CHAPTER 4

CONDITIONAL PROBABILISTIC RELIABILITY COEFFICIENT-BASED SELFISH NODE MITIGATION MECHANISMS

In MANET, reputation is considered as the vital entity for enforcing cooperation among the mobile nodes. But reputation tumbles down drastically under the influence of selfish nodes since it drop maximum number of packets and depletes significant amount of energy. Besides this, selfish nodes dramatically decrease the PDR and throughput of the network which in turn increases the communication overhead.

From the recent past, a number of selfish node mitigation mechanisms are also proposed based on reputation factor that is calculated using conditional probabilistic reliability coefficient. Further, conditional probabilistic detection approaches are considered to be superior to history-based detection mechanisms. Since they facilitate the detection of selfishness by monitoring the present behaviour of mobile nodes based on the assumption that they are reliable in the past. But most of the conditional probabilistic mitigation approaches available in literature address selfishness by calculating conditional probability using naive probability, apriori probability, Bayes probability or Dempster Shafer theory of evidences. It is evident that they have not utilized any advanced conditional probabilistic techniques like Erlang or Laplace Stleltjes transform for computing reputation of mobile nodes. Moreover, Erlang and Laplace Stleltjes transform-based reliability coefficient are identified as the predominant conditional probability computation techniques that posses the capability of integrating two independent events of nodes’ behaviour [86, 90].

This chapter presents two conditional probabilistic mitigation mechanisms, viz., i) Erlang Coefficient-Based Conditional Probabilistic Mitigation (ECBCPM) and ii) Laplace Stleltjes Transform-Based Conditional Probabilistic Mechanism (LSTBCPM). ECBCPM calculates the Conditional Probabilistic Erlang Coefficient (CPEC) for consolidating the impact of selfish nodes that affects the sur-
vivability of mobile nodes and the resilience of the entire network. LSTBCPM isolates selfish nodes by calculating Conditional Probabilistic Survivability Coefficient (CPSC) that identifies the co-operation level of mobile nodes for checking their consistency in packet forwarding.

The subsequent sections of this chapter focus on the significance of ECBCPM and its impact on rapid mitigation of selfish nodes with supporting algorithms and illustrations.

### 4.1 Erlang Coefficient-Based Conditional Probabilistic Mitigation Mechanism (ECBCPM)

ECBCPM enables rapid mitigation of selfish nodes by computing a conditional probabilistic reputation factor called CPEC. This reputation factor aids in estimating the degree of negative impact induced by the selfish nodes towards the resilience of the network. The objective of mitigating selfish nodes is achieved by considering three key points. First, the probability of packet delivery rate of each mobile node is quantified for estimating its co-operation factor. Second, the selfish-impact factor is estimated for determining the degree of influence induced by the increasing number of selfish nodes in the network. Finally, the resilience of the routing path of the network is also measured using Erlang coefficient that combines the failure rate of co-operative and selfish nodes into an integral value.

ECBCPM is a distributed conditional probabilistic mitigation scheme for mitigating selfish nodes in which the reputation value is computed in each active mobile node rather than a centralized node.

Thus ECBCPM approach mitigates selfish nodes from the routing path by utilizing the following four steps.

a) Detection of Selfish node based on Co-operative Factor \( C_F \)

b) Estimation of Selfishness-Impact Factor \( \lambda \)


d) Isolation of Selfish nodes based on CPEC.
a) Detection of Selfish node based on Co-operative Factor \( (C_F) \)

ECBCPM computes Co-operative Factor \( (C_F) \) based on the second hand information gathered from the neighbouring nodes of each mobile node. Similar to the previous approaches, \( X_{PDIN(1)} \), \( X_{PDIN(2)} \), \ldots, \( X_{PDIN(k)} \) and \( X_{PDOUT(1)} \), \( X_{PDOUT(2)} \), \ldots, \( X_{PDOUT(k)} \) denote the number of packets received and forwarded by the mobile nodes as observed by their neighbours in ‘k’ sessions respectively. Then the probability of packet delivery \( (PPD_s) \) attributed by a mobile node in each session ‘s’ is given by

\[
PPD_s = \frac{X_{PDOUT(s)}}{X_{PDIN(s)}} \quad (4.1)
\]

where \( 1 \leq s \leq k \)

Further, the Mean Probability of Packet Delivery \( (MPPD) \) which quantifies the average number of packet delivered by each participating mobile node for the entire ‘k’ sessions is represented by

\[
MPPD(k) = \frac{\sum_{s=1}^{k} PPD_s}{k} \quad (4.2)
\]

The Normalized Reputation Coefficient ‘NRC’ is calculated for highlighting the mobile nodes’ behaviour by converging the packet dropping rate exhibited by them into a specific range of values that lie between the upper and lower bounds of misbehaviour. Hence, ‘NRC’ for each mobile node ‘i’ is calculated based on \( MPPD(k) \) as

\[
NRC = 2^{MPPD(k) - NL} - NU/(NU - NL) \quad (4.3)
\]

where

\( NL \) - Lower bound value of normalization (0)
\( NU \) - Upper bound value of normalization (1)

Thus \( C_F \) of a mobile node is calculated as

\[
C_F = e^{-NRC} \quad (4.4)
\]

This \( C_F \) portrays the degree of co-operation rendered by each mobile node towards the act of forwarding packets to their neighbours. Hence, ECBCPM identifies a mobile node as selfish when ‘\( C_F \)’ of a mobile node is found to be less than 0.50 as proposed in [87], else the mobile node is designated as co-operative.

Algorithm 4.1 highlights the steps involved in estimating \( (C_F) \) of each mobile
node present in the routing path.

Algorithm 4.1: Co-operative Factor \((C_F)\) based Selfish Node Detection

1. Let the number of nodes in the network be \(N\).
2. \(N_R\) - Set of nodes of the routing path, in which two specific nodes are referred as \(N_S\) (source node) and \(N_D\) (destination node) respectively. Initially, \(N_R \leftarrow \phi\) (empty).
3. Let this algorithm step (4 - 27) be executed for an intermediate node say \(u\), of the routing path \((N_R)\) that transmits packets through \(k\) number of sessions.
4. For every node \(u \in N_R, u = 1, 2, ..., M\) do
/* Let \(M\) be the number of nodes in the routing path */
5. For every communication session \(v \in t_{session}, v = 1, 2, ..., k\) do
/* Computation of probability of packet delivery of a mobile node */
6. If \((X_{PDIN}(u,v) > 0)\) and \((X_{PDOUT}(u,v) > 0))\) then
7. \(PPD(u,v) = X_{PDOUT}(u,v)/X_{PDIN}(u,v)\)
8. End if
/* Computation of mean probability of packet delivery of a mobile node */
9. Initialize the value of \(SUM(PPD)(u,v) = 0\)
10. If \((PPD(u,v) > 0)\) then
11. \(SUM(PPD)(u,v) = SUM(PPD)(u,v) + PPD(u,v)\)
12. \(MPPD(u) = SUM(PPD)(u,v)/k\)
13. End if
14. End for.
/* Normalized Reputation Coefficient */
15. Let \(X_{NL} = 0\) and \(X_{NU} = 1\)
16. If \((MPPD(u) > 0)\) then
17. \(NRF(u) = 2^{MPPD(u) - NL - NU}/(NU - NL)\)
18. End if
/* Computation of Co-operative Factor*/
19. If \((NRF(u) > 0)\) then
20. \(C_F(u) = e^{-NRF(u)}\)
21. End if
/* Detection of selfish nodes based on Co-operative Factor */
22. If \((C_F(u) < 0.5)\) then
23. node \(u\) is selfish
24. \(u(s) \leftarrow 1\)
25. Else
node u is co-operative.

27. \( u(s) \leftarrow 0 \)

/* Count the number of identified selfish nodes and co-operative nodes */

28. Let number of selfish and co-operative nodes be \( N(S) \) and \( N(C) \)

29. Initialize \( N(S) = 0 \) and \( N(C) = 0 \)

30. If \( u(s) == 1 \) then

31. \( N(S) = N(S) + 1 \)

32. Else

33. \( N(C) = N(C) + 1 \)

34. End if

35. End for.

Figure 4.1 illustrates an ad hoc environment that contains a routing path represented by \( S \rightarrow A \rightarrow B \rightarrow F \rightarrow E \rightarrow G \rightarrow D \), where \( S \) and \( D \) are designated as source and destination nodes of the routing path. This ECBCPM scheme estimates the reputation level of mobile node based on the value of \( C_F \). In this scenario, the \( C_F \) value for the nodes \( B \) and \( F \) is found to be lower than the threshold value of 0.5. Hence, these nodes \( B \) and \( F \) are identified as selfish.

![Diagram of routing path with nodes identified as selfish or co-operative](image)

**b) Estimation of Selfishness-Impact Factor \((\lambda)\)**

The Selfishness-Impact factor \((\lambda)\) depends on the number of co-operative and selfish nodes of the network. Within the network lifetime \(x\), if a mobile mode of the routing path is designated as selfish with \( C_F \). Then it is said to be co-operative with \( 1 - C_F \) as represented by

\[
c_x(v = 0) = 1 - C_F \tag{4.5}
\]
\[ s_x(v = 1) = C_F \tag{4.6} \]

where ‘\(v\)’ is the random variable utilized for differentiating selfish nodes from co-operative nodes.

Based on the aforementioned probability of co-operation, the number of co-operative and selfish nodes in the network are identified as ‘\(c\)’ and ‘\(s\)’ respectively. Thus the Selfishness-Impact factor (\(\lambda\)) is calculated based on the packet delivery rate with its impact on the co-operative factor as

\[ \lambda = (n - (c \cdot C_F))/c \tag{4.7} \]

where ‘\(n\)’ denotes the total number of mobile nodes of the network.

This Selfishness-Impact factor ‘\(\lambda\)’ defines the degree of non co-operation attributed by each and every mobile node of the network.

Algorithm 4.2 depicts the steps involved in estimating the Selfishness-Impact factor (\(\lambda\)) is calculated based on the total number of selfish and co-operative nodes present in the environment.

---

**Algorithm 4.2: Estimation of Selfishness-Impact Factor (\(\lambda\))**

1. **Let** the number of nodes in the network be \(N\).
2. \(N_R\) - Set of nodes of the routing path, in which two specific nodes are referred as \(N_S\) (source node) and \(N_D\) (destination node) respectively. Initially, \(N_R \leftarrow \phi\) (empty)
3. **Let** this algorithm step (4 - 9) be executed for a intermediate node say, \(u\), of the routing path (\(N_R\)), that transmits packets through \('k'\) number of sessions.
4. **For** every node \(u \in N_R, u = 1, 2, ..., M\) do
   /* Let M be the number of nodes in the routing path */
5. **For** every communication session \(v \in t_{\text{session}}, v = 1, 2, ..., k\) do
6. **If** \((C_F(u) > 0)\) then
7. \(\lambda(u) = (N - (N(C) \times C_F))/C\)
8. **End if**
9. **End for**.
Figure 4.2(a) represents an ad hoc environment that contains a routing path considered as $S \rightarrow A \rightarrow B \rightarrow F \rightarrow E \rightarrow G \rightarrow D$, where $S$ and $D$ are designated as source and destination nodes of the routing path with minimum number of selfish nodes.

Here, node $B$ is identified as selfish based on $C_F$. Hence, the number of selfish nodes ($s$) is found to be 1 and the remaining nodes of the network is considered as co-operative ($c$). This ECBCPM approach calculates the Selfishness-Impact factor using (4.7) and infers the value of $\lambda = 0.71$.

Figure 4.2(b) also depicts an ad hoc environment with maximum number of selfish nodes. In this scenario, nodes $B$, $F$ and $E$ are identified as selfish based on the value of $C_F$. Hence, the number of selfish nodes ($s$) are found to be 3. Therefore, the Selfish-Impact factor of the entire routing path is estimated as $\lambda = 1.35$.

c) Computation of Conditional Probabilistic Erlang Coefficient (CPEC)

ECBCPM calculates the value of CPEC using the instantaneous failure rate of co-operative nodes and the instantaneous failure rate of selfish nodes present in the routing path within the network lifetime ‘$x$’. Hence, the instantaneous failure rate of co-operative nodes with Co-operative factor $(1 - C_F)$ is

$$IF_c = \lambda e^{-\lambda t}$$ (4.8)
Similarly, the instantaneous failure rate of selfish nodes is Erlang distributed and it is given as

\[ IF_s = \lambda^2 te^{-\lambda t} \] (4.9)

Since Erlang distribution is a special kind of conditional probabilistic distribution that highly depends on the sum of two independent exponential random variables which represents the instantaneous failure rate of co-operative and selfish nodes. At the same time, the computation of failure rate of selfish nodes becomes vital because it further decreases the reputation level of the mobile nodes participating in the routing path.

Therefore, the failure rate of the entire routing path is calculated using

\[ IF_{rp} = \lambda(1 - C_F)e^{-\lambda t} + \lambda^2 C_F te^{-\lambda t} \] (4.10)

The failure rate of the entire network completely relies on the instantaneous failure rate of co-operative nodes and instantaneous failure rate of selfish node.

Thus CPEC calculated through \( IF_c \) and \( IF_s \) for quantifying the impact of selfish nodes towards the resilience of the network is given by

\[ CPEC = (1 + C_F \lambda t) e^{-\lambda t} \] (4.11)

Algorithm 4.3 portrays the steps for estimating failure rate of the entire network based on CPEC computed by considering the instantaneous failure rate of selfish nodes and co-operative nodes.

Algorithm 4.3: Computation of Conditional Probabilistic Erlang Co-Efficient (CPEC)

1. Let the number of nodes in the network be \( N \).
2. \( N_R \) - Set of nodes of the routing path, in which two specific nodes are referred as \( N_S \) (source node) and \( N_D \) (destination node) respectively. Initially, \( N_R \leftarrow \phi \) (empty)
3. Let this algorithm step (4 - 19) be executed for a intermediate node say, \( u \), of the routing path \( (N_R) \), that transmits packets through \( k \) number of sessions.
4. For every node \( u \in N_R, u = 1, 2, ..., M \) do
   /* Let M be the number of nodes in the routing path */
5. Let ‘t’ represents each time instance of communication.
   /* Computation of instantaneous failure rate of co-operative nodes */
6. If \((\lambda(u) > 0)\) then
7. \(IF_c = \lambda(u)e^{-\lambda(u)t}\)
8. End if

/* Computation of instantaneous failure rate of selfish nodes */
9. If \((\lambda(u) > 0)\) then
10. \(IF_s = (\lambda(u))^2te^{-\lambda(u)t}\)
11. End if

/* Computation of failure rate for the entire routing path */
12. If \((C_F(u) > 0)\) and \((\lambda(u) > 0)\) then
13. \(IF_{rp} = \lambda(u)(1 - C_F(u))e^{-\lambda(u)t} + (\lambda(u))^2C_Fte^{-\lambda(u)t}\)
14. End if
/* Computation Conditional Probabilistic Erlang Coefficient (CPEC) */
15. If \(((C_F(u) > 0)\) and \((\lambda(u) > 0)\)) then
16. \(CPEC(u) = (1 + C_F(u)\lambda(u)t)e^{-\lambda(u)t}\)
17. End if
18. End for
19. End for.

Figure 4.3(a) represents the computation of CPEC in the mobile nodes of the routing path, S→A→B→F→E→G→D that contains minimum number of selfish nodes. Here, node B is identified as selfish based on the value of \(C_F\). Then, ECBCPM estimates the impact of selfish nodes towards the resilience of the network through \(IF_c\) and \(IF_s\) as \(CPEC = 0.74\).

![CPEC = 0.74](image)

Figure 4.3: ECBCPM - Estimation of CPEC under the influence of (a) minimum number of selfish nodes (b) maximum number of selfish nodes

In contrast, Figure 4.3(b) illustrates the computation CPEC in the mobile nodes of the routing path, \(S \rightarrow A \rightarrow B \rightarrow F \rightarrow E \rightarrow G \rightarrow D\) that contains maximum number of selfish nodes. In this scenario, nodes B, F and D are identified as selfish through \(C_F\). This ECBCPM approach estimates the impact of selfish nodes present in the routing path based on \(\lambda\), \(IF_c\) and \(IF_s\) as \(CPEC = 0.32\).
d) Isolation of Selfish nodes based on CPEC

ECBCPM incorporates the decision of isolating selfish nodes based on computed CPEC value. This CPEC highlights the influence of selfish nodes towards the resilience of the network. Thus, ECBCPM isolates the selfish nodes of the network when the value of CPEC is found to be less than the resilience threshold.

Algorithm 4.4 focuses on the process of isolating the identified selfish nodes based on the value of CPEC.

Algorithm 4.4: Isolation of Selfish nodes based on CPEC

1. Let the number of nodes in the network be $N$.
2. $N_R$ - Set of nodes of the routing path, in which two specific nodes are referred as $N_S$ (source node) and $N_D$ (destination node) respectively.
3. Let this algorithm step (4 - 11) be executed for an intermediate node say, $u$, of the routing path ($N_R$) that transmits packets for ‘$k$’ number of sessions.
4. For every node $u \in N_R, \ u = 1, 2, ..., M$ do
   /* Let $M$ be the number of nodes in the routing path */
5. Let resilience threshold $R_{Thres} = 0.60$
6. If $(CPEC(u) < R_{Thres})$ then
7. node $u$ is selfish
8. Else
9. node $u$ is co-operative
10. End if
11. End for.

Figure 4.4 illustrates the selfish node isolation process of ECBCPM. This ECBCPM computes the CPEC value using $\lambda$, $IF_c$ and $IF_s$ values. Since the value of CPEC is found to be less than the resilience threshold of the network (0.60), the identified selfish nodes B, F and E are isolated from the routing path.
4.1.1 Experimental Analysis and Results Discussions

In this section, the performance and characteristics of the proposed ECBCPM is thoroughly analyzed using ns-2 simulation (version 2.26). The performance of ECBCPM is compared with the existing algorithms like EHRFBDM, SHRFBDM, ENRFBDM and PCMA that are also simulated with the same characteristics and network related parameters of ECBCPM. The comparative analysis of ECBCPM is also investigated based on the performance metrics such as PDR, throughput, control overhead, total overhead and energy consumptions by varying the number of mobile nodes, number of selfish nodes, maximum and minimum threshold detection point set for selfishness detection.

Simulation Environment

The simulated network consists of 100 nodes randomly distributed in a rectangular area of 1000 x 1000 meters. Each of the simulation runs for 300 seconds and the collected data is averaged for each point. IEEE 802.11 is used as the underlying MAC layer communication model with the data rate and radio range set to 2 Mbps and 250 m respectively. This environment also uses random waypoint as the node mobility model with the minimum and maximum speed of 10 meters per second and 30 meters per second respectively. The number of source and destination pairs are varied between 10 and 40 with the packet size of 512 bytes.

Results and Discussions

Initially, the simulation experiments are conducted for identifying the optimal threshold detection point of ECBCPM by varying the range set for detection. The results strongly emphasize that ECBCPM detects maximum numbers of selfish nodes at 0.30 greater than the baseline mitigation techniques like EHRFBDM,
SHRFBDM and PCMA considered for study as shown in Figure 4.5. Hence, 0.30 is fixed as the optimal threshold point of detection.

From the results, it is evident that ECBCPM has the potential to identify significant number of selfish nodes within the detection range of 0.25 to 0.35. Hence, 0.25 and 0.35 are identified as the maximum and minimum threshold detection points of ECBCPM respectively.

The performance of ECBCPM is also evaluated by varying the a) number of mobile nodes, b) number of selfish nodes, c) number of mobile nodes with maximum threshold detection point of 0.25 and d) number of mobile nodes with minimum threshold detection point of 0.35.

a) Experiment 1 - Comparative Analysis of ECBCPM based on varying number of mobile nodes

In this experiment, the performance of ECBCPM is initially evaluated by comparing it with the contributed mitigation approaches like EHRFBDM, SHRFBDM, ENRFBDM and traditional PCMA. Further, ECBCPM is also compared with the baseline conditional probabilistic mitigation approaches like PPBST [87] and PBNST [88]. This comparative analysis is performed by considering 20% of mobile nodes as selfish with optimal threshold detection point of 0.30.

Figures 4.6 and 4.7 demonstrate the results of PDR and throughput observed by varying the number of mobile nodes involved in data transmission. It is identified that PDR of all the implemented mitigation mechanisms decrease systematically with increase in the number of mobile nodes. This decrease in PDR and
throughput are mainly due to the lack of a reliable process that polices enormous amount of data introduced into the network.

However, ECBCPM exhibits an improved PDR and throughput than EHRFBDM, SHRFBDM, ENRFBDM and PCMA as it utilizes an Erlang distribution based reliability coefficient for isolating selfish nodes. This also quantifies the cooperation factor of individual mobile nodes and the survivability rate of the entire routing path. Hence, ECBCPM shows an improvement in PDR by 5%-7% over EHRFBDM, 9%-11% over SHRFBDM, 14%-16% over ENRFBDM and 19%-22% over PCMA. In addition, ECBCPM on an average shows a phenomenal improvement of 16.5% in PDR.
Similarly, ECBCPM exhibits an increase in throughput by 4%-6% over EHRFBDM, 8%-10% over SHRFBDM, 12%-14% over ENRFBDM and 15%-17% over PCMA. It is also clear that ECBCPM on an average increases the throughput of the network by 13%.

Moreover, increase in the number of mobile nodes of an ad hoc network proportionally increases the number of transmissions that increases the control overhead and total overhead as presented in Figures 4.8 and 4.9. But ECBCPM minimizes the control overhead by 6%-8% over EHRFBDM, 14%-16% over SHRFBDM, 18%-20% over ENRFBDM and 22%-25% over PCMA.

![Figure 4.8: ECBCPM - Experiment 1 - Control Overhead](image)

![Figure 4.9: ECBCPM - Experiment 1 - Total Overhead](image)

Similarly, ECBCPM also reduces the total overhead by 9%-12% over EHRFBDM, 14%-16% over SHRFBDM, 18%-20% over ENRFBDM and 21%-23% over PCMA.
PCMA. The results also confirm that ECBCPM on an average reduces the total overhead and control overhead by 22% and 18% respectively.

In addition, ECBCPM outperforms PPBST and PBNST in terms of PDR and throughput on an average by 14% and 12% respectively. It also decreases the control overhead and total overhead by 16% and 15% respectively.

b) Experiment 2 - Comparative Analysis of ECBCPM based on varying the number of selfish nodes

In this experimental analysis, the performance of ECBCPM is evaluated by varying the number of selfish nodes from 10 to 50 with 0.30 as optimal threshold point of detection. Figures 4.10-4.13 show the plots of PDR, control overhead, delay and detection rate of ECBCPM, EHRFBDM, SHRFBDM, ENRFBDM and PCMA considered for study.

Figure 4.10 presents the PDR of ECBCPM studied under the influence of varying number of selfish nodes. PDR decreases with increase in the number of selfish nodes as they force the nodes to drain considerable amount of energy. This energy drain affects the stability of mobile nodes and influences the resilience of the network. However, ECBCPM handles this issue by employing Erlang-based reliability coefficient that accurately quantifies the influence of selfish nodes for improving the lifetime of network. Hence, ECBCPM shows an improvement in PDR by 9%-15% over EHRFBDM, 11%-17.2% over SHRFBDM, 14%-26% over ENRFBDM and 18.2%-28% over PCMA.

![Figure 4.10: ECBCPM - Experiment 2 - Packet Delivery Ratio](image)

Figures 4.11 and 4.12 show the results of control overhead and delay identified
by varying the number of selfish nodes in the network. The control overhead and delay increases with increased number of selfish nodes as they induce high energy drain that disturbs the stability of the node. Thus ECBCPM demonstrates that control overhead is reduced by 14%-17% over EHRFBDM, 19%-21% over SHRFBDM, 23%-26% over ENRFBDM and 25%-28% over PCMA. Further, ECBCPM portrays that the delay is reduced by 15%-18% over EHRFBDM, 19%-22% over SHRFBDM, 24%-25.5% over ENRFBDM and 26%-28% over PCMA. The results also confirm that ECBCPM on an average reduces the control overhead and delay by 21.2% and 23.4% respectively.

![Figure 4.11: ECBCPM - Experiment 2 - Control Overhead](image1)

![Figure 4.12: ECBCPM - Experiment 2 - Delay](image2)

Figure 4.13 represents the results for detection rate facilitated by the selfish node mitigation mechanisms considered for comparative analysis. ECBCPM exhibits a phenomenal improvement in detection rate even when the number of
selfish nodes in the network increases. Thus it is transparent that ECBCPM demonstrates an increased detection rate by 14%-17% over EHRFBDM, 19%-21% over SHRFBDM, 23%-26% over ENRFBDM and 5%-28% over PCMA.

In addition, ECBCPM outperforms PPBST and PBNST in terms of PDR on an average by 16%. It also decreases the control overhead and total overhead by 15% and 12% respectively.

c) Experiment 3 - Comparative Analysis of ECBCPM with maximum detection threshold point

In experiment 3, the performance of ECBCPM over EHRFBDM, SHRFBDM, ENRFBDM and PCMA is analyzed with maximum detection threshold point of 0.25 by varying number of mobile nodes. It is inferred that, ECBCPM exhibits an improved performance in terms of throughput. The results indicate that, this significance of improvement provisioned by ECBCPM is slightly higher than the performance identified at its minimum detection threshold point. From Figure 4.14, it is obvious that ECBCPM improves the throughput by 7%-11% over EHRFBDM, 15%-19% over SHRFBDM, 20%-22% over ENRFBDM and 24%-26% over PCMA. It is observed that, ECBCPM on an average improves the throughput by 21.2%.
Further, the results from Figures 4.15 and 4.16 infer that ECBCPM reduces the total overhead and energy consumptions at maximum threshold point of detection. Since at maximum threshold point, the number of retransmissions and energy consumptions is highly reduced. Thus the performance of ECBCPM at this point is considerably higher than its performance at minimum threshold point. Figure 4.15 portrays that ECBCPM reduces the total overhead by 17%-22% over EHRFBDM, 19%-26% over EHRFBDM, 19%-26% over SHRFBDM and 24%-32% over PCMA.

Similarly, Figure 4.16 proves that ECBCPM reduces the energy consumptions 13%-19% over EHRFBDM, 15%-19% over SHRFBDM, 19%-23% over ENRFBDM and 26%-31.2% over PCMA. Hence, ECBCPM is highly effective in reducing the total overhead and energy consumptions at an average rate of 18.4% and 25.5%
respectively.

Finally, it is also inferred that ECBCPM outperforms PPBST and PBNST in terms of throughput on an average by 16.2%. It also decreases the control overhead and energy consumptions by 18% and 23.6% respectively.

d) Experiment 4 - Comparative Analysis of ECBCPM with minimum detection threshold point

Finally, the performance of ECBCPM over EHRFBDM, SHRFBDM, ENRF-BDM and PCMA is analyzed with minimum detection threshold point of 0.35 through experiment 4. It is inferred that, ECBCPM exhibits an improved performance in terms of throughput which is slightly lower than the performance exhibited at its maximum point of selfish node detection. From Figure 4.17, it is inferred that ECBCPM improves the throughput by 4%-6% over EHRFBDM, 9%-11% over SHRFBDM, 13%-15% over ENRFBDM and 17%-20% over PCMA. Thus ECBCPM on an average improves the throughput by 16.4%.
Further, the results from Figures 4.18 and 4.19 indicates that ECBCPM reduces total overhead and energy consumptions at the minimum threshold point of selfishness detection. The number of retransmissions and energy consumptions are considerably reduced but not competent enough when compared to the maximum threshold detection point. Thus, ECBCPM reduces the total overhead by 12%-14% over EHRFBDM, 16%-18% over EHRFBDM, 19%-20% over SHRFBDM and 22%-25% over PCMA. Similarly, Figure 4.19 demonstrates that ECBCPM reduces the energy consumption rate by 10%-13% over EHRFBDM, 15%-17% over SHRFBDM, 19%-21% over ENRFBDM and 23%-25% over PCMA. Hence, ECBCPM is highly effective in reducing the total overhead and energy consumption on an average rate of 15.2% and 22.4% respectively.
In addition, ECBCPM outperforms PPBST and PBNST in terms of throughput on an average by 13.4%. It also decreases the control overhead and energy consumption by 14% and 19.6% respectively. Thus, ECBCPM is found to be capable of mitigating selfish mobile nodes at a rapid rate of 30% superior to the existing conditional probabilistic mitigation approaches.

Similar to Erlang reliability coefficient, Laplace Steltjes Transform-Based Survivability coefficient can be also used for mitigating selfish nodes.

4.2 Laplace Steltjes Transform-Based Conditional Probabilistic Mechanism (LSTBCPM)

LSTBCPM detects selfish nodes based on the computation of conditional probability quantification factor namely Conditional Probabilistic Survivability Coefficient. This CPSC is mainly calculated for identifying the co-operation level of mobile nodes in order to check their consistency level of packet forwarding behaviour observed in the past and present event of monitoring. CPSC quantifies the magnitude of conditional probability using two independent exponentially distributed parameters that relate to the survivability rate of co-operative and selfish nodes. Further, Laplace Steltjes Transform-Based Survivability coefficient is highly suitable for estimating conditional probability because it numerically derives the relation between failure time distribution and temporal events causing the failure. It is further used in a number of reliability models for incorporating the effect of stochastic and dynamic change of mobile node’s lifetime. In addition,
it is also used for describing the stability rate function and failure rate function of events that are analyzed during the detection process. This conditional probabilistic mechanism also helps in estimating the degree of negative influence posed by the selfish nodes towards the survivability of the network.

In LSTBCPM, let ‘\(x\)’ denotes the overall network lifetime of the mobile nodes and \(X_{PDIN(1)}, X_{PDIN(2)}, \ldots, X_{PDIN(k)}\) and \(X_{PDOUT(1)}, X_{PDOUT(2)}, \ldots, X_{PDOUT(k)}\) represent the number of packets received and forwarded by a mobile node as observed by its neighbours in the entire ‘\(k\)’ sessions respectively.

Then the survivability factor \((S_f)\) estimated for each and every mobile node within the network lifetime ‘\(x\)’ based on coefficient of determination is given by

\[
S_f = \sqrt{\frac{1}{k-1} \sum_{s=1}^{k} \left( \frac{X_{PDIN(s)} - \text{Average}_{IN(k)}}{SSDEV_{IN(k)}} \right) \left( \frac{X_{PDOUT(k)} - \text{Average}_{OUT(k)}}{SSDEV_{OUT(k)}} \right)}
\]

(4.12)

where, \(\text{Average}_{IN(k)}, SSDEV_{IN(k)}, \text{Average}_{OUT(k)}\) and \(SSDEV_{OUT(k)}\) is calculated using (3.6), (3.7), (3.10) and (3.11) respectively.

Further, the survivability factor ‘\(S_f\)’ categorizes the mobile nodes of the network into co-operative or selfish and labels them by using a random variable ‘\(r\)’ as

\[
S_k(r = 0) = S_f
\]

(4.13)

\[
C_{n-k}(r = 1) = 1 - S_f
\]

(4.14)

Moreover, if the estimated ‘\(S_f\)’ of a mobile node is found to be less than the survivability detection threshold of 0.50 as specified in [87], the node is detected as selfish and the random variable is set to \(r = 0\). Otherwise, the mobile node is said to be co-operative with \(r = 1\). Further, this random variable ‘\(r\)’ is used for identifying the number of co-operative and selfish nodes of the network.

In this context, let ‘\(k\)’ and ‘\(n - k\)’ represent the number of co-operative and selfish nodes identified among ‘\(n\)’ nodes of the network. Then, the instantaneous survivability rate ‘\(\delta\)’ imposed by ‘\(k\)’ co-operative nodes of the network is given by

\[
\delta = \frac{k}{n}(S_f)
\]

(4.15)

Likewise, the instantaneous failure rate ‘\(\zeta\)’ induced by ‘\(n-k\)’ selfish nodes of
the network is quantified by

\[ \zeta = \frac{n - k}{n} (1 - S_f) \]  \hspace{1cm} (4.16)

Hence, it is evident that survivability rate of the network depends on the instantaneous rate of survivability rendered by co-operative nodes and the instantaneous rate of failure induced by the selfish nodes. This, instantaneous rate of survivability and instantaneous rate of failure is found to be independent and exponentially distributed. Hence, Laplace Steltjes transform based reliability coefficient [114] is found to be highly suitable for calculating the reputation for each mobile node participating in the network. Thus, the lifetime of the network ‘x’ can be estimated using a conditional probability value based on the integrated exponentially distributed parameter ‘\( \delta + \zeta \)’.

In addition, the aforementioned integrated exponential parameter is computed using the probability mass functions of co-operative and selfish nodes as \( \delta/(\delta + \zeta) \) and \( \zeta/(\delta + \zeta) \) respectively. Hence, the survivable probability possessed by a mobile node with its associated probability mass function during its co-operation or selfish behaviour is represented by

\[ P_S(r = 0) = \frac{\delta(S_f)}{\delta + \zeta} \]  \hspace{1cm} (4.17)

and

\[ P_C(r = 1) = \frac{\zeta(1 - S_f)}{\delta + \zeta} \]  \hspace{1cm} (4.18)

Thus, the survivable probability of co-operative nodes and selfish nodes are computed by using the product of their corresponding probability mass function and survivability factor identified for their behaviour.

Further, conditional probability estimating ‘CPSC’ depends on the survivable probability of co-operative nodes and selfish nodes. Hence, Laplace Steltjes transform is applied to both the survivable probabilities in order to identify the resultant conditional probabilistic survivability coefficient of the network as represented by

\[ L_{x(S)}(r = 0) = (\delta + \zeta)e^{-(\delta + \zeta)t} \]  \hspace{1cm} (4.19)

\[ L_{x(C)}(r = 1) = (\delta + \zeta)e^{-\delta t}e^{-(\delta + \zeta)t} \]  \hspace{1cm} (4.20)

Furthermore, the integrated conditional probabilistic survivability coefficient
‘CPSC’ computed for entire network through total theorem of probability is

\[ CPSC_i(t) = \delta e^{-(\delta+\zeta)t}(1 - S_f + \delta S_f e^{-\delta t}) \]  \hspace{1cm} (4.21)

When the value of \( CPSC_i(t) \) is less than 0.30, the process of isolating selfish nodes takes place.

Algorithm 4.4 depicts the steps involved in computing CPSC for selfish node mitigation.

---

Algorithm 4.4: Conditional Probabilistic Survivability Coefficient (CPSC) based Selfish Node Mitigation

1. Let the number of nodes in the network be \( N \)
2. \( N_R \) - Set of nodes of the routing path, in which two specific nodes are referred as \( N_S \) (source node) and \( N_D \) (destination node) respectively. Initially, \( N_R \leftarrow \emptyset \) (empty)
3. Let this algorithm step (4 - 47) be executed for a intermediate node say, \( u \), of the routing path (\( N_R \)) that transmits packets for ‘\( k \)’ number of sessions.
4. For every node \( u \in N_R, u = 1, 2, ..., M \) do
   /* Let \( M \) be the number of nodes in the routing path */
5. For every communication session \( v \in t_{session}, v = 1, 2, ..., k \) do
   /* Find the average of received and forwarded packets by each node in a session */
6. Initialize the value of \( \text{Average}_{IN}(u,v) = 0, \text{Average}_{OUT}(u,v) = 0 \)
7. \( \text{Average}_{IN}(u,v) = \text{Average}_{IN}(u,v) + \text{Sum}_{IN}(u,v)/k \)
8. \( \text{Average}_{OUT}(u,v) = \text{Average}_{OUT}(u,v) + \text{Sum}_{OUT}(u,v)/k \)
9. End for
/* deviation in the number of packets forwarded*/
10. For every communication session \( v \in t_{session}, v = 1, 2, ..., j, ..., k \) do
11. If ((\( \text{Average}_{IN}(u,v) > 0 \)) and (\( \text{Average}_{OUT}(u,v) > 0 \))) then
12. \( \text{DEV}_{IN}(u,v) = X_{PDIN}(u,v) - \text{Average}_{IN}(u,v) \)
13. \( \text{DEV}_{OUT}(u,v) = X_{PDOUT}(u,v) - \text{Average}_{OUT}(u,v) \)
14. End if
/* sum of squares of deviation */
15. Initialize the value of \( SSDEVI_{IN}(u,v) = 0, SSDEVI_{OUT}(u,v) = 0 \)
16. If ((\( \text{DEV}_{IN}(u,v) > 0 \)) and (\( \text{DEV}_{OUT}(u,v) > 0 \))) then
17. \( SSDEVI_{IN}(u,v) = SSDEVI_{IN}(u,v) + (\text{DEV}_{IN}(s))^2 \)

131
18. \( SSDEV_{OUT(u,v)} = SSDEV_{OUT(u,v)} + (DEV_{OUT(s)})^2 \)
19. **End if**
20. **End for**

/* Find the survivability factor for each node in a session */
21. \[ S_f = \sqrt{\frac{1}{k-1} \sum_{s=1}^{k} \left( \frac{X_{P_{IN}(s)} - \text{Average}_{IN(k)}}{SSDEV_{IN(k)}} \right) \left( \frac{X_{P_{OUT}(k)} - \text{Average}_{OUT(k)}}{SSDEV_{OUT(k)}} \right)} \]

/* Detection of selfish nodes based on survivability factor */
22. **If** \( S_f(u) < 0.5 \) **then**
23. node \( u \) is **selfish**
24. \( u(s) \leftarrow 0 \)
25. **Else**
26. node \( u \) is co-operative
27. \( u(s) \leftarrow 1 \)

/* Count the number of identified co-operative nodes */
28. Let number of co-operative nodes in the routing path be \( N(C) \) and Initialize \( N(C) = 0 \)
29. **If** \( (u(s) == 0) \) **then**
30. \( N(C) = N(C) + 1 \)
31. **End if**
32. **End for.**

/* Computation of instantaneous survivability rate \( (\delta) \) for \( N(C) \) number of co-operative nodes */
33. **For** every node \( u \in N_R, u = 1, 2, ..., M \) **do**
34. **For** every communication session \( v \in t_{session}, v = 1, 2, ..., k \) **do**
35. **If** \( (S_f(u) > 0) \) **then**
36. \( \delta = N(C)/N \times (S_f(u)) \)
37. **End if**
38. /* Computation of instantaneous survivability rate \( (\zeta) \) for \( N - N(C) \) number of co-operative nodes */
39. **If** \( (S_f(u) > 0) \) **then**
40. \( \zeta = N - N(C)/N \times (1 - S_f(u)) \)
41. **End if**

/* Compute associated probability mass function during its co-operation */
42. **If** \( (\delta > 0) \) **then**
43. \( L_{x_c}(r = 0) = (\delta + \zeta)e^{-(\delta+\zeta)t} \)
44. **End if**

/* Compute associated probability mass function during its selfish */
45. **If** \( (\zeta > 0) \) **then**
46. \( L_{x_c}(r = 1) = (\delta + \zeta)e^{-\delta t}e^{-(\delta+\zeta)t} \)
45. End if
/*Conditional Probabilistic Survivability Coefficient (CPSC)*/
46. If ($S_f(u) > 0$) then
47. $CPSC_u(t) = \delta e^{-\delta t}(1 - S_f + \delta S_f e^{-\delta t})$
48. End if
/* Detection of selfish nodes based on Conditional Probabilistic Survivability Coefficient */
49. If ($CPSC_u(t) < 0.3$) then
50. Call Selfish Isolation
51. Else
52. Normal Routing
53. End if
54. End for
55. End for.

4.2.1 Illustrations for LSTBCPM

In this section, the effectiveness of LSTBCPM in identifying selfishness is illustrated based on two scenarios, viz., i) ad hoc environment with minimum number of selfish nodes and ii) ad hoc environment with maximum number of selfish nodes.

Scenario 1: Ad hoc environment with minimum number of selfish nodes.

Consider the group of mobile nodes ($n = 10$) of an ad hoc network, that are classified into co-operative ($k = 7$) and selfish nodes ($n - k = 3$) based on the survivability factor and further labeled as ‘0’ or ‘1’ as shown in Figure 4.20. Since the number of co-operative nodes are comparatively high than the selfish nodes, the rate of survivability ‘$\delta$’ of the co-operative nodes and rate of failure ‘$\zeta$’ of the selfish nodes are identified as 0.42 and 0.12 respectively. Further, $CPSC$ value computed for the entire network is 0.702 ($CPSC = 0.702$) using (4.21) with the aid of survivability rate of cooperative and selfish nodes computed through (4.19) and (4.20) respectively.
Scenario 2: Ad hoc environment with maximum number of selfish nodes

In contrast to scenario 1, Consider the group of nodes \( n = 10 \) of an ad hoc network that are classified into co-operative \( k = 5 \) and selfish nodes \( n - k = 5 \) based on the survivability factor as shown in Figure 4.21. Since the number of co-operative nodes are comparatively high than the selfish nodes. The rate of survivability \( \delta \) and rate of failure \( \zeta \) for the entire network is identified as 0.18 and 0.28 respectively. The value of CPSC for the entire network is computed as 0.249 \( (CPSC = 0.249) \).
4.2.2 Simulation Experiments and Results Analysis

Similar to ECBCPM approach, the performance and characteristics of LSTBCPM is also thoroughly studied using ns-2.26 simulator. The performance of LSTBCPM is compared with the proposed mitigation techniques like EHRFBDM, SHRFBDM and classical PCMA. It is also compared with the baseline conditional probabilistic mitigation mechanisms like PPBST and PBNST. These comparative analyses of LSTBCPM is performed based on PDR, throughput, control overhead, total overhead, delay and energy consumptions by varying the number of mobile nodes and selfish nodes.

Simulation Environment

LSTBCPM is implemented in a simulated network environment that contains 100 nodes randomly distributed in a terrain area of 1000 x 1000 meters. The simulation is run for 300 seconds with data rate and radio range set to 2 Mpbs and 250 meters respectively. Further, random way-point model is considered as the node mobility model with IEEE 802.11 as the underlying MAC layer communication model for simulation. Furthermore, the number of source and destination pairs is varied from 10 to 40 connections with packet size of 512 bytes by varying the node mobility speed from 10 m/sec to 30 m/sec.

Results and Discussions

Initially, LSTBCPM is first investigated with the proposed mitigation approaches for identifying the conditional probabilistic survivability coefficient. From Figure 4.22, it is inferred that LSTBCPM exhibits an effective performance at 0.30 since maximum numbers of selfish nodes are detected by LSTBCPM at this point than EHRFBDM, SHRFBDM, ENRFBDM and PCMA considered for study. Hence, the conditional probabilistic survivability coefficient of LSTBCPM is set to 0.30 for comparative analysis.
Figure 4.22: LSTBCPM - Experiment 1 - Identification of saddle point for selfish node detection

From Figure 4.22, it is inferred that LSTBCPM is also equivalently capable of identifying maximum number of selfish nodes within the detection range of 0.25 and 0.35. Hence, maximum and minimum detection threshold point of LSTBCPM is considered as 0.25 and 0.35 respectively.

Further, the performance of LSTBCPM is evaluated using four experiments by varying the a) number of mobile nodes, b) number of selfish nodes, c) number of mobile nodes with maximum detection threshold point of 0.25 and d) number of mobile nodes with minimum detection threshold point of 0.35.

a) Experiment 1 - Comparative Analysis of LSTBCPM by varying the number of mobile nodes

In this experiment, the performance of LSTBCPM is compared with the proposed EHRFBDM, SHRFBDM and PCMA by varying the number of mobile nodes with ‘0.30’ set as conditional probabilistic survivability coefficient.

Figures 4.23 and 4.24 depicts the results of packet delivery ratio and throughput observed by varying the number of mobile nodes. The PDR and throughput of LSTBCPM and the compared mitigation approaches considerably decreases as the number of mobile nodes increases. This is mainly due to the absence of a rapid routing process that handles an additional sum of data introduced into the network. But LSTBCPM improves the packet delivery rate and throughput by incorporating Laplace Stleltjes transform based reliability coefficient that integrates the survivability rate of co-operative and selfish nodes. It isolates selfish nodes at rapid rate of 30% greater than EHRFBDM, SHRFBDM, and PCMA techniques.
Hence, LSTBCPM shows an improvement in PDR by 4%-6% over EHRFBDM, 8%-10% over SHRFBDM and 14%-16% over PCMA. Similarly, LSTBCPM improves the throughput by 4%-6% over EHRFBDM, 8%-10% over SHRFBDM and 15%-17% over PCMA. In addition, LSTBCPM on an average shows a phenomenal improvement of 12% and 10% in terms of PDR and throughput.

Further, total overhead and control overhead of the network proportionally increases with increased number of transmissions. However, LSTBCPM minimizes the control overhead by 6%-8% over EHRFBDM, 14%-16% over SHRFBDM and 22%-25% over PCMA by enforcing the isolation of selfish nodes at a rapid rate of 30% as depicted in Figures 4.25 and 4.26 respectively. Similarly, LSTBCPM also demonstrates that total overhead is reduced by 9%-12% over EHRFBDM, 14%-16% than SHRFBDM and 21%-23% over PCMA. The results confirm that
LSTBCPM on an average reduces the total overhead and control overhead by 22% and 18% respectively.

In addition, LSTBCPM outperforms PPBST and PBNST in terms of PDR and throughput on an average by 11% and 9% respectively. It also decreases the control overhead and total overhead by 12% and 10% respectively.

b) Experiment 2 - Comparative Analysis of LSTBCPM based on varying the number of selfish nodes

In experiment 2, the performance of LSTBCPM is evaluated by varying the number of selfish nodes from 10 to 50 with '0.30' set as conditional probabilistic survivability coefficient.
Figure 4.27 presents the PDR of LSTBCPM analyzed by varying the number of selfish nodes. PDR decreases with increased number of selfish nodes as it forces the mobile nodes to exhaust energy. This intentional act affects the lifetime of an individual node and the routing path. However, LSTBCPM handles this impact by utilizing Laplace Stieltjes transform based reliability coefficient for estimating the influence of selfish nodes towards the lifetime of network. Hence, LSTBCPM shows an improvement in PDR by 6%-8% over EHRFBDM, 9%-11% over SHRFBDM and by 14%-17% over PCMA.

![Figure 4.27: LSTBCPM - Experiment 2 - Packet Delivery Rate](image.png)

Figures 4.28 and 4.29 shows the control overhead and delay estimated by varying the number of selfish nodes of an ad hoc network. The control overhead and delay increases with increase in number of selfish nodes due to their induced high energy consumption rate that crumbles the stability of the wireless link. Thus, LSTBCPM demonstrates that control overhead is reduced by 10%-12% over EHRFBDM, 14%-16% over SHRFBDM and from 18%-21% over PCMA. Further, LSTBCPM portrays that the delay is reduced by 9%-12% over EHRFBDM, 14%-17% over SHRFBDM and 19%-22% over PCMA. In addition to this, the results also confirm that LSTBCPM on an average reduces the control overhead and delay by 13.2% and 16.4% respectively.
Figure 4.28: LSTBCPM - Experiment 2 - Control Overhead

Figure 4.29: LSTBCPM - Experiment 2 - Delay

Figure 4.30 presents the results for detection rate facilitated by the selfish node mitigation mechanisms considered for comparative analysis. It is inferred that, the detection ratio of LSTBCPM increases with increase in the number of selfish nodes due to its rapid detection strategy. Thus it is transparent that LSTBCPM demonstrates an increased detection rate by 9%-11% over EHRFBDM, 14%-17% over SHRFBDM and 19%-22% over PCMA.
In addition, LSTBCPM outperforms PPBST and PBNST in terms of PDR on an average by 13% and it also decreases the control overhead and delay by 8% and 6% respectively.

c) Experiment 3 - Comparative Analysis of LSTBCPM with maximum detection threshold point

In experiment 3, the performance of LSTBCPM is compared with EHRF-BDM, SHRFBDM and PCMA with maximum detection threshold point of 0.25 by varying number of mobile nodes. It is inferred that, LSTBCPM exhibits an improved performance in terms of throughput.

However, the results indicate that, this significance of improvement provisioned by LSTBCPM is slightly higher than its performance identified at the min-
imum detection threshold point. From Figure 4.31, it is obvious that LSTBCPM increases the throughput by 4%-7% over EHRFBDM, 9%-13% over SHRFBDM and 15%-18% than PCMA. In addition, it is observed that LSTBCPM on an average improves the throughput by 14.4%.

Moreover, the results from Figures 4.32 and 4.33 indicate that LSTBCPM significantly improves the network performance by decreasing the total overhead and energy consumptions at the maximum threshold point of detection. Since, the number of retransmissions and energy consumptions is greatly reduced at this point. Thus, the performance of LSTBCPM is seem to be slightly higher than its minimum threshold point of detection. Figure 4.32 portrays that LSTBCPM reduces the total overhead 10%-14% over EHRFBDM, 16%-19% over SHRFBDM and 18%-21% over PCMA. Similarly, Figure 4.33 demonstrates that LSTBCPM reduces the energy consumptions by 11%-14% over EHRFBDM, 17%-20% over SHRFBDM and 22%-25% over PCMA. Hence, it is obvious that LSTBCPM is highly effective in reducing the total overhead and energy consumptions at an average rate of 15.2% and 21.4% respectively.

![Figure 4.32: LSTBCPM - Experiment 3 - Total Overhead](image)
In addition, LSTBCPM outperforms PPBST and PBNST in terms of throughput at an average rate of 14.2%. It also decreases the total overhead and energy consumptions by 15% and 18.4% respectively.

d) Experiment 4 - Comparative Analysis of LSTBCPM with minimum detection threshold point

Finally, the performance of LSTBCPM over EHRFBDM, SHRFBDM and PCMA is also analyzed with minimum detection threshold point of 0.35 through experiment-4. It is inferred that, LSTBCPM exhibits an improved performance in terms of throughput which is slightly lower than the performance exhibited at its maximum point of selfish node detection.
5%-7% over EHRFBDM, 10%-12% over SHRFBDM and 16%-19% over PCMA. In addition, it is observed that LSTBCPM on an average improves the throughput by 15.2%.

Further, Figures 4.35 and 4.36 indicate that LSTBCPM significantly improves the network performance by decreasing the total overhead and energy consumptions at the minimum threshold point of selfishness detection. Since, the number of retransmissions and energy consumptions is considerably reduced at this point.

Thus, the Figure 4.35 portrays that LSTBCPM reduces the total overhead by 10%-14% over EHRFBDM, 16%-18% over SHRFBDM and 20%-22% over PCMA. Similarly, Figure 4.36 demonstrates that LSTBCPM reduces the energy consumption rate by 10%-13% over EHRFBDM, 15%-17% over SHRFBDM and 23%-25%...
over PCMA. Hence, it is obvious that LSTBCPM is highly effective in reducing the total overhead and energy consumptions on an average rate of 12% and 17.4% respectively.

In addition, LSTBCPM outperforms PPBST and PBNST in terms of throughput on an average by 11.6%. It also decreases the control overhead and energy consumptions by 11% and 16.6% respectively.

Besides, LSTBCPM is also compared with ECBCPM and the results infer that ECBCPM is superior to LSTBCPM in mitigating selfish nodes. Since, ECBCPM facilitates a detection rate of 5% greater than LSTBCPM. It is found that ECBCPM outsmarts LSTBCPM in terms of PDR and throughput on an average by 5% and 4% respectively and it also decreases the control overhead and total overhead by 6.4% and 7.2% respectively.

4.3 Summary

In this chapter, two proposed conditional probability based selfish node mitigation mechanisms such as ECBCPM and LSTBCPM are presented. It is observed from the experimental results that the ECBCPM is found to outperform LSTBCPM in terms of throughput, control overhead, total overhead and delay. Further, the dynamic change of mobile node’s behaviour can be forecasted using futuristic trust coefficient based on Semi-Markov process. Hence, the forthcoming chapter 5 focuses on Futuristic Trust Coefficient-Based Semi-Markov Prediction (FTCBSMP) process.