CHAPTER 7

PREDICTIVE CLUSTER BASED DISTRIBUTED HIERARCHICAL KEY MANAGEMENT SCHEME FOR MANET

CONTENTS

7.1 Overview ........................................................................................................................................ 172
7.2 Prediction Technique ...................................................................................................................... 173
  7.2.1 Based on Route Expiration Time ....................................................................................... 173
  7.2.2 Based on Link Stability .................................................................................................. 176
7.3 Weighted Clustering Algorithm .................................................................................................... 178
  7.3.1 Cluster Formation ........................................................................................................ 178
7.4 Effect of Node Mobility ................................................................................................................. 180
7.5 Simulation Results ......................................................................................................................... 180
  7.5.1 Simulation Model and Parameters .................................................................................. 180
  7.5.2 Performance Metrics ..................................................................................................... 182
  7.5.3 Results .......................................................................................................................... 183
7.6 Conclusion ...................................................................................................................................... 187

7.1 Overview

In this chapter, an improved progressive key management scheme utilizing a stable and power proficient cluster management procedure has been proposed. The mobility prediction strategy is combined in the proposed hierarchical key management scheme. The method predicts the node movement and sends information if there should arise an occurrence of cluster movement. The consolidated metric for prediction is evaluated taking into account route expiration time and node velocity. Every cluster head
holds the public key of its part nodes just and go about as a router when managing nodes of other cluster individuals. Utilizing this procedure, the overhead on centralized server is decreased. Also, the need of every node putting away all public keys is lessened consequently minimizing the stockpiling overhead on every node. By Simulation results, it is demonstrated this scheme accomplishes better delivery ratio and resilience with lessened delay and overhead.

7.2 Prediction Technique

Here we are going to predict the mobility based on the route expiration time and link stability of individual links. A probabilistic approach is used to find out the link availability. Link stability is going to be calculated based on the transmission range and distance travelled. The combined metric of these two will be the mobility prediction metric.

7.2.1 Based on Route Expiration Time

The $T_{RE}$ is the minimum time selected from a set of link expiration times ($T_{LE}$)s designed for the sake of sufficient path. The time period between nodes is given as $T_{LE}$. Hence, the minimum value of $T_{LE}$ attained in each path and the maximum number of $T_{RE}$ is selected which is representing the reliable routing path [107].

$$T_{RE} = \text{Min} (T_{LEs}) \quad (7.1)$$

Thus for the feasible path $T_{RE}$ is the maximum value among $T_{LE}$ s. This link availability is going to be predicted based on the probability distribution theory ($T_{pr}$).

Global positioning system is the method used to achieve the principle of $T_{LE}$ which is for estimating future disconnection time with the help of two neighbors in
motion. It determines the movement parameters of two neighboring nodes. The assumptions made are as follows:

- Signal strength of free space propagation model is exclusively depended on the distance to the transmitter.
- GPS clock helps all nodes to synchronize themselves with their clock values.

By having knowledge of the motion parameters of two nodes, the time period for the nodes can be calculated. These parameters obtained from GPS include speed, direction, and radio range.

On the continuously available time for an active link between two nodes the link expiration time at time \( T_0 \) with a given prediction \( T_{pr} \), the link availability is defined as,

\[
L(T_{pr}) = P\{T_0 \to T_0 + T_{pr} \mid \text{Available at } T_0\} \tag{7.2}
\]

Here if we denote link availability of a node by \( L_A \), then it is seen that \( L_A \) follows an exponential distribution \( L_A(x) = \lambda e^{-\lambda x} \), for \( x \geq 0 \).

Hence from equation (7.2) it follows that \( L(T_{pr}) = P\{L_A > T_0 + T_{pr} \mid L_A > T_0\} \)

\[
= P\{L_A > T_{pr}\}, \text{ by Memory less property of exponential distribution}
\]

\[
= \int_{T_{pr}}^{\infty} \lambda e^{-\lambda x} dx, \text{ as Link availability follows an exponential distribution}
\]

\[
= \lambda \left[ \frac{e^{-\lambda x}}{-\lambda} \right]_{T_{pr}}^{\infty} = -\lambda \left[ e^{-\lambda T_{pr}} \right]_{T_{pr}} = -\lambda \left[ 0 - e^{-\lambda T_{pr}} \right] = e^{-\lambda T_{pr}} \tag{7.3}
\]

Also we have \( L(T_{pr}) = P\{L_A > T_0 + T_{pr} \mid L_A > T_0 + T_{pr}\} \)

\[
= P\{L_A > T_{pr}\}, \text{ by Memory less property of exponential distribution}
\]

\[
= 1 - P\{L_A \leq T_{pr}\}, \text{ by complément probability} \tag{7.4}
\]
\[1 - \int_{0}^{T_{pr}} \lambda e^{-\lambda x} \, dx = 1 - \lambda \left[ \frac{e^{-\lambda x}}{-\lambda} \right]_{0}^{T_{pr}} = 1 + \left[ e^{-\lambda T_{pr}} \right]_{0}^{T_{pr}} = 1 + 1 - e^{-\lambda T_{pr}} = e^{-\lambda T_{pr}} (7.5)\]

We usually denote \( P[L_{A} \leq T_{pr}] \) as the distribution function \( F(T_{pr}) \).

Thus From (7.4), we obtain \( L(T_{pr}) = 1 - F(T_{pr}) \) (7.6)

This indicates the probability of link availability existing from \( T_{0} \) to \( T_{0} + T_{pr} \).

The calculation of \( L(T_{pr}) \) can be divided into two parts:

\( L_{1}(T_{pr}) \): the link availability when the velocities of the two nodes keep unchanged between \( T_{0} \) and \( T_{0} + T_{pr} \),

\( L_{2}(T_{pr}) \): the one for the other cases

(i.e.) \( L(T_{pr}) = L_{1}(T_{pr}) + L_{2}(T_{pr}) \) (7.7)

Calculation of \( L_{1}(T_{pr}) \), which is equal to the probability that the epochs from \( t_{0} \) onwards for the two nodes are longer than \( T_{pr} \) because \( T_{pr} \) is an accurate prediction if the movements of the two nodes keep unchanged. Since node movements are independent of each other and exponential distribution is memory less, \( L_{1}(T_{pr}) \) is given by

\[ L_{1}(T_{pr}) = \text{link availability of the first node} \times \text{link availability of the second node} = (\text{link availability of the first node})^{2} \text{ (using the fact that the probability that the epochs from } t_{0} \text{ onwards for the two nodes are longer than } T_{pr} \text{ and because } T_{pr} \text{ is an accurate prediction if the movements of the two nodes keep unchanged)} \]
\[ L(T_{pr})^2 = \left( e^{-\lambda T_{pr}} \right)^2 \] (by using the fact that nodes’ movements follow an exponential distribution and exponential distribution is ‘memory less—see equation (7.3 or 7.5))

\[ L(T_{pr})^2 = e^{-2\lambda T_{pr}} \]

We can also see by equation (7.6) that \[ L(T_{pr}) = [1 - F(T_{pr})]^2 \]

Thus \[ L_1(T_{pr}) = [1 - E(T_{pr})]^2 = e^{-2\lambda T_{pr}} \] (7.8)

Where, E and F are probability functions. However, it is complicated to give an accurate calculation for \[ L_2(T_{pr}) \] because of the difficulties in learning changes in link status caused by changes in a node’s movement. Here we are only considering the link availability when the velocities of two nodes keep unchanged.

### 7.2.2 Based on Link Stability

Link stability in terms of link expiration time is defined as maximum time connectivity between any two neighbor nodes. For calculating the link expiration time, it is assumed that motion parameters of any two neighbors are known [108].

Let \( n_1 \) and \( n_2 \) be two nodes within the transmission range \( r \) and \( (x_1, y_1) \) and \( (x_2, y_2) \) be the coordinate for node \( n_1 \) and \( n_2 \) with velocity \( v_1 \) and \( v_2 \) and direction \( \theta_1 \) and \( \theta_2 \) respectively.

![Fig 7.1 Distance Calculation](image-url)
After a time interval $t$ the new coordinate will be $(x'_1, y'_1)$ for $n_1$ and $(x'_2, y'_2)$ for $n_2$. For a time $t$, let $d_1$ and $d_2$ be the distance traveled by node $n_1$ and $n_2$. $d_1$ and $d_2$ are calculated using the following formula: distance = velocity * time

$$d_1 = v_1 t$$

$$d_2 = v_2 t$$

(7.9) (7.10)

Referring to the figure above, new coordinates (with respect to old coordinates) can be calculated as

$$x'_1 = x_1 + d_1 \cos \theta_1 = x_1 + v_1 t \cos \theta_1$$

(7.11)

$$y'_1 = y_1 + d_1 \sin \theta_1 = y_1 + v_1 t \sin \theta_1$$

(7.12)

$$x'_2 = x_2 + d_2 \cos \theta_2 = x_2 + v_2 t \cos \theta_2$$

(7.13)

$$y'_2 = y_2 + d_2 \sin \theta_2 = y_2 + v_2 t \sin \theta_2$$

(7.14)

Distance $D$ between two nodes at time $t$ can be obtained from:

$$D = \sqrt{(x'_1 - x'_2)^2 + (y'_1 - y'_2)^2}$$

$$= \sqrt{[x_1 + v_1 t \cos \theta_1 - x_2 - v_2 t \cos \theta_2]^2 + [y_1 + v_1 t \sin \theta_1 - y_2 - v_2 t \sin \theta_2]^2}$$

$$= \sqrt{(x_1 - x_2)^2 + t^2 [v_1 \cos \theta_1 - v_2 \cos \theta_2]^2 + (y_1 - y_2)^2 + t^2 [v_1 \sin \theta_1 - v_2 \sin \theta_2]^2}$$

(7.15)

When the distance between two nodes becomes larger than the transmission range the nodes will be disconnected. For transmission range $r$, link stability $L_{\text{stab}}$ between any two nodes overtime period $t$ can be calculated by:

$$L_{\text{stab}} = \frac{r}{D}$$

(7.16)

$L_{\text{stab}}$ is the link stability of individual links between any two nodes and for a path, it is a concave parameter (as $L_{\text{stab}}$ tends to zero either when transmission range $r$ too small
or distance between two nodes becomes too large) and it is same as the minimum link stability along the path.

Finally, the combined metric for mobility prediction is given by

\[ CP = L_1 (T_{Pr}) + L_{stab} \]  \hspace{1cm} (7.17)

7.3 Weighted Clustering Algorithm [58]

7.3.1 Cluster Formation

Let \( n \) be the maximum allowable number of members of cluster. Let \( c \), a counter maintained by each node. When a new node joins a cluster then the cluster head is going to check with the counter value, if \( c \) is less than \( n \), then the counter value is going to be incremented. If \( c \) is greater than or equal to \( n \) then, that node cannot be able to join in the cluster [109].

Each node maintains a table, which contains the information about its neighbors. Each cluster heads are also going to maintain a table, containing the details of all other cluster heads.

Each node is going to calculate a weight value depends upon four parameters. The degree of difference \( \delta_i \), Sum of distance to its neighbors \( (d_n) \), Average speed of every node \( (t_c) \) and the remaining battery power \( (p) \). The cluster head is going to be selected depends upon the weight value.

The degree of difference, \( \delta_i = |d_i - N| \), where \( d_i \) is the number of neighboring nodes within the transmission range and \( N \) is the maximum cluster size. The sum of distance of every node is calculated by \( D_n = \sum \text{dst} (n, n') \), which is the sum of distance from a node to its neighbors. Average speed of every node is calculated by using,
\[
Avsi = \frac{1}{t} \sum_{t_i \geq t} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}, \text{ where } x_t, y_t \text{ are the coordinates of node at time } t.
\]

The remaining battery power \( P \) is calculated by considering how much battery power has been consumed, the different kinds of roles like cluster heads or ordinary nodes. The weighted sum of these four factors are going to taken into account for calculating the combined metric of node weight.

\[
W_n = (w_1 \delta_i) + (w_2 d_n) + (w_3 Avsi) - (w_4 P_n)
\]  

(7.18)

HELLO messages are used to update the node and cluster tables. Each hello message contains the state of the node which is periodically exchanged between CHs or between each CH and its members. Before considering the cluster maintenance procedure, it is necessary to describe the process by which the node is able to compute its weight and several metrics under consideration. Depending upon the weight values of the node, the cluster head is elected.

The node with the smallest \( W_n \) is elected as a cluster-head. All the neighbors of the chosen cluster-head are no more allowed to participate in the election procedure.

All the above calculations are repeated for remaining nodes which is not yet elected as a cluster-head or assigned to a cluster.

In order to update the node_tables and CH_tables, node periodically calculates its weights and sends hello messages to its members and to the neighboring CHs. CH monitors the communication channel whether it hears any HELLO message or leave message. When the CH receives a leave message, it updates the node_table and broadcasts a HELLO message to its members and to its neighboring cluster heads. When the CH receives a HELLO message from a neighboring CH, it updates the CH_table. If
HELLO’s source is a node member, CH updates a node_table and verifies the weight. In case of lowest weight, the CH must invoke the re-election procedure.

7.4 Effect of Node Mobility

In the event that a node moves starting with one cluster then onto the next cluster, both the cluster heads (cluster in which the node leaves and the cluster in which the node joins) need to know details of the moving node. Consider node 19 of cluster C moves to cluster D (Figure 6.4), then both the cluster heads (CHC and CHD) needs to think about the movement.

Depends upon the mobility prediction, the corresponding updates are going to be done in each cluster heads, before the movements of the nodes. By using this mobility prediction technique, we can be able to predict the movements and depends upon that, the corresponding changes are going to be made in each cluster.

7.5 Simulation Results

7.5.1 Simulation Model and Parameters

Network Simulator (NS2) is utilized to simulate the proposed algorithm. In the simulation, the channel limit of mobile hosts is set to the same value: 2 Mbps. The distributed coordination function (DCF) of IEEE 802.11 is utilized for wireless LANs as the MAC layer protocol. It has the functionality to inform the network layer about link breakage.

In the simulation, mobile nodes move in a 1000 meter x 1000 meter network region for 50 seconds simulation time. The number of nodes are fluctuated as 10,20,30...50. Accept that every node moves autonomously with the same normal speed. All nodes have the same transmission range of 250 meters. In the simulation, the minimal
speed is 5 m/s and maximal speed is 10 m/s. The simulated traffic is Constant Bit Rate (CBR). The no. of attackers are fluctuated as 1 to 10.

The simulation settings and parameters are summarized in table 7.4

<table>
<thead>
<tr>
<th>No. of Nodes</th>
<th>10,20,30,…50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area Size</td>
<td>1000 X 1000</td>
</tr>
<tr>
<td>Mac</td>
<td>802.11</td>
</tr>
<tr>
<td>Radio Range</td>
<td>250m</td>
</tr>
<tr>
<td>Simulation Time</td>
<td>50 sec</td>
</tr>
<tr>
<td>Traffic Source</td>
<td>CBR</td>
</tr>
<tr>
<td>Packet Size</td>
<td>512</td>
</tr>
<tr>
<td>Speed</td>
<td>5m/s to 10m/s</td>
</tr>
<tr>
<td>Misbehaving Nodes</td>
<td>1 to 10</td>
</tr>
</tbody>
</table>

Table 7.1  Simulation Settings for PCTEKM

The following are the assumptions used for the proposed framework,

- A malicious node can compromise the key, create packet drop attack, routing overflow attack etc.
- A trusted node will be having a trust value greater than 0.5.
- A threshold trust is fixed as 0.5.
- Malicious node will be having a trust value less than 0.5
- Trust value 1 refers to full trust and 0 refers to complete distrust.
- The designated cluster head will be the coordinator node.
- The coordinator node is in charge of CRL update.
- Protocol used is AWCBRP.
- Nodes are moving with different velocities, ranging 5m/s to 10m/s.

7.5.2 Performance Metrics

We compare the proposed Predictive Clustering Technique for Effective Key Management in Mobile Ad Hoc Networks (PCTEKM) with Distributed Hierarchical Key Management Scheme for Mobile Ad hoc Networks (DHKM) proposed in chapter 6.

We evaluate mainly the performance according to the following metrics [91]:

- **Average end-to-end Delay**: the normal time taken by the data packets from sources to destinations, including buffer delays during a route discovery, lining delays at interface lines, retransmission delays at MAC layer and propagation time.
- **Packet Delivery Ratio**: or packet throughput, the fraction of the data packets conveyed to destination nodes to those sent by source nodes.
- **Packet Drop**: It is the number of packets dropped during the transmission.
- **MisDetection Ratio**: the proportion of the number of nodes whose behavior (malicious or considerate) is not recognized effectively to the real number of such nodes in the network.
- **Routing packet overhead**: the number of control packets (including route request/reply/update) for establishing connection over a period of time.
- **Resilience against Node Capture**: the fraction of communications compromised to the total number of communications by a capture of x-nodes.
7.5.3 Results

Varying Number of Attackers

The number of attackers is increased from 1 to 10 and the performances of the techniques are measured in terms of Delivery Ratio, Misdetection and Resilience. Since the prediction methodology for identifying the node movement, the performance of the proposed method will improve and the security will be high. The trust calculation and the isolation of malicious nodes are also incorporated.

Figure 7.2 shows the average Packet delivery Ratio of the schemes, when the attackers are increased from 1 to 5. We can see that the delivery ratio decreased linearly as the attacker increases. But, the delivery ratio of our proposed PCTEKM is greater than DHKM. Here also the trusted mechanism for the ID exchange and revocation is used.

![Fig 7.2 Packet Delivery Ratio](image-url)
Fig 7.3 Misdetection Ratio

Fig 7.4 Resilience against Node Capture
The ratio of the number of nodes whose behavior is not identified correctly to the actual number of such nodes in the network is shown in figure 7.3. Our proposed method is capable to detect more malicious nodes while comparing with DHKM.

The result of fraction of compromised communications is shown in figure 7.4. Because of the trust prediction mechanism, the number of compromised communications is less in PCTEKM. Hence the proposed PCTEKM is more resilient than DHKM.

**Varying Number of Nodes**

The CBR data packets and control packets dropped due to the attackers, presented in figures 7.5. As the number of attacker increases, more data packets are dropped. But PCTEKM has less packet drops when compared to DHKM. The cluster head updates are done depends upon the predicted mobility value, there by improves the reliability of clustering algorithm.
Figure 7.6 Average end-to-end Delay

Figure 7.6 depicts the delay involved in the communication by each pair of source and destinations. The number of nodes is varied from 10 to 50, and corresponding delay for the PCTEK and DHKM are measured. The proposed method outperforms the DHKM in case of delay. Because of the mobility prediction, the cluster heads are updated proactively and the delay will be less compared to existing methodology. Because of the updates the network resilience also improves.

Figure 7.7 shows the Routing packet overhead of the schemes, when the nodes are increased from 10 to 50. We can see that the overhead of our proposed PCTEK is greater than the DHKM since the proposed method contains the mobility prediction based on the past performance of the other nodes.
7.6 Conclusion

In this chapter, our proposed framework (distributed hierarchical key management scheme) has been enhanced by a predictive clustering technique. A prediction based clustering technique is used to reduce the delay and packet drop due to node mobility.

Here we are predicting the movement of a node from one cluster to another based on the route expiration time and link stability. So each cluster head is going to predict the movement of its members from one cluster to another. Based on the predicted value, the public/private key pair update is happening in the cluster heads. The cluster heads are updated proactively, based on the combined mobility metric. The overall delay of the proposed system is reduced due to the prediction based clustering technique that we are
using. The overhead will be high because of the prediction mechanism, we have incorporated.

The clustering technique used to select a CH, is based on weighted clustering algorithm. The CH is stored with public keys of all its member nodes. The communication of nodes between two different clusters happens through their CH. This method also discusses about the effects of node mobility between clusters. By Simulation results, it is shown that the proposed scheme achieves better delivery ratio and resilience with reduced delay and packet drop. The detection ratio of the proposed method is better than the existing one, as we have incorporated the method to predict the node movement.