CHAPTER 5

ENHANCED NEGATIVE SELECTION ALGORITHM FOR MALICIOUS NODE DETECTION IN MANET

5.1 INTRODUCTION

Intrusion detection in MANETS is inevitable as these networks are dynamic, open and no central authority to monitor the nodes’ activities. Since the network is open, it allows the node to enter and exit at any point time which provides ease of access to all nodes. Misbehaving nodes utilize these opportunity to enter into the network and cause disruption. This requires to build an appropriate IDS to monitor the activities of nodes in the network. An Artificial Immune System (AIS) analogous to Human Immune System (HIS) is presented to provide appropriate IDS. The major role of AIS is to classify the samples as self (which are specific to the system) and non-self (which are the foreign body to the system) by means of proposed Enhanced Negative Selection Algorithm (ENSA). The non-self patterns act as a defense mechanism to detect the anomalies caused by invaders. ENSA regards the immune system as a classification system for matching patterns.

The proposed ENSA, which optimizes the detector generation process and performs accurate and precise classification of the network traffic is discussed. The ENSA includes two operation detector generation and classification systems. ENSA adapts Particle Swarm Optimization (PSO) technique to enhance the random detector generation to achieve maximum coverage in the non-self space. The classification involves matching the bit
strings of the antigen with the generated detectors. The strings which match with the defined detector set are classified as intruders (non-self). The performance result shows that ENSA significantly outperforms other traditional classification algorithms in terms of classification accuracy, detection rate and classification time.

5.2 EVOLUTIONARY ALGORITHMS

5.2.1 Analogy between HIS and MANET

HIS can be made analogous to a mobile ad hoc network which has similar features like decentralized network, dynamic, prone to various kinds of internal and external attack. Hence the defence mechanism which is carried by HIS can be applied to defend MANETs. Table 5.1 depicts the common features which are analogous between a HIS and a MANET.

Table 5.1 Analogy between HIS and MANET

<table>
<thead>
<tr>
<th>HIS</th>
<th>MANET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body</td>
<td>Mobile ad hoc network</td>
</tr>
<tr>
<td>Self cells/ Gene</td>
<td>Legitimate nodes</td>
</tr>
<tr>
<td>Non-Self cells/ Pathogen</td>
<td>Invader nodes</td>
</tr>
<tr>
<td>Antigen</td>
<td>Sequence of observed protocol events</td>
</tr>
<tr>
<td>Antibody</td>
<td>A set of abnormal activities observed</td>
</tr>
</tbody>
</table>

Antigens in a MANET, can be represented using variables as shown in the following Table 5.2.
Table 5.2 Representation of events in a routing protocol

<table>
<thead>
<tr>
<th>Variable</th>
<th>Event</th>
<th>Variable</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>RREQ sent</td>
<td>F</td>
<td>RERR received</td>
</tr>
<tr>
<td>B</td>
<td>RREQ received</td>
<td>G</td>
<td>Data sent</td>
</tr>
<tr>
<td>C</td>
<td>RREP sent</td>
<td>H</td>
<td>Data received</td>
</tr>
<tr>
<td>D</td>
<td>RREP received</td>
<td>I</td>
<td>Packet drop</td>
</tr>
<tr>
<td>E</td>
<td>RERR sent</td>
<td>J</td>
<td>Energy level</td>
</tr>
</tbody>
</table>

The combination of the protocol events can be used to form the gene sequence which represents the self-space. The antigen could be denoted by a gene pattern as given below

Gene 1: B in sequence
Gene 2: (B(A or C)) in sequence
Gene 3: H in sequence
Gene 4: (H (D or null ) in sequence
Gene 5: D in sequence
Gene 6: (D or G (B or A)) in sequence

5.2.2 existing work using NSA for Intrusion Detection in MANET

The misbehaving nodes in ad hoc networks interrupt the normal network routing and they degrade the networks’ performance. These nodes can drop or modify packets which are sent between source and destination nodes. Several works in literature have focused on providing secure communication in the network using AIS techniques by Khannous et al. (2014b), Mohamed & Abdullah (2009).
The Negative Selection Algorithm (NSA) is one of the vital techniques given by Ji & Dasgupta (2007). The NSA imitates the self or non-self-discrimination process which takes place in HIS with the help of detectors also known as antibodies illustrated by Yiqing et al. (2013). NSA aids in identifying the legitimate and illegitimate nodes by the detector generation process in MANET. The detector/antibody in MANET will be similar to the activities performed by the misbehaving node which aids in classifying the nodes’ activities into self or normal pattern and non-self or abnormal pattern given by Khannous et al. (2014a). Idris et al. (2015) carried out the detector generation process using the evolutionary algorithm named PSO with NSA for the detection of email spam. In this method local maximum is considered for generating detectors using the PSO algorithm which helps in creating a large number of antibodies.

Govindan & Mohapatra (2012) proposed trust computation for establishing secure route between the source and destination. This approach faced a higher computational complexity of computing trust value for each node in the network. Also there are various works which involve cryptographic techniques for providing integrity, confidentiality and other security needs among the nodes in the network as given by Zhao et al. (2012). However, these techniques require a central administrator to manage the members, for exchanging keys.

An Adaptive Routing Strategy (ARS) with an immune system concept was proposed by Yiqing et al. (2013). It provides a secure and efficient routing with low network overhead. ARS provides varied routing protocols depending on the network condition. The network condition is based on three measurements: connectivity, unpredictability and security. There exist a trade-off between efficient routing and security. Security is
enforced whenever there is a need to monitor a node by setting a threshold value. This system leads to higher false positive and false negatives.

AIS techniques were combined in previous works to identify the intrusive nodes and providing appropriate response measures. Khannous et al. (2014a) proposed a technique which combines NSA with danger theory to improve the accuracy in detection and minimizes false positive. NSA and danger theory combine T cell concept in the negative selection algorithm with the dendritic cells to confirm whether a suspicious node is an attacker or non-attacker by processing the alarm signal generated from the IDS. Sarafijanovic & Boudec (2005) proposed a combination of two techniques in AIS, danger signal and clonal selection for misbehaviour detection. Clonal selection produces a large number of pathogens (detectors) which gives positive feedback in results. But the response time for finding the intruder is increased.

AIS technique was also implemented by deploying agents in the network discussed by Ye & Li (2010); Mitrokotsa & Dimitrakakis (2013). These authors proposed two types of agents to provide immunity: they are detection agent and counterattack agent. The detector agent performs monitoring over the network and detects any abnormal behaviour as analogous to the T cells in HIS. This agent is responsible for taking appropriate response action. The counterattack agents are similar to the antibody generated in response to any invader. It may surround the invader and cause death. This method uses the pattern matching technique to identify abnormality. Barani (2014) used immunological principles to design multi-agent security architecture. Various agents are used to provide communication among nodes, to make decision on detecting abnormal pattern and to provide appropriate response. Three agents are deployed in the network: monitor agent, decision agent, and killer agent. Monitor agent is regarded as similar to the T cells of the body. Decision agent gathers information from the monitor
agent. If the collected information is present in the immune memory, then, the response will be faster. If new information is collected, then, decision must be made to identify whether it belongs to normal or abnormal behaviour. Killer agent performs operations like moving near the invader, locating its position, isolating the invader and if it cannot locate the invader or the invader dies then the agent commits suicide. There were several evolutionary algorithms are used in detector generation process. Gao et al. (2007) proposed a scheme, where the NSA detectors are optimized by the PSO to collectively occupy the maximal coverage of the non-self-space so that they can achieve the best anomaly detection performance. It also improves the detection rate by generating non-overlapping detectors. However, it does not handle any partial overlap between the detector and the self-space. Considering only non-overlap between the detector and the self-space leads to lesser detection of anomaly as the anomalous pattern may have a few features similar to that of the self-space.

5.2.3 Motivation from Existing System

The detector generation and misbehaving node identification processes of NSA in the existing works on the principle given by Sarafijanovic & Boudec (2005); Khannous et al. (2014a) and Barani (2014) opted the same matching rule which has led to higher false positive and false negative detection rates. Also the detector generation process incurs detectors that do not cover a large amount of the non-self-space causing the anomalous patterns to escape and there is higher overlap among the generated detectors and the self-space patterns. Thus, the overlap can cause the normal behaviour of the network to be classified as abnormal.

As the detector plays a major role in locating the abnormal behavior generated by the misbehaving node in the network. The proposed ENSA inherits the basic steps in NSA with improved detector generation process and
a learning mechanism to classify the nodes into legitimate and misbehaving. The improvement in detector generation is performed to cover the maximum non-self-space patterns (i.e., the misbehaving nodes behavior which forms the abnormal part of the training set) with minimal overlap among the detectors and the self-space patterns. This scheme considerably reduces the time to identify the malicious nodes and shows higher classification accuracy to enhance the network security.

5.3 ENHANCED NEGATIVE SELECTION ALGORITHM

ENSA performs the detection of illegitimate nodes in a two-step detection mechanism. The first step, detector generation, generates the detector patterns which recognize the abnormal behaviour of the misbehaving nodes in the network and the second step is malicious node identification by categorizing the nodes’ activities into normal and abnormal behavior. The detector generation is carried out using the PSO algorithm. The antibody generation process takes the legitimate nodes activities which form the self-space patterns and the initial set of detectors as input. The valid detector set or the antibodies obtained after detector generation will be similar to the abnormal activities performed by the misbehaving nodes. The detailed design of ENSA is depicted in Figure 5.1. Feature Space (FS) is an $n$-dimensional hyperspace where attributes $[a_0, a_1, \ldots, a_n] \in A$ which can be mapped to feature vector $[F_0, F_1, \ldots, F_n] \in FS$ is also called vector or configuration space. The attributes denote the vital properties of the network traffic which are capable of distinguishing between normal and abnormal patterns. Initially, we assume that the network is devoid of malicious nodes which yield the normal self-space of the network. ENSA uses a binary feature space of $n$-dimension which can be formed using an inference system.
Figure 5.1 Design of enhanced negative selection algorithm
There are many rules written in the inference system based on which the binary patterns are generated. The rules are formed from the activities performed by the nodes in the routing layer of the network. Activities can be based on the event, node’s energy level, layer on which communication happens, packet drop reason, etc. Some of the rules formed are:

Rule 1: if (a node sends RREQ) then A = ‘001’
Rule 2: if (a node sends RREP) then C = ‘010’
Rule 3: if (a node receives RREQ) then B = ‘011’
Rule 4: if (a node receives RREP) then D = ‘100’
Rule 5: if (a node drops packet due to loop) then I = ‘0001’
Rule 6: if (node drops packet due to collision) then I = ‘0010’
Rule 7: if (node drops packet due to TTL expiry) then I = ‘0101’
Rule 8: if (node drops due to low energy) then I = ‘0111’

There are still more packet drop reasons which form further rules forming the feature space in MANET analogous to the feature space in NSA. Let the self-space patterns formed from the above rule be $S$, where $S$ is defined as follows

$$S = (s_1...s_n) = \begin{bmatrix} s_{11} & \cdots & s_{1m} \\ \vdots & \ddots & \vdots \\ s_{m1} & \cdots & s_{nm} \end{bmatrix}$$

where $S_{ij} \in FS, i = [1,\ldots,n]$ and $j = [1,\ldots,m]$ and $n$ is the total number of patterns generated, $m$ is the total number of attributes. The self-space $S$ is normalized as

$$S_i = \frac{s_i}{|S_i|}$$
Thus, $s_i$ is the $i^{th}$ self-space pattern and $s_{ij}$ is the $j^{th}$ attribute value of $i^{th}$ self-space pattern.

### 5.3.1 Detector Generation Process

The random set of detectors is generated using random detector generator which is similar to random number generation for generating the binary numbers. The random detectors are as follows:

$$d = (d_1 ... d_n) = \begin{bmatrix} d_{11} & \ldots & d_{1m} \\ \vdots & \ddots & \vdots \\ d_{kn} & \ldots & d_{km} \end{bmatrix}$$

where $n$ is the total number of patterns generated, $m$ is the total number of attributes considered. The detectors act as antibodies present in AIS which are then fine-tuned by applying PSO algorithm. PSO algorithm helps in producing optimized solution from iteratively solving a given set of candidate solution. The generated detectors must be in maximum number to cover the large non-self-space denoting pathogens. The main objective of ENSA is to generate valid detectors that must satisfy two main features such as maximizing non-self-space coverage and minimizing overlap among the detectors and self-space. The algorithm involves the particles to traverse the initial detector set to find out the valid detectors which satisfy the local maxima. Each particle is moved across the randomly generated detectors by changing its position $p$ and velocity $v$ until the particle finds an appropriate detector which is similar to the anomalous behavior of the misbehaving node. The position of particle $i$ at iteration $(t + 1)$ is determined using the sum of its position and velocity at iteration $t$. The velocity of particle $i$ at iteration $(t + 1)$ is determined using the sum of velocity at iteration $t$ and difference between the local best position ($p_L$) and current best position ($p_i$) of the particle. At the $i^{th}$ iteration the position and velocity of the particle is represented as,
\[
p_i(t + 1) = p_i(t) + v_i(t)
\]
\[
v_i(t + 1) = v_i(t) + c \times r \left[ p_i - pL_b^i(t) \right]
\]

where \( c, r \in [0, 1] \) denotes random value to control the value \( v \). The detectors which satisfy the fitness function are screened to form the valid detectors. The Fitness function \( FN \) is defined to be the self-space pattern \( SS \). The initial detectors, \( DS \) which do not match the self-space patterns within the threshold, \( T_v \) in \( FN \) are retrieved to form the candidate detectors \( DC \). Hamming distance is used as the similarity measure between \( SS \) and \( DS \). The distance between the self-space \( S_i \) and the detector \( d_i \) can be defined as

\[
H_s(s_i, d_i) = (s_{i1} \oplus d_{i1}) \cdot (s_{i2} \oplus d_{i2}) \ldots (s_{im} \oplus d_{im})
\]

The candidate detectors are then chosen based on the threshold, \( T_v \). The similarity value, \( \theta \), given in Equation (5.1), is the deciding factor for selecting a detector/antibody.

\[
\theta = H_s(s_i, d_i) - T_v \quad (5.1)
\]

For each detector, \( d_i \) which does not match the self-space \( S_i \) the \( \theta \) value will be greater than 0. Hence, the detector threshold should be minimal so that only detectors which do not match self-space are selected. \( T_v \) is updated after each iteration.

\[
T_v(i) = \min(\theta) \text{ if } (\theta \geq 0) \quad (5.2)
\]

where \( T_v(i) \) is the threshold value obtained for \( i^{th} \) detector pattern. The candidate sets of detectors are obtained after the matching processes are expanded to form the valid detectors \( DV \) based on the condition that certain combination of bits must be 1. For example, if the candidate detector is \( d_{ij} \in DC \), then, the condition can be represented as
\[ d_{ij} = (1110110111) \quad (5.3) \]

where \( d_{ij} \) must contain at least one bit as 1 \( c \in [5 \ldots 8] \). The \( c \) bits denote the fields corresponding to the abnormal activity caused by the misbehaving nodes. The valid detectors help to cover most of the non-self-space patterns \( SN \). Algorithm 5.1 shows how the detectors are generated.

**Algorithm - 5.1. Detector generation**

// \( pL \) is the local best
// \( F_N \) is the fitness function
// \( p \) and \( v \) are the particles position and velocity
// \( P \) is the population size
// \( D_V \) is the valid detector set

Generate random detector set, \( D \)

Initialize all the particles initial \( p \) and \( v \) value

Repeat

for each detector \( i \) in \( D \) do

Compute the fitness function as given in Equation (5.2) and (5.3)

If particle satisfies \( F_N \) then

Add corresponding \( d_i \) to \( D_V \)

end if

end for

// Update each particle \( p \) and \( v \) value based on \( pL \)

for each particle \( i \) in \( P \) do

for each detector \( j \) in \( D \) do

\[ v_i(t+1) = v_i(t) + c \times r \times [p_i - pL_i(t)] \]

\[ p_i(t+1) = p_i(t) + v_i(t) \]

end for

end for

While the maximum iteration or the stopping criteria is attained
5.3.2 Malicious Node Identification

The generated detectors and self-patterns obtained from the network activity without misbehaving nodes are given as input for retrieving the patterns which does not belong to the self, i.e., the patterns which match the detectors are said to form the anomalous pattern. We utilize a supervised learning algorithm for classifying the protocol sequence into normal or abnormal pattern as given by Mitrokotsa & Dimitrakakis (2013); Gonzalez et al. (2002) and Govindan & Mohapatra (2012). The classification between two classes normal \( (N) \) and abnormal \( (AN) \) is performed using Support Vector Machine (SVM) classifier. It separates all data points from the origin and maximizes the distance from the hyperplane to the origin. The hyperplane can be described by

\[
wx + b = 0
\]

where \( w \) is the normal to the hyperplane and \( \frac{b}{||w||} \) is the perpendicular distance from the hyperplane to the origin. During the learning phase, the classifier is trained using the dataset \( Train = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \) where \( x_i \) is the \( i^{th} \) input data points; \( x_i \in [DV, SS] \) and \( y_i \in [N, AN] \) is the \( i^{th} \) label for the corresponding input pattern. The training data is represented as

\[
y_i(x_i.w + b) - 1 \geq 0 \forall i
\]

The points which lie on two planes can be described as

\[
x_i.w + b = N \text{ for patterns which belong to } SS
\]

\[
x_i.w + b = AN \text{ for patterns which belong to } SN
\]
To obtain an optimum classification, SVM classifier is trained using huge possibility of both positive and negative samples. As MANET is vulnerable to many unknown attacks, the classifier must be well trained to distinguish between normal and abnormal pattern to obtain the correct positive and negative samples. The candidate detector is, then, expanded to form many valid detectors using fitness function, which thus helps in covering huge non-self-space with less overlap among detectors and self-space. The performance of ENSA with other existing systems is based on the following metrics: Estimated coverage, classification accuracy, relevance, precision and recall. ESNA gives more significant amount of accuracy than the other system.

\[
\text{Train} = \begin{bmatrix}
    x_1 \\
    \vdots \\
    x_n
\end{bmatrix}
\begin{bmatrix}
    y_1 \\
    \vdots \\
    y_n
\end{bmatrix}
= \begin{bmatrix}
    s_{11} & \cdots & s_{1m} \\
    \vdots & \ddots & \vdots \\
    s_{m1} & \cdots & s_{nm} \\
    d_{11} & \cdots & d_{1m} \\
    \vdots & \ddots & \vdots \\
    d_{k1} & \cdots & d_{km}
\end{bmatrix}
\begin{bmatrix}
    N \\
    \vdots \\
    N
\end{bmatrix}
\begin{bmatrix}
    N \\
    \vdots \\
    N
\end{bmatrix}
\begin{bmatrix}
    \text{AN} \\
    \vdots \\
    \text{AN}
\end{bmatrix}
\]

The test data is formed from the network behavior with both legitimate and misbehaving nodes. The tuples in the test dataset \( t_i \) is similar to the tuples of the training dataset but with no class labels. The position of the pattern is located in the obtained test dataset, TD. After locating the pattern in the dataset, the field corresponding to the node \( ID, N_{id} \) is retrieved. The \( N_{id} \) corresponds to an internal malicious node which causes abnormality. The operation of ENSA can be explained with help of an example. A sample self-set/self-antigen can be as follows:

\[ s_1 = (1 010 00001) \quad \text{and} \quad s_2 = (1 000 00001) \]
The random detector set is shown as

$$DS = \begin{bmatrix} 1001 00101 \\ \vdots \\ 1110 00001 \end{bmatrix}$$

We see that $d_1$ in $DS$ matches $s_1$ with the $\theta$ value to be less than 0. Hence, $d_1$ is eliminated. Whereas $d_2$ does not match $s_1$ or $s_2$, i.e., it does not form any of the self-antigen as the $\theta$ value is greater than 0. Therefore, $d_2$ is accepted as the candidate detector/antibody. The candidate detector is, then, expanded to form many valid detectors using fitness function and thereby helps in covering huge non-self-space with less overlap among detectors and self-space. The example for valid detector/antibody is shown as

$$d_1 = (111110101)$$
$$d_2 = (100111110)$$

We know that $(s_1, s_2) \in N$ and $(d_1, d_2) \in AN$. The example test dataset can be denoted as

$$t_1 = (111011011)$$
$$t_2 = (110010001)$$

The given test dataset is finally classified. The result is $t_1 \in N$ and $t_2 \in AN$. Thus ENSA provides a classification result with reduced false positive and false negative rates.
5.4 SIMULATION RESULTS

5.4.1 Description of Simulation

5.4.1.1 Experimental setup

ENSA is implemented in MATLAB. To measure the protocol events NS2 simulator is used, the network is simulated with 50 nodes. Initially the network is assumed to be devoid of misbehaving nodes. Hence, the patterns generated correspond to the self-space or the self-antigen or genes. Table 5.3 shows the default values for the system parameters which are selected from numerous simulation runs.

Table 5.3 System parameters with default value

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximal number of self-antigens/self-space patterns collected for detector generation</td>
<td>420</td>
</tr>
<tr>
<td>Maximal number of detectors/antibodies generated</td>
<td>100</td>
</tr>
<tr>
<td>Maximal time for creating feature space/genes</td>
<td>500s</td>
</tr>
<tr>
<td>Accepted deviation threshold for similarity measure, $T_v$</td>
<td>0.5</td>
</tr>
<tr>
<td>Maximum time for generating detectors</td>
<td>30s</td>
</tr>
<tr>
<td>Maximum time for classification of nodes</td>
<td>10s</td>
</tr>
<tr>
<td>Accepted classification error ratio</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

The detectors are generated considering the self-space patterns obtained during the learning phase of initial network simulation. In the detection phase the detectors identify the misbehaving nodes by classifying the node’s activities based on its normal and abnormal activities.
5.4.1.2 Misbehaving node’s behavior

The misbehaving node performs various illegitimate activities like dropping packets, modifying packets, replaying, etc. The node’s misbehaviour characteristic involves non-forwarding the control and data packets, packet dropping and misroutting.

5.4.1.3 Performance metrics

The performance of ENSA is compared with other existing systems by Sarafijanovic & Boudec (2005) and Barani (2014) based on the following metrics: Estimated Coverage denotes how well the detectors/antibodies cover the non-self/abnormal space without any overlapping among the antibody and the self-antigen. Classification Accuracy is defined as identifying the correct normal and correct abnormal pattern in the network traffic. It can be computed using true positive ($T_p$), true negative ($T_n$), false positive ($F_p$) and false negative ($F_n$) detection. The true positive detection is the number of actual malicious nodes in the network which is correctly classified as malicious. The true negative detection is the number of actual legitimate nodes in the network which is correctly classified as legitimate. The false positive detection denotes the number of legitimate nodes in the network, which are classified as misbehaving nodes whereas the false negative detection denotes the number of misbehaving nodes in the network are classified as legitimate. The value of $T_p$ and $T_n$ should be as high as possible whereas the value of $F_p$ and $F_n$ should be as low as possible. Classification time is defined as the time past the detector generation process is complete and the overall time required for classifying the nodes activities.
5.4.2 Performance Evaluation

In the simulation runs, the misbehaving nodes are detected and classified as malicious by the neighbour nodes. The impact on classification accuracy lies mainly on the detectors/antibodies which can be generated using various techniques. The estimated coverage space of ESNA is compared with other existing techniques as shown in Figure 5.2. Figure 5.3(a) shows the true positive detection of the malicious node as malicious. The proposed ENSA identifies the correct malicious nodes at an average of 12 nodes with threshold value to be 0.5 with default detector count which is better than other techniques from Sarafijanovic & Boudec (2005) and Barani (2014).

![Figure 5.2 Comparison of estimated coverage between ENSA and NSA](image)

**Figure 5.2 Comparison of estimated coverage between ENSA and NSA**

Similarly, the true negative detection of the legitimate node as legitimate is shown in Figure 5.3(b). ENSA obtains on an average of 13 nodes compared to existing methods.
The false negative and false positive detection are shown in Figure 5.4(a) and 5.4(b). The proposed ENSA performs better than the existing techniques, with an average $F_p$ and $F_n$ values for ENSA at 24% and 22.64%.

The false detection rate obtained for 300 detectors with the threshold value 0.5, the ENSA obtained 23.32% and other techniques obtained 55.34% and 42.67%. This shows the significance of PSO in improving the detector generation process of ENSA. Hence ESNA results in reduced false detection rate. Table 5.4 shows the comparison of the relevance of the classifier used in ENSA and existing techniques. The precision, $p$ and recall, $r$ value are calculated to know the relevance of the classifier. Precision, also known as positive predictive value is defined as the fraction of the classified node that is relevant. It can be calculated using Equation (5.4).

![True positive detection of misbehaving nodes](image)

**Figure 5.3(a)** True positive detection of misbehaving nodes
Figure 5.3(b) True negative detection of legitimate nodes

Figure 5.4(a) False positive detection of legitimate nodes
Figure 5.4(b) False negative detection of misbehaving nodes

\[ p = \frac{T_p}{T_p + F_p} \]  

(5.4)

Recall also known as sensitivity, is defined as the fraction of relevant nodes that are classified.

\[ r = \frac{T_p}{T_p + F_n} \]

The F-measure is also computed for the proposed ENSA. It gives the measure of classification accuracy and is measured in terms of \( p \) and \( r \) as

\[ F_1 = 2\cdot \frac{p \cdot r}{p + r} \]
The comparison of classification accuracy and classification time between ENSA and existing techniques is shown in Figure. 5.5. The proposed model outperforms existing techniques given by Sarafijanovic & Boudec (2005) and Barani (2014). ENSA obtained 81.4% classification accuracy while NSA with clonal selection approach attained 46.8% and GAAIS obtained 60.31%. Thus, these findings prove that the proposed ENSA provides an accurate and precise classification result.

Table 5.4 Comparison of precision, recall and F-measure values between ENSA and NSA

<table>
<thead>
<tr>
<th>Precision</th>
<th>Technique / No. of Nodes</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSA with Clonal Selection</td>
<td>0.6</td>
<td>0.5</td>
<td>0.53</td>
<td>0.4</td>
<td>0.28</td>
<td></td>
</tr>
<tr>
<td>ENSA</td>
<td>1.0</td>
<td>0.8</td>
<td>0.8</td>
<td>0.75</td>
<td>0.72</td>
<td></td>
</tr>
<tr>
<td>GAAIS</td>
<td>0.4</td>
<td>0.55</td>
<td>0.66</td>
<td>0.7</td>
<td>0.8</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recall</th>
<th>NSA with Clonal Selection</th>
<th>0.5</th>
<th>0.5</th>
<th>0.53</th>
<th>0.44</th>
<th>0.33</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSA</td>
<td>0.83</td>
<td>0.8</td>
<td>0.85</td>
<td>0.83</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>GAAIS</td>
<td>0.66</td>
<td>0.63</td>
<td>0.66</td>
<td>0.57</td>
<td>0.48</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>F-measure</th>
<th>NSA with Clonal Selection</th>
<th>0.64</th>
<th>0.5</th>
<th>0.53</th>
<th>0.42</th>
<th>0.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSA</td>
<td>0.91</td>
<td>0.8</td>
<td>0.82</td>
<td>0.78</td>
<td>0.73</td>
<td></td>
</tr>
<tr>
<td>GAAIS</td>
<td>0.5</td>
<td>0.59</td>
<td>0.66</td>
<td>0.63</td>
<td>0.55</td>
<td></td>
</tr>
</tbody>
</table>
5.5 CONCLUSION

ENSA technique is proposed to locate malicious nodes in the network. ENSA is built to work in MANET environments, where the network is prone to huge variety of security threats due to the dynamic and open nature. ENSA provides an improved detector generation process which aids in covering most of the anomalous behavior patterns exhibited by misbehaving nodes in the network.