CHAPTER 5

ENHANCED DAO-LEACH VITALITY EVALUATION

5.1 INTRODUCTION

Data aggregation protocols are prescribed in Wireless Sensor Networks (WSNs) to extend the network lifetime by diminishing the energy consumption. The existing Data Aggregation Optimal - Low Energy Adaptive Clustering Hierarchy (DAO-LEACH) protocol for WSN is insecure and prone to false data injection.

This is enhanced in terms of security and fault tolerance based on Gracefully Degraded Data Aggregation (GDDA) to ensure the integrity of the aggregated data. LEACH is the architecture for remote micro sensor networks which combine the concept of energy efficient cluster based routing and media access, together with application specific data aggregation to achieve good performance in terms of system lifetime and latency.

5.2 GRACEFULLY DEGRADED DATA AGGREGATION SCHEME

This approach has been proved to improve system lifetime by an order of magnitude, compared to general purpose approach when the node
energy is limited. This method is able to detect the false data in the sensed data and eliminate them. This ensures the fault-tolerance in the WSN. It is based on Locality Sensitivity Hashing (LSH) technique.

Supported values and support counts are defined for each cluster. Minimum support values for each cluster are estimated in terms of local cluster (\(\text{minSup}_l\)) and neighboring cluster (\(\text{minSup}_n\)). The working of the GDDA scheme in each cluster is given in Figure 5.1

**Figure 5.1 Working of GDDA scheme**
5.3 LOCALITY SENSITIVE HASHING ALGORITHM

Let a $s$-dimensional real space $\mathbb{R}^s$ with two data points $a$ and $b$ be considered. $a$ is known as an $r$-near neighbor of $b$, since distance between $a$ and $b$ is less than $r$. Locality Sensitive Hashing (LSH) algorithm depends on the presence of locality sensitive hash functions. The family of hash functions mapping $\mathbb{R}^s$ to some universal set is denoted as $\mathcal{H}$. A sub-function in $\mathcal{H}$ is denoted as $h$. For data points $a$ and $b$, $h(a) = h(b)$. The family of hash functions $\mathcal{H}$ is said to be *locality sensitive* under the following conditions:

- If $\|a-b\| \leq r$ then $\Pr_{\mathcal{H}}[h(b) = h(a)] \geq c_1$
- If $\|a-b\| \geq kr$ then $\Pr_{\mathcal{H}}[h(b) = h(a)] \leq c_2$

Here, $k$ is a constant, $c_1 = 1 - (r/s)$, and $c_2 = 1 - (c \theta \theta r/s)$. A family of hash functions must satisfy the condition $c_1 > c_2$. $\mathcal{H}$ can determine whether $a$ and $b$ are in the $r$-neighborhood of each other. Random hyper plane-based hash functions $h_{rd}(m)$, $h_{rd}(n)$ are considered with vectors $(m, n) \in \mathbb{R}^s$, $rd$ is the random hyper plane. The vectors $(m, n)$ give the cosine similarity metric of the data points.

\[(m, n) = \arccos \left( \frac{m \cdot n}{\|m\| \|n\|} \right) \quad (3.1)\]

\[h_{rd}(m) = \begin{cases} 1, & r \cdot d(m) \geq 0 \\ 0, & r \cdot d(m) < 0 \end{cases} \quad (3.2)\]

The probability distribution (Pr) for the vectors $(m, n)$ is defined as:

\[\Pr[h_{rd}(m) = h_{rd}(n)] = 1 - \frac{d_H(LSH_m, LSH_n)}{\pi} \quad (3.3)\]

\[d_H(LSH_m, LSH_n) = j \cdot (1 - Pr) = j \cdot \frac{(m, n)}{\pi} \quad (3.4)\]
The hash function determines the similarity between any pair of datasets in terms of angle between two vectors. Equation (3.4) defines the Hamming distance between the LSH codes of vectors $m$ and $n$, i.e., $LSH_m$ and $LSH_n$. The bit length ($j$) of each LSH code is much smaller than the original vectors $m$ and $n$. The similarity threshold ($\Theta$) is expressed in terms of Hamming distance as:

$$\Theta_{dt} = j\Theta/\pi$$ (3.5)

### 5.4 DATA AGGREGATION

Sensor networks are collection of sensor nodes which co-operatively send sensed data to the base station. As sensor nodes are low in energy, an efficient utilization of power is essential in order to use networks for longer duration. Hence it is needed to reduce data traffic inside sensor networks, and then data aggregation algorithms is to gather and aggregate data in an energy efficient manner so that network lifetime is enhanced. Wireless Sensor Network (WSN) offers an increasingly sensor node needs less power for processing compared to transmitting data. It is desirable to do with network processing inside network and reduce packet size. One such way is data aggregation which is an attractive method of data gathering in distributed system architectures and dynamic access via wireless connectivity.

The process of grouping the sensor nodes in a densely deployed large-scale network is known as clustering. The intelligent way to combine and compress the data belonging to a single cluster is known as data aggregation in the cluster based environment. There are some issues involved with the process of clustering in a Wireless Sensor Network (WSN). The first issue is, how many clusters should be formed which could optimize some performance parameter.
The second could be how many nodes should be taken in a single cluster. The third important issue is the selection procedure of CH in a cluster. Another issue is that user can put some more powerful nodes, in terms of energy, in the network which can act as a CH and other simple node work as cluster member only.

The main aim of data aggregation algorithms is to gather and aggregate data in an efficient manner so that the network lifetime is enhanced. Wireless Sensor Networks (WSNs) offer an increasingly attractive method of data gathering in distributed system architectures and dynamic access via wireless connectivity.

Data aggregation protocols aim at eliminating redundant data transmission and thus improve the lifetime of energy constrained Wireless Sensor Network (WSN). In Wireless Sensor Network (WSN), data transmission takes place in multi-hop fashion where each node forwards its data to the neighbor node which is nearer to sink.

One profit of a wireless sensor network is the fine-grain sensing which the large and impenetrable sets of nodes can contribute. The sensed values must be aggregated to avoid devastating amounts of traffic back to the base station. All locations must be secured.
The general data aggregation algorithm works as shown in the above Figure 5.2.

There are many types of aggregation techniques present and some of them are listed below.

- Centralized Approach
- In-Network aggregation
- Tree Based Approach
- Cluster Based Approach

Each SN senses the environment p times and stores the sensed values, each of q bits length. Thus, each SN has a data vector of size \((p \times q)\) - bit. Transferring data vector to CHs results in rapid depletion of the SN’s battery. LSH codes are generated by the SNs to decrease the amount of data transmitted.

LSH codes are used to define the sensor data using less number of bits. LSH algorithm is used to evaluate a \(j\)-bit LSH code where \(j \ll (p \times q)\).
The false data detection accuracy increases when the values of $j$ and $(p \times q)$ are close to each other.

Each CH requests the SNs in its cluster to transmit their LSH codes for data aggregation. The SNs append their unique IDs along with the LSH codes. Using (3.4) and (3.5), the CH compares the LSH codes of any SN pair.

### 5.4.1 Different LSH Codes

The relation between the different pairs of LSH codes is determined by the CH based on their Hamming distance and similarity threshold $\theta_{dlh}$. When a LSH code is similar to the other LSH code, its support count is incremented by 1.

When false data are present in the local cluster, the support count of the LSH codes is less than $\text{minSup}_l$. The CHs of the neighboring clusters exchange their local false data among themselves to determine if these false data would affect their support count. Each CH compares the LSH codes of its neighboring false data with its cluster’s LSH codes. The support counts are updated after each comparison with the neighboring cluster’s false data. The neighboring CHs exchange the updated support count of local false data.

### 5.4.2 Similar LSH Codes

The CHs determine the SNs which transmitted similar LSH codes. This is used to eliminate the recursive data transmission from SNs to CHs. When more than one SN contains similar LSH codes, then the CH selects only one SN among them to transfer its original data.

After the determination of false data and LSH codes, the CHs obtain the list of false data and SNs containing similar LSH codes. CHs
eliminate the false data and request only one SN to send the original data for each similar LSH code.

Only the requested SNs transmit their sensed data to the CH, and the CH does not accept data from any other SNs. This ensures that no false data are included in the aggregated data and there is no repetitive data transmission from SNs to the CH. The CH aggregates the received data and transmits the aggregated data to the base station.

Data aggregation protocols are essential for WSN to lengthen network lifetime by diminishing energy consumption of sensor networks. For critical WSN, however, not only the energy consumption of SNs but also the correctness of the data aggregation results is critical.

As WSNs are mostly deployed in harsh and hostile environments, malfunctioning and compromised SNs negatively affect the correctness of the data aggregation results. Data aggregation pattern eliminates the false data sent by malfunctioning and compromised SNs. To conserve energy while ignoring false data, an in-network outlier detection technique is based on locality sensitive hashing pattern. Figure 5.3 depicts the outlier sources in WSN.

![Figure 5.3 Outlier sources in WSN](image-url)
Gracefully Degraded Data Aggregation (GDDA) pattern uses an in-network outlier detection mechanism based on Locality Sensitivity Hashing (LSH) technique. With the help of LSH technique, GDDA protocol is able to detect outliers in a distributed and energy-efficient manner.

GDDA protocol ignores the redundant data transmission from Sensor Nodes (SNs) to data aggregators thereby incrementing the efficiency of data aggregation process. GDDA protocol increases the accuracy of aggregated data and diminishes the amount of data transmission in the network.

5.5 PROTOCOL GDDA

**Input:** A Wireless Sensor Network with densely deployed Sensor Nodes (SNs), some of which are designated as data aggregators.

**Output:** Even though there are malfunctioning and compromised nodes in the network, false data are not included in aggregated data. Redundant data transmissions from Sensor Nodes (SNs) to data aggregators are prevented.

// **Phase 1:** Data collection and LSH code generation (At Sensor Node (SNs))
for all data aggregation session do
    Collect data set D
    Generate respective LSH code for data set D
end for

// **Phase 2:** Outlier detection and Redundant Data Elimination (At data aggregator)
for all Data aggregation session do
Request LSH codes from Sensor Node (SN)
Measure the similarities among LSH codes
Detect Sensor Nodes (SNs) that sent the same data set
Detect outliers
end for

// Phase 3: Data Aggregation (At data aggregators)
for all Data aggregation session do
    Eliminate outliers
    Determine the Sensor Nodes (SNs) which have distinct LSH codes
    Request only one Sensor Node (SN) to send the actual data for each distinct LSH code
    Aggregate the received data
end for

In distributed and resource-constrained environments, such as WSNs, identifying outliers without increasing the communication overhead is a complex task. Moreover, SNs suffer from the severely limited memory efficiencies.

Therefore, in WSNs, in-network outlier detection approaches which reduce communication and memory consumption of SNs must be employed.

The LSH algorithm used in GDDA allows compact representation of sensor data, which diminishes the communication overhead of outlier detection. GDDA takes advantage of LSH technique by estimating the similarity of sensor data from their LSH codes.

In a clustered WSN, the network is divided into clusters. Each cluster owns an aggregator having a more powerful wireless transceiver that can transmit data directly to the server.
In this framework each sensor transmits data only to the aggregator; hence, each sensor mote can diminish overhead in forwarding data packets. The assumption here is, sensor motes have no mobility, i.e., they are fixed in a position and will not be moved forever. By using a clustered network to reduce power consumption, a data aggregation method is proposed which maintains both secrecy and privacy.

In terms of secrecy, each sensor mote encrypts its reading and transmits the encrypted data to the aggregator. Adversaries will not be able to recognize what reading it is during data transmission.

In terms of privacy, the design aims to diminish redundant reading for data aggregation but this reading remains secret to the aggregator, i.e., the aggregator cannot know anything about these readings.

Besides, the design can also prevent known-plain text attacks, chosen-plain text attacks and cipher text-only attacks.

The data aggregator does not include the faculty data of outliers to data aggregation process and computes the aggregated data.

In addition, while detecting false data, the data aggregator also discovers the SNs which exactly the same LSH codes and prevents redundant data transmission from these sensors. Elimination of redundant data transmission improves the bandwidth and energy efficiency of GDDA.

GDDA protocol is able to detect outliers in a distributed and energy efficient manner. By using LSH codes, GDDA protocol diminishes the redundant data transmission from SNs to data aggregators there by incrementing the efficiency of data aggregation process. GDDA protocol increases the accuracy of aggregated data and diminishes the amount of data transmission in the network.
Energy efficient LSH based outlier detection scheme improves the accuracy and efficiency of the data aggregation process. Data are periodically collected and aggregated in data aggregation sessions. Figure 5.4 shows the Sensor Nodes (SNs) affected by the events.

![Sensor nodes may be affected by the events](image)

**Figure 5.4 Sensor nodes may be affected by the events**

The data aggregator sends aggregated data to the base station over multi hop paths. The assumption is that each data aggregator aggregates its cluster data only and hierarchical data aggregation is not allowed. GDDA protocol can be used for hierarchical data aggregation as well.

GDDA protocol consists of three phases,

- Data collection and LSH code generation
- Outlier detection and redundant data elimination
- Data aggregation

### 5.6 DATA COLLECTION AND LSH CODE GENERATION

In GDDA, data collection and aggregation is performed in sessions. Data aggregation informs their cluster members at the beginning of each data collection phase.
In each data collection session, each sensor node senses the environment $a$ times and stores the sensed values. Assuming that each sensed values is $b$ bits; each sensor node has a data vector of size $(a \times b)$-bit.

In order to diminish the amount of data transmission, sensor nodes generate LSH codes of their data vectors. LSH codes can represent sensor data using less number of bits.

A Sensor Node (SN) applies LSH algorithm to its data and obtain an $n$-bit LSH code where $n \ll (a \times b)$. It is necessary to note that there is a trade-off between the values of $(a \times b)$ and $n$ values are close to each other. The false data detection ability of the protocol increases.

### 5.6.1 Outlier Detection and Redundant Data Elimination

Each data aggregator requests SNs in its cluster to send their LSH codes for the current data aggregation session. The data aggregator looks for the following two cases.

#### 5.6.2 Case I

If there are LSH codes which are significantly different from the rest of the LSH codes, based on the hamming distance between pairs of LSH codes and the similarity threshold, the data aggregator determines that the compared pair of LSH codes is similar.

If an LSH code is found to be similar with another LSH code, then its support count is increased. The LSH codes which have a support count, which is less than predetermined value, are labeled as local outliers.
These local outliers, however, might be affected by the events which occurred in the neighboring clusters.

Therefore, neighboring data aggregators exchange their local false data lists among them to determine these false data who can improve their support count. Each data aggregator compares LSH codes of its neighboring local false data with its cluster’s LSH codes and updates their support counts. Neighboring data aggregators exchange the updated support counts of local outliers.

5.6.3 Case II

If there are LSH codes which are exactly the same: During the comparison LSH code pairs, data aggregators also found the SNs which sent exactly the same LSH codes

In other words, data aggregators discover the SNs which have the same data. This information is particularly useful to eliminate redundant data transmission from SNs to the data aggregator.

If there are more than one SNs which has the same LSH code, then the data aggregator selects only one SN among them to send its actual data, thereby diminishing transmission amount.

5.7 DATA AGGREGATOR

The data aggregator eliminates the false data, and then it determines the sensor nodes which have distinct LSH codes and request only one sensor node to send the actual data for each distinct LSH code.
Only requested sensor nodes send their data to the data aggregator, and the data aggregator does not accept data from any other sensor nodes. This process ensures that,

- No outlier data is included in the aggregated data
- There is no redundant data transmission from sensor nodes to the data aggregator. The data aggregator aggregates the received data and sends aggregated data to the base station.

Energy efficient LSH based false data detection pattern to improve the accuracy and efficiency of the aggregation process.

In large sensor networks, in-network data aggregation significantly diminishes the amount of communication and energy consumption. The aggregation framework does not address the problem of false sub aggregate values contributed by compromised nodes resulting in large errors in the aggregate computed at the BS, which is the root node in the aggregation hierarchy.

This is an important problem since sensor networks are highly vulnerable to node compromises due to the unattended nature of SNs and the lack of tamper-resistant hardware.

The main aim of data aggregation algorithms is to gather and aggregate data in an energy efficient manner so that network lifetime is enhanced.

WSN offers an increasing number of SNs needing less power for processing compared to transmitting data. It is preferable to do in-network processing inside the network and diminish packet size.
One such approach is data aggregation which is an attractive method of data gathering in distributed system architectures and dynamic access via wireless connectivity.

WSNs have limited computational power and limited memory and battery power. This leads to increased complexity for application developers and often results in applications which are closely coupled with network protocols.

In network, aggregation is the global process of gathering and routing information through a multi-hop network, processing data at intermediate nodes with the objective of reducing resource consumption.

Data aggregation protocols aim at diminishing redundant data transmission and thus improving the lifetime of energy constrained WSN. In WSN, data transmission took place in multi-hop fashion where each node forwards its data to the neighbor node which is nearer to sink.

Data aggregation is a process of aggregating the sensor data using aggregation approaches. The general data aggregation algorithm uses the sensor data from the Sensor Node and then aggregates the data by using some aggregation algorithm such as LEACH.

5.8 RESULTS AND DISCUSSIONS

The proposed work Enhanced DAO-LEACH is compared with the former process SCAR. The proposed approach is an energy efficient protocol as it communicates with the help of cluster heads. The Figure 5.5 proves that the proposed approach is energy efficient by plotting the time in X axis which is having unit in Seconds (S) and in the Y axis Average remaining energy is taken which is having the unit in Joules (J).
Figure 5.5 Average remaining energy (J) vs. time (s)

The proposed approach shows energy efficiency significantly. Table 5.1 represents the percentage of difference in average remaining energy between SCAR and enhanced DAO-LEACH corresponding to time.

Table 5.1 Performance analysis-average remaining energy relative to time

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Average Remaining Energy</th>
<th>% of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SCAR</td>
<td>Enhanced DAO-LEACH</td>
</tr>
<tr>
<td>1</td>
<td>0.999</td>
<td>1.2</td>
</tr>
<tr>
<td>2</td>
<td>0.98</td>
<td>1.112</td>
</tr>
<tr>
<td>3</td>
<td>0.936</td>
<td>1.002</td>
</tr>
<tr>
<td>4</td>
<td>0.912</td>
<td>0.982</td>
</tr>
<tr>
<td>5</td>
<td>0.899</td>
<td>0.954</td>
</tr>
<tr>
<td>6</td>
<td>0.885</td>
<td>0.932</td>
</tr>
<tr>
<td>7</td>
<td>0.856</td>
<td>0.925</td>
</tr>
</tbody>
</table>
The Figure 5.6 is also a proof for the proposed system for energy efficiency. The axis has been taken as Number of Received Packets with the Average Remaining Energy (J). The average remaining time for 8 packets has been calculated and by increasing double time it has been extended up to 576 packets. The experimental results show that the proposed Enhanced DAO-LEACH consumes less energy when compared with the former approach ESPA.

![Figure 5.6 Numbers of received packets vs. average remaining energy (J)](image)

**Table 5.2** Performance analysis – average remaining energy relative to number of received packets

<table>
<thead>
<tr>
<th>Number of received packets</th>
<th>Average Remaining Energy</th>
<th>% of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ESPA</td>
<td>Enhanced DAO-LEACH</td>
</tr>
<tr>
<td>8</td>
<td>0.98</td>
<td>1.12</td>
</tr>
<tr>
<td>64</td>
<td>0.95</td>
<td>0.992</td>
</tr>
<tr>
<td>128</td>
<td>0.94</td>
<td>0.982</td>
</tr>
<tr>
<td></td>
<td>192</td>
<td>256</td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Energy remained ESPA</td>
<td>0.915</td>
<td>0.885</td>
</tr>
<tr>
<td>Energy remained DAO-LEACH</td>
<td>0.932</td>
<td>0.905</td>
</tr>
<tr>
<td>% Difference</td>
<td>1.285%</td>
<td>2.21%</td>
</tr>
</tbody>
</table>

Table 5.2 represents the percentage of difference in average remaining energy between ESPA and enhanced DAO-LEACH corresponding to number of received packets.

Figure 5.7: Number of nodes vs. the system energy consumption (mJ)

Another proof for the proposed approach is energy efficiency. From the above graph it is known that the proposed approach Enhanced DAO-LEACH. In the above Figure 5.7 the number of nodes in X axis is taken and 450 nodes have been taken for the experiment.
### Table 5.3 Performance analysis – system energy consumption

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>System energy consumption</th>
<th>% of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SEDAN</td>
<td>Enhanced DAO-LEACH</td>
</tr>
<tr>
<td>50</td>
<td>3.96</td>
<td>3.9512</td>
</tr>
<tr>
<td>100</td>
<td>3.963</td>
<td>3.95234</td>
</tr>
<tr>
<td>150</td>
<td>3.96356</td>
<td>3.95246</td>
</tr>
<tr>
<td>200</td>
<td>3.96345</td>
<td>3.95564</td>
</tr>
<tr>
<td>250</td>
<td>3.96368</td>
<td>3.95642</td>
</tr>
<tr>
<td>300</td>
<td>3.96389</td>
<td>3.95401</td>
</tr>
<tr>
<td>350</td>
<td>3.96375</td>
<td>3.95468</td>
</tr>
<tr>
<td>400</td>
<td>3.96372</td>
<td>3.95567</td>
</tr>
<tr>
<td>450</td>
<td>3.96378</td>
<td>3.95584</td>
</tr>
</tbody>
</table>

In the Y axis System Energy Consumption (mJ) has been taken. The result is showing that the proposed Enhanced DAO-LEACH is showing significant reduction in System Energy Consumption. Table 5.3 represents the percentage of difference in system energy consumption between SEDAN and enhanced DAO-LEACH corresponding to the number of nodes.
In the above Figure 5.8 we have taken the Time (S) is taken in the X axis and Number of alive nodes in the Y axis. The existing protocol ESPA is having lesser number of live nodes but in the proposed Enhanced DAO-LEACH, the number of live nodes is nearly 60 nodes and in the existing protocol it is 20 nodes. This proves that the proposed protocol is having more number of live nodes.

Table 5.4 Performance analysis – number of live nodes

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Number of alive nodes</th>
<th>% of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ESPA</td>
<td>Enhanced DAO-LEACH</td>
</tr>
<tr>
<td>----------</td>
<td>------</td>
<td>---------------------</td>
</tr>
<tr>
<td>1100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>1150</td>
<td>84</td>
<td>96</td>
</tr>
<tr>
<td>1200</td>
<td>76</td>
<td>85</td>
</tr>
<tr>
<td>1250</td>
<td>44</td>
<td>73</td>
</tr>
<tr>
<td>1300</td>
<td>30</td>
<td>58</td>
</tr>
</tbody>
</table>
Table 5.4 represents the percentage of difference in number of live
nodes between ESPA and enhanced DAO-LEACH corresponding to time.

![End-to-end delay graph](image)

**Figure 5.9 Number of nodes vs. end-to-end delay (ms)**

In the above Figure 5.9 the number of nodes has been compared
with the End-to-End delay. Number of nodes have been taken in the X axis
and End-to-End delay (ms) in the Y axis. From the above graph the end-to-
end delay of the proposed DAO-LEACH is lesser than the existing protocol
DKS-LEACH. For this experiment the nodes ranging from 50 to 200 have
been taken. The result showed that the proposed protocol reduces the end to
end delay significantly.

**Table 5.5    Performance analysis – end-to-end delay relative to number
of nodes**

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>End-to-End Delay (ms)</th>
<th>% of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DKS-LEACH</td>
<td>Enhanced DAO-LEACH</td>
</tr>
<tr>
<td>50</td>
<td>99.99</td>
<td>96.26</td>
</tr>
<tr>
<td>100</td>
<td>100.15</td>
<td>96.59</td>
</tr>
</tbody>
</table>
Table 5.5 represents the percentage of difference in end-to-end delay between DKS-LEACH and enhanced DAO-LEACH corresponding to number of nodes.

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>150</td>
<td>100</td>
<td>96.89</td>
<td>3.110%</td>
</tr>
<tr>
<td>200</td>
<td>100.25</td>
<td>97.26</td>
<td>2.983%</td>
</tr>
</tbody>
</table>

**Figure 5.10 Probability of reporting data vs. bandwidth consumption**

The above Figure 5.10 represents the comparison between the former approaches with the proposed approach. In this graph reporting data has been taken in the X axis and Bandwidth consumption has been taken in the Y axis. From the experimental result obtained the proposed approach Enhanced DAO-LEACH’s bandwidth consumption is lesser when compared with the former approach.
Table 5.6 Performance analysis – bandwidth consumption

<table>
<thead>
<tr>
<th>Probability of reporting data</th>
<th>Bandwidth consumption (KB)</th>
<th>% of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PEC2P</td>
<td>Enhanced DAO-LEACH</td>
</tr>
<tr>
<td>0.1</td>
<td>5</td>
<td>2.8</td>
</tr>
<tr>
<td>0.2</td>
<td>11.5</td>
<td>10</td>
</tr>
<tr>
<td>0.3</td>
<td>15</td>
<td>12.5</td>
</tr>
<tr>
<td>0.4</td>
<td>17.5</td>
<td>15.2</td>
</tr>
<tr>
<td>0.5</td>
<td>19.5</td>
<td>16</td>
</tr>
<tr>
<td>0.6</td>
<td>23</td>
<td>20</td>
</tr>
<tr>
<td>0.7</td>
<td>24.5</td>
<td>22</td>
</tr>
<tr>
<td>0.8</td>
<td>27</td>
<td>24.2</td>
</tr>
<tr>
<td>0.9</td>
<td>30</td>
<td>26.1</td>
</tr>
<tr>
<td>1</td>
<td>31.5</td>
<td>27.3</td>
</tr>
</tbody>
</table>

Table 5.6 represents the percentage of difference in bandwidth consumption in Perturbation-based Efficient Confidentiality Preserving Protocol (PEC2P) and enhanced DAO-LEACH corresponding to the probability of reporting data.

5.9 CONCLUSION

Data aggregation and optimal clustering are incorporated in WSN to increase the energy efficiency of LEACH protocol, known as Data Aggregation Optimal LEACH (DAO-LEACH). The data aggregation process of DAO-LEACH is unreliable, insecure, and prone to false data injection.
This is enhanced in terms of reliability, security, and fault-tolerance based on Gracefully Degraded Data Aggregation (GDDA) to ensure the integrity of the aggregated data. It performs better in terms of aggregation accuracy, energy consumption, number of alive nodes, End-to-End Delay (EED), Simple Cluster-based data Aggregation and Routing (SCAR), ESPA (Energy-efficient Secure Path Algorithm), DKS-LEACH (deterministic key management based LEACH), Perturbation-based Efficient Confidentiality Preserving Protocol (PEC2P) and Secure and Efficient Data Aggregation protocol for WSNs (SEDAN). The enhanced DAO-LEACH protocol is energy-efficient over SCAR by an average of 12.105%, over ESPA by an average of 8.34%, over DKS-LEACH by 3.605%, and over SEDAN by 0.45% respectively.