is 15 pairs of transmitting and receiving nodes} \]
\[ E_T = 15 * E_{\text{pair}} = 15 * 0.1025J = 1.5375 J \quad (5.13) \]

5.7 Related Work

Different aggregation algorithms have been implemented to prolong the network life time. A large variety of problems regarding this topic have been extensively studied. Representative problems include the construction of most energy efficient aggregation tree discussed in [90]. In this study a data gathering tree is constructed to maximize the network lifetime, which is defined as the time until the first node
depletes its energy. The problem is shown to be NP-complete. An algorithm is designed which starts from an arbitrary tree and iteratively reduces the load on bottleneck nodes. It is proved that the algorithm terminates in polynomial time and is provably near optimal. In [91] focus is on data aggregation scheduling problem. Based on maximal independent sets, a distributed algorithm to generate a collision-free schedule for data aggregation in wireless sensor networks is proposed. It also proposes an adaptive strategy for updating the schedule when nodes fail or new nodes join in a network. The analysis and simulation results show that the proposed algorithm outperforms other aggregation scheduling algorithms. In [92,93] a lot of work is done on maximizing the lifetime of the aggregation tree. Sensor networks are not just data networks with sensors being the sources of data. Rather, they are often developed and deployed for a specific application. For overall system efficiency, it is necessary for nodes to perform computations on data, rather than simply originating or forwarding data. Thus, in [94] the entire network can be viewed as performing an application-specific distributed computation. It is useful in developing a theory of in-network computation, which aims to elucidate WSN its efficiency to perform under such distributed computation.

The framework of network utility maximization (NUM) was first developed in the context of wire line network in [95,96]. The main idea of the framework is based on the decomposition of a system-wide optimization problem. A distributed congestion control mechanism is designed to drive the solutions of the decomposed problems towards the global optimum. Later on, NUM was studied in the context of wireless networks. In wireless networks, this problem is more difficult because of the interference nature of wireless communication. To address this challenge, a scheduling mechanism is integrated into the optimization framework to stabilize the system [97,98,99,100]. Table 5.1 summarizes the contribution of different data aggregation tree.

5.8 Discussions and Perspectives

The most important challenge in wireless sensor network is to reduce the energy consumption of each node and increase the network lifetime. Many networking schemes are used to minimize the amount of data transmission by data aggregation. Three main factors affecting the lifetime of sensor nodes are as follows: (1) The
consumed energy for sending data from the leaves to the sink. (2) Queuing delay during aggregation. (3) The tree’s delay which is equal to the tree’s depth should be considered. The unique challenges faced during data aggregation in WSN are identified. The importance of time and energy is analyzed while aggregating data in WSN and formulated this problem as network utility maximization problem. An algorithm is also proposed which is implemented to solve the problem. An adaptive scheme then dynamically adjusts the optimal time allotted for intermediate node data aggregation and optimal delay at each higher aggregation node is proposed which is proved to be better than the simple method.

<table>
<thead>
<tr>
<th>Reference</th>
<th>Method Used</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>[90]</td>
<td>Construction of data aggregation tree</td>
<td>It reduces the traffic on nodes near the sink</td>
<td>Delay between the event triggered and data sent to sink</td>
</tr>
<tr>
<td>[91]</td>
<td>Collision-free schedule for data aggregation</td>
<td>An adaptive strategy for updating the schedule when nodes fail or new nodes join the network.</td>
<td>A lot of time and energy is lost in restructuring the network</td>
</tr>
<tr>
<td>[92]</td>
<td>Aggregation node performs computation on data for a specific purpose</td>
<td>Data accuracy is more</td>
<td>Energy consumption is more, queuing time is more</td>
</tr>
<tr>
<td>[98]</td>
<td>The framework for network utility maximization was developed</td>
<td>A distributed congestion control mechanism is designed</td>
<td>Decomposition problem towards the global optimum</td>
</tr>
<tr>
<td>Proposed</td>
<td>Intelligent timer is used to aggregate the data</td>
<td>An efficient aggregation timing control protocol is implemented</td>
<td>High-level knowledge of the network is required</td>
</tr>
</tbody>
</table>