4.1 Introduction

4.2 Proposed iXED method

4.3 Enhancement of XED method
   4.3.1 Offline Step
   4.3.2 Online Step
   4.3.3 Calculation of Packet Loss
   4.3.4 Calculation of Average Efficiency

4.4 Integration of integrated AIS algorithm

4.5 Detection of replicas

4.6 Steps involved in the iXED method

4.7 Pseudo-code of iXED method

4.8 Experimental Setup and Results

4.9 Computational Complexity of the Proposed iXED method
   4.9.1 Storage Overhead
   4.9.2 Communication Overhead
   4.9.3 Computation Overhead

4.10 Achievements of Proposed iXED method

4.11 Limitations of iXED method

4.12 Chapter Summary
4. INFORMATION EXCHANGE BASED REPLICA DETECTION USING PROPOSED iXED METHOD

4.1 Introduction

The distributed replica node detection methods are of different types namely, information exchanged based method, node meeting based method and the mobility assisted based detection method. Of these methods, the proposed method discussed in this chapter is based on the information exchange based detection method. In this method, replicas that exchange the shared information in the mobile Wireless Sensor Networks are detected. In the proposed method, the replica detection is done by enhancing XED with integrated Artificial Immune System (iAIS) algorithm. This method is named as iXED. The main objective of the proposed method is to improve the detection accuracy with improved Packet Delivery Ratio.

4.2 Proposed iXED Method

The proposed iXED method provides a solution to the problem related to information exchanged based node replication attack. The flow diagram of the proposed methodology is given in Figure 4.1.

![Figure 4.1: Proposed iXED Method](image-url)

Enhanced Distributed Node Replication attack detection methods in Mobile Wireless Sensor Networks using Artificial Immune System algorithms
The proposed iXED method is explained in terms of two processes namely

- **Enhancement of XED and**
- **Integration of iAIS to the enhanced XED method**

The proposed iXED method is explained as follows.

### 4.3 Enhancement of eXtremely Efficient Detection (XED) Method

The first step in the proposed iXED method is to enhance the XED method. The eXtremely Efficient Distribution (XED) [68] is one of the information exchanged based detection method. The XED method was proposed to detect the colluding replicas that exchange the shared information between them in the mobile wireless sensor networks. The XED method [65] is based on the challenge and response strategy. The XED method detects the replica nodes locally. The XED method is composed of two steps namely, the Offline and the Online step. They are explained below.

#### 4.3.1 Offline Step

The offline step is performed before the deployment of the sensor nodes and is explained as follows.

Every node stores cryptographic hash function $h(.)$ and security parameter $x$. When a node ‘$p$’ communicates in the network, it stores the random number received from the other nodes. Moreover the node ‘$q$’ also stores the required authentication details to check the received random numbers. Hence, the two arrays of length ‘$l$’ namely $A_p^r$ and $A_s^p$ are used to store the random number of the received node and its authentication details for the node ‘$p$’. Each node has the ability to detect the replica nodes in the network. So, additionally, a node ‘$p$’ stores a set $B(p)$ to store the blacklist nodes identified by the node ‘$p$’. Initially all the arrays are initialized to zero. The pseudo-code for the offline step of iXED method is illustrated in Table 4.1.
Table 4.1: Pseudo-code of the Offline Step of iXED Method

```plaintext
// XED-offline method
// the algorithm is performed by the node p
Store h(.) in p
Store x in p
Set \( A_r^p = \emptyset \)
Set \( A_s^p = \emptyset \)
Set \( B(p) = \emptyset \)
```

The offline method is followed by the online method which is explained in the next section.

### 4.3.2 Online Step

The online step is performed by each node after the deployment of the sensor nodes. The threat model is explained in Section 3.2.2. Based on the threat model described in Chapter 3.2.1, the online step has been explained.

When a node \( p \) meets another node \( q \) for the first time, the two nodes exchange a random number \( r_{p-q} \) and \( r_{q-p} \). The random number exchanged between the nodes for the first time is defined as,

\[
p = r_{p-q} \\
q = r_{q-p}
\] (4.1)

The node \( p \) generates a random integer value \( \delta \in [1, 2^x - 1] \). The generated random value is used as the string value to calculate \( h(\delta) \). The calculated value is stored in an array \( A_s(p) \) which is known as the random number set and sent to \( q \). \( A_s(p) = 0 \), when the nodes \( p \) and \( q \) meet each other for the first time. When the node \( p \) encounters the node \( q \) again, it requests for the previously exchanged number \( r_{p-q} \).

If node \( q \) replies with the correct random number, nodes exchange and store new random numbers \( nr_{p-q} \) and \( nr_{q-p} \) as follows.

\[
A_r^{(p)}[q] = nr_{p-q} \tag{4.3}
\]
\[
A_r^{(q)}[p] = nr_{q-p} \tag{4.4}
\]
The new random number $nr_{p \rightarrow q}$ and $nr_{q \rightarrow p}$ are calculated using the hash function $h(\delta)$ as explained earlier. Otherwise, if node $q$ cannot provide a correct number, node $p$ assumes $q$ as the replica node and stores the array $q$ in the blacklist $B(p)$.

4.3.3 Calculation of Packet Loss

Usually the adversary compromise and replicates the node to perform several malfunctions like injecting the false messages, deleting, dropping and delaying the messages that travel through the replica nodes. By calculating the packet loss, the replica nodes that drop and deletes the messages that travel them can be detected. Hence, for each and every node in the network, the iXED method additionally calculates the packet loss (PL). The PL for each node in the network is calculated as

$$Packet \ Loss = \sum_{i=1}^{n}(S_i - R_i)$$  \hspace{1cm} (4. 5)

where $n$ is the number of nodes, $R_i$ is the number of packets received, $S_i$ is the number of packets sent

When the $PL_i$ has not occurred, the node is considered as the genuine node or self node.

4.3.4 Calculation of Average Efficiency

When the PL has occurred, the average efficiency of each node is calculated because the packet loss might have occurred due to some physical problem with the sensor nodes. Moreover, the replica node will not have maximum data forwarding capability during the communication in the network. The Average Efficiency of each node is calculated as,

$$Average \ Efficiency = \sum_{i=1}^{n} \frac{(R_i - D_i)}{N_i}$$  \hspace{1cm} (4. 6)

where $n$ is the number of nodes, $R_i$ is the number of packets received, $D_i$ is the number of packets delivered and $N_i$ is the total number of packets

The data forwarding capability is denoted by the threshold value that is assigned dynamically. When the node has a greater threshold value, the average efficiency of each node is compared with the existing random number set. When the matching between them
occurs, immediately they are stored in the self node set. Otherwise, the node is compared with the replica node set. If it matches, they are stored in the replica node set. Figure 4.2 explains about the steps 1, 2 and 3.

Figure 4.2: Flow Diagram of Enhanced XED Method
The suspected replica nodes are detected as replica nodes by combining iAIS to the enhanced XED method in order to reduce the false detection rate. Since, iAIS algorithm reduces false positive rate it has been selected for further enhancement. The obtained self node set and the replica node set are used for detecting the replicas that are resilient against collusive replicas with minimum control overheads.

4.4 Integration of iAIS algorithm to the enhanced XED Method

In iAIS algorithm, the concepts of B-cells, T-cells and Dendritic Cells (DCs) are combined and used with enhanced XED method for clone detection.

The B-cell model does the activity of adaptive immunity, which removes the antigens by launching an attack. It is presented by using the classical context of the self non-self discrimination paradigm. The two phases of the B-cell model are the learning phase and the operational phase. The benign behaviour of the system is done in the learning phase. Whereas, in the operational phase, the received antigen is classified as self or nonself.

The basic model of the DCs is inspired from the innate immune system. The innate immune system is an in built immune system that defends against the antigens (Ag). The antigens are used to identify the pathogens. The DCs act as a first line of defense. It represents the functional behaviour starting from sampling Ag in the tissue till determining the context of the tissue as safe or dangerous. The DCs determine the co-stimulation level by processing the signals which are present in the tissue at the time of sampling. If the co-stimulation threshold is exceeded, then the dangerous context is transformed into the mature (non-self) state and the safe context is transmitted into the semi-mature (self) state as shown in Figure 4.3. To present the sampled Ags by DCs in thymus (place where the unsafe context is removed) and maturation/activation of T-cells, the basic Dendritic cell model is extended.
In the enhanced XED method, the co-stimulation threshold is determined by the average efficiency and packet loss. When the packet loss increases, and when the average efficiency decreases, the co-stimulation threshold increases and the nodes are classified as self node and non-self node. As a result of the enhanced XED method, the classified self node is marked as semi-mature state and non-self node is marked as mature state.

4.5 Detection of Replicas

The two states namely, mature and semi-mature states are migrated to the thymus. The thymus is a place where the B-cells produce the antibodies to remove the antigens that cause harm to the self nodes. It then checks the sampled Ags from the replica node set with the T-cells.

**Figure 4.3: State Transition of Self Node and Non-Self Node**
Mature DC

Migrate to thymus and check Ag with T-Cells

Yes

If matches

No

Leave T-Cell Detector Set unchanged

Check whether it is a mature DC

Yes

If mature DC

No

Delete the matching T-cell and generate non-negative T-Cell

Mark the matching T-cells as T-helper cell and match Ag to T-helper Cell Detector Set

No

No

Yes

Mark Ag as replica and mutate B-Cell Detector for affinity maturation with Ag

Mark Ag as self and leave B-Cell Detector Set Unchanged

Mark B-Cell detector as memory detector

End

Figure 4.4: Flow Diagram of Enhanced XED using iAIS algorithm
When the matching occurs between the sampled Ags and the T-cells, it is checked once again whether it is a mature DC. If it is a mature DC, then the appropriate T-cells are marked as T-helper cell and stored it in the T-helper cell detector set. The incoming Ags are matched with the T-helper cell detector set. If it matches, then the Ag is declared as non-self node i.e., the replica nodes. The B-cells are mutated for affinity maturation. The detected B-cells are marked as memory detector. Otherwise the B-cell detector set is marked as unchanged and they are declared as self node. The T-cell detector set co-stimulates the B-cell to finally decide upon the classification of self node and the replica node. Such a co-stimulation process helps to achieve higher detection accuracy.

4.6 Steps involved in the proposed iXED method

The steps involved in the detection of replica nodes based on the information exchanged between the replica nodes are discussed below:

**Step 1:** Initialize a node as ‘q’ and the neighbours of the node as $P_1, \ldots, P_n$.

**Step 2:** During each packet transfer from source node, when the node communicates for the first time, the node ‘q’ issues the random number and maintain the same in the random number set.

**Step 3:** If the nodes happen to meet again, the random numbers issued are checked and stored in appropriate set.

**Step 4:** Calculate the Packet Loss ($PL_i$) of each node using the equation (4.5).

**Step 5:** Calculate the Average Efficiency of each node using the equation (4.6), when the $PL_i$ is high.

**Step 6:** Maintain the node in self node set when the node present in random number set has greater Average Efficiency. Otherwise, maintain the node in replica node set.

**Step 7:** Consider the replica node set for verification. Verify the node’s ID with the IDs present in the network. Maintain them in a separate T-Helper Cell detector set.

**Step 8:** Declare as non-self nodes when the node’s ID is present in the T-helper cell detector set. Remove that particular node from the network.

**Step 9:** Declare the node as self node when the node’s ID is unavailable in the T-helper cell detector set.
4.7 Pseudo-code of the proposed iXED Method

The pseudo-code applied to detect the collusive replicas that share the information is illustrated in Table 4.1.

Table 4.2: Pseudo-code of Proposed iXED Method

| P₁, ..., Pₙ are neighbors of q |
| R(q) |
| For each packet p transfer from source node s |
| While (node ≠ destination node D) do |
| If (node communicates first time) |
| Node q send random number Xᵣ(q) to P₁, ..., Pₙ respectively |
| P₁, ..., Pₙ receive random number Xᵣ(q) and store it to the random number set |
| End if |
| If (node meets second time) |
| Check the random number issued |
| for m=1 to n |
| if (h(Xᵣ(q))(Pₘ)) = Xᵣ(q) |
| add node to X(q) |
| else |
| add node to R(q) |
| Next |
| end if |
| End if |
| Calculate Packet Loss by Packet Loss = ∑ᵢ₌₁ⁿ(Sᵢ - Rᵢ) |
| If (PLᵢ is high) then |
| Calculate AE of each node as Average Efficiency = ∑ᵢ₌₁ⁿ(Rᵢ - Dᵢ) |
| If (AE > threshold) then |
| If (AE (node) is in random number set) then |
| Declare as self node |
| Else |
| Declare as replica node |
| End if |
| End if |
| Transfer replica node set to the thymus (a separate place to detect replica) |
| If (Ag = T cells) then //Ag is the ID of the suspected replica node, T-cells ID of the previously detected replica node |
| If (mature DC) then |
| T-Helper Cell TH = T Cell // T-Helper cell detector is the replica node set |
| Add TH to {TH₁, ..., THₙ} in T Helper cell detector set |
| If (Ag = {TH₁, ..., THₙ}) |
| Non self Ag = Ag |
| Mutate B-cell detector for affinity maturation with Ag |
| Else |
| Self Ag = Ag |
Chapter 4

The proposed iXED method helps in detecting the clone nodes that exchange the shared information between them. Hence, the detection accuracy can be improved by detecting the replicas with higher packet delivery ratio. The next section explains the experimental setup and results.

4.8 Experimental Setup and Results

In the experimental analysis, the mobile based sensor network behaviour and its performance are analyzed with proposed iXED method. The entire experiments are conducted in a simulated environment using NS-2. This tool is widely used to study the performance of mobile ad hoc networks and wireless sensor network, etc. Figure 4.5 shows the simulated methodology.

**Figure 4.5:** Simulation Methodology of iXED Method

---

<table>
<thead>
<tr>
<th>Simulate the mobile WSN of terrain area for different parameters</th>
<th><strong>Parameters used for simulation</strong></th>
</tr>
</thead>
</table>
| Inject Node Replication Attack of 1% - 10% on the average | • Channel type  
• Routing protocol  
• Simulation time  
• Number of nodes and mobile nodes  
• Link layer and  |
| Collect the details of the later communication between nodes | • Antenna model  
• Maximum packet  
• Radio propagation model  
• Network interface type  
• MAC type  
• Traffic  
• Mobility |
| Apply XED and iXED methods | **Data collected are** |
| Evaluate the performance using the collected details | • Data packets received  
• Control packets generated  
• Sent packets  
• Sum of all packets delay  |
| Tabulate and represent graphically the performance results | • Total number of received packets  
• Total bytes received per second and total number of kilobytes |

---

Enhanced Distributed Node Replication attack detection methods in Mobile Wireless Sensor Networks using Artificial Immune System algorithms
The simulation parameters used while implementing this technique is summarized below in the Table 4.2.

**Table 4.3: Simulation Parameters for iXED Method**

<table>
<thead>
<tr>
<th>Simulation Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Propagation</td>
<td>TwoRayGround</td>
</tr>
<tr>
<td>Medium Access Control</td>
<td>802_11</td>
</tr>
<tr>
<td>X dimension of the</td>
<td>1000 (Minimum: 500, Maximum: 1000)</td>
</tr>
<tr>
<td>Y dimension of the</td>
<td>1000 (Minimum: 500, Maximum: 1000)</td>
</tr>
<tr>
<td>Adhoc Routing</td>
<td>AODV</td>
</tr>
<tr>
<td>No of nodes simulated</td>
<td>10 to 50</td>
</tr>
<tr>
<td>Connection pattern</td>
<td>Cbr10</td>
</tr>
<tr>
<td>Scenario</td>
<td>nodes50mob</td>
</tr>
<tr>
<td>Simulation time</td>
<td>500s(Minimum:200s, Maximum:10000s)</td>
</tr>
<tr>
<td>Energy</td>
<td>EnergyModel</td>
</tr>
<tr>
<td>Initial Energy</td>
<td>1000000</td>
</tr>
<tr>
<td>Aodv Minimum Neighbour</td>
<td>6</td>
</tr>
</tbody>
</table>

The simulation time varies from 500 seconds to 1000 seconds. During the simulation time, the statistics are collected. The statistics includes sum of percent energy consumed by all nodes, data packets received, control packets generated, sent packets, packet delay, sum of all packets delay and total number of received packets. Using the above statistics the following metrics are defined.

i. Packet Delivery Ratio

ii. Control Overhead

iii. Average Delay

iv. Message Drop

v. Throughput

vi. Detection Ratio

vii. False Alarm Rate
The performance of the proposed method is evaluated in terms of the above parameters

**Packet Delivery Ratio (PDR)** is defined as percentage of packets successfully received at the destinations and the total number of packets sent by the sources defined in equation (4.7).

\[
PDR = \frac{\text{Received Packets}}{\text{Sent Packets}} \times 100
\]  
(4.7)

**Overhead**: It is defined as the percentage of total numbers of control packets generated to the total number of data packets received during the simulation time given in equation (4.8).

\[
\text{Overhead} = \frac{\text{Control Packet generated}}{\text{Data Packets Received}}
\]  
(4.8)

**Average Delay**: The average delay is computed by sum of every data packet delay to the total number of received packets as defined below in equation (4.9). The parameter is measured only when the data transmission has been successful.

\[
\text{Average Delay} = \frac{\text{Sum of All Packets Delay}}{\text{Total Number of Received Packets}}
\]  
(4.9)

**Message drop**: Message drop is defined as rate of number of messages received at the destination by total number of message sent from source. It is represented by percentage (%).

\[
\text{Message Drop} = \frac{\text{Number of message received in a packet}}{\text{Total Number of Message Sent}} \times 100
\]  
(4.10)

**Throughput**: Throughput is defined as total file size transmitted in a given range. It is represented by kbps.

\[
\text{Throughput} = \frac{\text{File size}}{\text{Transmission range}}
\]  
(4.11)

**Detection ratio**: Detection ratio is defined as percentage of replica node correctly found by total number of replica nodes injected into the system. It is represented by percentage (%).

\[
\text{Detection Rate} = \frac{\text{Number of Replica Node correctly found}}{\text{Total Number of Replica node}} \times 100
\]  
(4.12)
False alarm rate: False alarm rate is defined as number of replica nodes correctly found in the total replica node by total number of replica node. It is represented by percentage (%).

\[
\text{False Alarm Rate} = \frac{\text{Total Number of Replica Node} - \text{Number of replica node correctly found}}{\text{Total Number of Replica Node}} \times 100
\]  

(4.13)

Figure 4.6: Results of XED and iXED methods for throughput

Figure 4.6 shows the throughput graph for XED and iXED; the throughput for the latter is higher than for the former.

Figure 4.7: Results of XED and iXED method for Message Drops

Figure 4.7 shows the graph of message drops for XED and iXED; the message drop for the latter is lower than for the former.
Figure 4.8: Results of XED and iXED method for Average Energy Consumed

Figure 4.8 shows the energy consumption graph for XED and iXED; the energy consumption of the latter is lesser than that of the former.

Figure 4.9: Results of XED and iXED method for Control Overhead

Figure 4.9 shows the control overhead graph for XED and iXED; the control overhead for the latter is lesser than for the former.
Figure 4.10: Results of XED and iXED method for Packet Delivery Ratio

Figure 4.10 shows the PDR graph for XED and iXED; the PDR for the latter is higher than for the former.

Figure 4.11: Results of XED and iXED method for Average Delay

Figure 4.11 shows the average delay graph for XED and iXED; the average delay for the latter is lower than for the former.
Figure 4.12: Results of XED and iXED method for Detection Ratio

Figure 4.12 shows the Detection ratio graph for XED and iXED; the detection ratio for the latter is higher than for the former.

Figure 4.13: Results of XED and iXED method for False Detection Ratio

Figure 4.13 shows the False Detection Ratio graph for XED and iXED; the false detection ratio for the latter is lower than for the former.

The performance of the proposed iXED method with the existing XED method by varying the number of nodes is given in Table 4.4.
Enhanced Distributed Node Replication attack detection methods in Mobile Wireless Sensor Networks using Artificial Immune System algorithms

Chapter 4

Table 4.4: Performance Comparison of XED and iXED methods for different node sizes

<table>
<thead>
<tr>
<th>No of Nodes</th>
<th>Control Overhead (kbps)</th>
<th>Energy Consumed (J)</th>
<th>Packet Delivery Ratio (%)</th>
<th>Average Delay (s * 10^3)</th>
<th>Message Drops (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>XED</td>
<td>iXED</td>
<td>XED</td>
<td>iXED</td>
<td>XED</td>
</tr>
<tr>
<td>50</td>
<td>5750</td>
<td>2000</td>
<td>1150</td>
<td>7500</td>
<td>92.00</td>
</tr>
<tr>
<td>100</td>
<td>1050</td>
<td>7500</td>
<td>1625</td>
<td>1100</td>
<td>84.75</td>
</tr>
<tr>
<td>150</td>
<td>1450</td>
<td>1150</td>
<td>2300</td>
<td>1600</td>
<td>79.75</td>
</tr>
<tr>
<td>200</td>
<td>1850</td>
<td>1525</td>
<td>2600</td>
<td>2250</td>
<td>75.00</td>
</tr>
<tr>
<td>Average</td>
<td>2525</td>
<td>9062</td>
<td>1918</td>
<td>1425</td>
<td>82.87</td>
</tr>
</tbody>
</table>

From the above Table 4.4, it is observed that, the proposed enhanced XED with iAIS shows a better result when compared to the XED method in terms of Control overhead, energy consumed, Packet Delivery Ratio, average delay and message drops.

The performance of the proposed method with that of the existing method by varying the number of replica nodes is given in Table 4.5.

Table 4.5: Performance Comparison of XED and iXED methods for different replica node sizes

<table>
<thead>
<tr>
<th>No of Replica Nodes</th>
<th>Detection Ratio (%)</th>
<th>False Alarm Rate (%) * 10^3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>XED</td>
<td>iXED</td>
</tr>
<tr>
<td>5</td>
<td>97.50</td>
<td>99.25</td>
</tr>
<tr>
<td>10</td>
<td>94.50</td>
<td>97.50</td>
</tr>
<tr>
<td>15</td>
<td>90.00</td>
<td>95.50</td>
</tr>
<tr>
<td>20</td>
<td>85.50</td>
<td>93.50</td>
</tr>
<tr>
<td>25</td>
<td>82.15</td>
<td>90.25</td>
</tr>
<tr>
<td>Average</td>
<td>89.93</td>
<td>95.20</td>
</tr>
</tbody>
</table>

From the above Table 4.5, it is observed that the detection ratio of the proposed iXED method with iAIS is improved. The false alarm rate is reduced to a greater extent when compared to the XED method.
The overall comparison of results for detection rate, false detection ratio, bandwidth, Message drop, Energy, Control Overhead, Average delay and the Packet Delivery Ratio is shown in the table below.

Table 4.6: Improvement of iXED method with XED method

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Existing XED Method (Kbps)</th>
<th>Proposed iXED Method (Kbps)</th>
<th>Improvement (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detection Rate</td>
<td>89.93</td>
<td>95.20</td>
<td>5.86</td>
</tr>
<tr>
<td>False Detection Ratio</td>
<td>11.69</td>
<td>7.190</td>
<td>38.39</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>28,00</td>
<td>33,00</td>
<td>17.8</td>
</tr>
<tr>
<td>Message drop</td>
<td>17.50</td>
<td>13.25</td>
<td>24.28</td>
</tr>
<tr>
<td>Energy</td>
<td>1918</td>
<td>1425</td>
<td>25.73</td>
</tr>
<tr>
<td>Control Overhead</td>
<td>2525</td>
<td>9062</td>
<td>64.10</td>
</tr>
<tr>
<td>Average Delay</td>
<td>1400</td>
<td>9500</td>
<td>32.14</td>
</tr>
<tr>
<td>Packet Delivery Ratio</td>
<td>82.87</td>
<td>86.37</td>
<td>4.82</td>
</tr>
</tbody>
</table>

The average values due to different scenarios using simulation is taken for comparison. The above Table 4.6 clearly shows the percentage of improvement achieved for various performance metrics when the proposed technique iXED is compared with existing XED. The proposed work shows improved performance in all the metrics specifically, the false detection ratio is decreased much better than other metrics.

When the XED method is combined with the iAIS algorithm, the detection accuracy of the colluding replicas increases. Moreover, the proposed method also provides a better result in terms of Packet Delivery Ratio. The computational complexity is computed for the iXED method.

4.9 Computational Complexity of the Proposed iXED Method

The computational complexity is calculated in terms of three overheads namely,

i. Storage Overhead,
ii. Communication Overhead and

iii. Computational Overhead

The calculations are as follows

### 4.9.1 Storage Overhead

The storage overhead is defined as the number of records needed to be stored in each node. During the communication, each node stores an array of length ‘l’. Hence, the storage overhead of iXED is $O(l)$.

### 4.9.2 Computational Overhead

The computational overhead is defined as the number of operations need to be executed by each node per move. It is assumed that the number of neighbours is ‘d’. In iXED, each node communicates with only one of its neighbours at any point of time. So the value of ‘d’ is 1. Hence, the computational overhead is $O(1)$.

### 4.9.3 Communication Overhead

The communication overhead is defined as the number of records needs to be transmitted by each node. In iXED, each node prepares a new random number and verifies the random number received from its neighbours. Hence, the communication overhead is $O(1)$.

### 4.10 Achievements of the Proposed iXED Method

The achievements of iXED method are

- Replicas that exchange the shared information are detected
- The detection rate is increased
- The energy consumption is low
- The Packet Delivery Ratio (PDR) is high
- The Control overheads are minimized

### 4.11 Limitations of the Proposed iXED Method

The limitations of the iXED method are
• The iXED method is ineffective when the replicas remain silent during communication.

• The time required to detect the replica has not considered while detecting the replicated node and there is no limitation for time.

4.12 Chapter Summary

In this chapter, the collusive replicas that exchange the information in the network are detected. The proposed work enhances the XED method and integrates with integrated AIS algorithm to improve the detection ratio and to reduce the false alarm rate. The proposed iXED method is energy efficient, reduces control overheads and it is suitable for deployment on identifying collusive replica nodes in mobile WSN. The experimental results shows that the proposed iXED method produces minimum false detection ratio, average delay, energy, control overhead and message drops and higher Packet Delivery Ratio value with 5.86% detection ratio than for the XED method. The achievements and the limitations of the iXED method are also discussed. In order to overcome the above mentioned limitations, node meeting based replica detection method using EEDD is proposed that is explained in the next chapter.