

CHAPTER 2

Chapter 2. LITERATURE SURVEY

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Name of the Sub Title</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Quality of Service in MANETs</td>
<td>16-16</td>
</tr>
<tr>
<td>2.2</td>
<td>Quality of Service Parameters</td>
<td>16-21</td>
</tr>
<tr>
<td>2.3</td>
<td>QoS Frameworks for Adhoc Networks</td>
<td>22-24</td>
</tr>
<tr>
<td>2.3.1</td>
<td>INSIGNIA</td>
<td>22-24</td>
</tr>
<tr>
<td>2.3.2</td>
<td>SWAN Model</td>
<td>24-25</td>
</tr>
<tr>
<td>2.4</td>
<td>Classification of QoS Approaches</td>
<td>25-26</td>
</tr>
<tr>
<td>2.5</td>
<td>QoS Routing Protocols</td>
<td>26-29</td>
</tr>
<tr>
<td>2.5.1</td>
<td>AODV Based Extension QoS Routing Protocols</td>
<td>26-29</td>
</tr>
<tr>
<td>2.5.2</td>
<td>DSR Based Extension QoS Routing Protocols</td>
<td>29-31</td>
</tr>
<tr>
<td>2.5.3</td>
<td>OLSR Based Extension QoS Routing Protocols</td>
<td>31-32</td>
</tr>
<tr>
<td>2.5.4</td>
<td>TORA Based Extension QoS Routing Protocols</td>
<td>32-33</td>
</tr>
<tr>
<td>2.6</td>
<td>Trust in MANETs</td>
<td>33-33</td>
</tr>
<tr>
<td>2.6.1</td>
<td>Trust Definition</td>
<td>33-33</td>
</tr>
<tr>
<td>2.6.2</td>
<td>Trust Metrics</td>
<td>33-34</td>
</tr>
<tr>
<td>2.6.3</td>
<td>Trust Properties</td>
<td>34-34</td>
</tr>
<tr>
<td>2.6.4</td>
<td>Trust Computation</td>
<td>35-36</td>
</tr>
<tr>
<td>2.6.5</td>
<td>Trust Based Protocols in MANETs</td>
<td>36-42</td>
</tr>
</tbody>
</table>
CHAPTER 2

LITERATURE SURVEY

In this chapter, the recent and past relevant work that has been done on QoS and Trust issues in the area of MANETs is discussed along with the drawbacks.

2.1 Quality of Service in MANETs

Implementation of applications with the required level of QoS in MANETs is a difficult task when compared to wired networks [18, 41, 42]. Effective implementation of multimedia or error sensitive applications demands the desired level of QoS paths between source and destinations. But the nature of MANETs like multi hop communication, limited resources, error prone wireless medium and lack of coordination are the obstacles in the provisioning of QoS in MANETs.

2.2 Quality of Service Parameters

A QoS protocol should support different quality metrics in data transmission. Based on application requirements, there are many quality metrics in MANETs. Here the computation of QoS parameters like energy, bandwidth, Reliability and Link Expiry Time (LET) are discussed, which are used in the proposed methods.

2.2.1 Link Band Width

In Time Division Multiple Access (TDMA) method [54,57], the link bandwidth is calculated in terms of time slots. Here the link bandwidth between two nodes depends on the common free time slots of the nodes.

MANET topology can be modelled as a graph(N,L) , where N and L represent set of nodes and links respectively. NB_i is the set of neighbours of node n_i i.e. \( NB_i = \{n_j \in N : (n_i, n_j) \in L \} \). At each node, the time slots are represented as \( S = \{s_1, s_2, s_3, \ldots, s_m\} \). For every node, TS_i (transmission schedule) is represented as a set of time slots, where a node can transmit the data. The receiving schedule RS_i is defined as a set of time slots, where a node can receive the data.

For a node n_i, the time slot s_t can be used to transfer the data to it’s neighbour node n_j based on the following conditions.
1. $s_t$ should not be assigned as transmitting/receiving slot at both the nodes $n_i$ or $n_j$.

2. $s_t$ should not be assigned as receiving slot at any neighbour node ($n_k$) of a node $n_i$.

From two conditions,

$$TS_i = \{s_i \in s : s_i \notin TS_i, s_i \notin RS_i, s_i \notin U_{n_i \in NB}, RS_k\}$$ (2.1)

The receiving slot $s_t$ of $n_i$ from its neighbour node $n_j$ is defined as per the below two conditions

1) $s_t$ should not be assigned as transmitting/receiving slot at the both nodes $n_i$ or $n_j$.

2) $s_t$ should not be assigned as transmitting slot at any neighbour node ($n_k$) of a node $n_i$.

From two conditions,

$$RS_i = \{s_i \in s : s_i \notin TS_i, s_i \notin TS_i, s_i \notin U_{n_i \in NB}, TS_k\}$$ (2.2)

In the Figure 2.1, each node has 7 time slots. Here the possible time slots for data transfer from node A to B are computed as per TDMA. Time slots 1, 2 are scheduled as a transmission and receiving slots of nodes A, B respectively. Slot 4 is assigned as receiving slot at node C. Slot -6 is scheduled as transmission slot at node-D. As the remaining common slots at nodes A, B are 3, 5, and 7, the link bandwidth between A, B is three time slots.
Throughput is defined as the number of data packets delivered at destination node in unit time, which depends on bandwidth.

2.2.2 Threshold Energy

Calculation of threshold energy is proposed in [5, 44]. In data forwarding, a node has to receive and transmit the data. At each node, the energy is consumed for amplifier activation ($E_{amp}$) and transmitter/receiver activation ($E_{ele}$). In forwarding of k-bit message over d distance, a node has to spend $E_{Tx}$ amount of transmission energy and $E_{Rx}$ amount of receiving energy.

$$E_{Tx}(k, d) = E_{amp} \times k \times d^2 + E_{ele} \times k$$  \hspace{1cm} (2.3) \\
and

$$E_{Rx}(k) = E_{ele} \times k$$  \hspace{1cm} (2.4) \\
Total energy consumed is

$$E_{total}(k) = E_{amp} \times k \times d^2 + 2 \times E_{ele} \times k$$  \hspace{1cm} (2.5)

2.2.3 Link Expiry Time (LET)

In [6, 44], the Link Expiry Time is calculated between two nodes. The link life time between two nodes depends on their velocities and direction of movement. Let’s consider A, B are two nodes. $(i,j), (k,l)$ are their locations and $s_1, s_2$ are their velocities. $\phi_1, \phi_2$ are their moving directions where$(0 \leq \phi_1, \phi_2 < 2\pi)$.

The Link Expiry Time of two nodes is calculated as in eq(2.6)

$$LET = \frac{-(mn + op) + \sqrt{(m^2 + o^2)r^2 - (mp - on)^2}}{(m^2 + o^2)}$$  \hspace{1cm} (2.6) \\
Here, $n = i-k$, $p = j-l$, $m = s_1 \cos \phi_1 - s_2 \cos \phi_2$, and $o = s_1 \sin \phi_1 - s_2 \sin \phi_2$.

2.2.4 Node Reliability

In [10], node reliability is evaluated based on its packet forwarding attitude. Through direct interactions, a node can estimate the other node reliability. In the wireless medium, a node can overhear its neighbour node activities. Reliability of a node depends on the number of packets it received and forwarded correctly.
In Bayesian theorem, node reliability is a random variable whose value lies in [0, 1]. Here if \( p \) number of total packets are received at a node and among them \( q \) number of packets are forwarded by it correctly, then node reliability computed as in equation (2.7).

\[
f(r, p / q) = \frac{P(q / r, p) f(r, p)}{\int_0^1 P(q / r, p) f(r, p) dr}
\]  
(2.7)

Here using binomial theorem, \( P(q/r, p) \) is computed as in eq(2.8)

\[
P(q / r, p) = \binom{p}{q} r^q (1-r)^{p-q}
\]  
(2.8)

\( f(r, b) \) is represented as a beta distribution

\[
Beta(r; \alpha, \beta) = \frac{r^{\alpha-1} (1-r)^{\beta-1}}{\int_0^1 r^{\alpha-1} (1-r)^{\beta-1} dr}
\]  
(2.9)

Finally beta distribution can be applied to compute \( f(r, p/q) \) as in eq (2.10)

\[
f(r, p | q) \equiv \frac{\binom{p}{q} r^{\alpha+q-1} (1-r)^{\beta+p-q-1}}{\int_0^1 P(q / r, p) f(r, p) dr \left( \int_0^1 r^{\alpha-1} (1-r)^{\beta-1} dr \right)}
\]  
(2.10)

i.e

\[
f(r, p | q) = Beta(\alpha + q, \beta + p - q)
\]  
(2.11)

The expected reliability of node is

\[
E(r) = \frac{\alpha}{\alpha + \beta}
\]  
(2.12)

Here the parameters \( \alpha, \beta \) parameters are represented as \( \alpha_n = \alpha_{n-1} + q_{n-1} \), \( \beta_n = \beta_{n-1} + p_{n-1} - q_{n-1} \). Initially a node doesn’t have any experience about neighbour node behaviour so \( \alpha_0 = \beta_0 = 1 \) and its reliability starts with 0.5. Here node reliability can be represented in iteratively as

\[
E_n(r) = \frac{\alpha_n}{\alpha_n + \beta_n}
\]  
(2.13)
2.2.5 Packet Delivery Time

It is the average travel time of a packet, from the source node to destination node. This delay is influenced by network congestion conditions and buffer time at intermediate nodes. If the node buffer time is high, a packet has to remain for a longer time in the buffer. This increases the packet delivery time. In case of link breaks, packets are stayed at the buffer and will be retransmitted after link repair.

2.2.6 Jitter Delay

In MANETs, the jitter delay represents the average deviation of each packet’s delay from average delay.

\[
\text{jitter Delay} = \frac{\sum_{i=1}^{n} |D_i - D|}{n} \tag{2.14}
\]

In the network, \( D \) represents average packet delay, \( D_i \) is the delay of \( i^{th} \) packet and \( n \) represents the total number of data packets. Jitter delay is the good quality metric in signal processing. Higher jitter delay can decrease the quality of live video session.

2.2.7 Packet Delivery Ratio (PDR)

It is the fraction of data packets arrived at destination node and the number of packets generated in the network (including duplicate packets). The value lies between [0,1].

\[
PDR = \frac{N_D}{N} \tag{2.15}
\]

Here \( N_D \) is the number of actual data packets reached at destination. \( N \) is the number of total packets used for data transmission. Link breaks demands retransmission of data packets which increases the denominator and hence reduce the PDR value.

2.2.8 Control Over Head (COH)

It is the fraction of total control packets used in the protocol and the number of data packets reached at destination. Its value lies between [0, 1].

\[
COH = \frac{N_C}{N_D} \tag{2.16}
\]
Here $N_C$ is the number of control packets and $N_D$ is the number of Data packets. If the protocol introduced the additional control packets then its COH is increased.

### 2.3 QoS Frameworks for MANETs

QoS framework is a comprehensive system that provides the services as per user requirements. The framework is the combination of many components, where all the components work together to provide the assured services. The main components of the QoS framework are QoS service model, QoS routing, QoS signalling, call admission control, QoS medium access control and packet scheduling schemes. The components are described as follows,

- **QoS Service model**: It is defined as the type of Quality provision either per class basis or per session basis. All the users having similar requirements are considered as a class.

- **QoS Routing Protocol**: The objective of the routing protocol module in the framework is to find out the route between source and destination nodes that fulfils the user QoS requirements. If several paths are available that satisfies QoS requirements, then routing protocol maintains this information, which helps to resume the services immediately in case of path failure. The performance of the routing protocols can influence the QoS framework performance. A routing protocol should minimum the number of control packets (control overhead) that are used to monitor network changes. The routing protocol has to communicate and work with other components like admission control, resource management, and signalling for end to end QoS provision. All the routing operations should use less amount of resources and adopt all the flow state and network state changes.

- **QoS Resource Reservation Signalling**: After establishing the path by routing protocols, all the necessary resources are reserved at each intermediate node by the resource reservation signalling protocol. For this, the signalling protocol sometimes communicates with QoS MAC (Medium Access Control) layer sub systems. The reserved resources are released only after completion of session
• Admission control: The admission control has to take decision whether to accept or reject new QoS service request. If the service is accepted, then signalling protocol reserves the resources. The newly accepted service should not disturb the QoS assurance that is given to already accepted services. The performance of QoS framework is evaluated in terms of number of the QoS sessions that are being handled by it and is represented by the average call acceptance ratio (ACAR).
• Packet scheduling: If multiple QoS sessions are running through a link simultaneously, then the scheduling scheme decides the QoS flow that has to be served next. If multiple packets are waiting in the queue of the node, those belong to different delay constrained sessions then scheduler schedules the packet transmissions. The effectiveness of the packet scheduler depends on the number of packets that are transmitted within their deadlines.

2.3.1 INSIGNIA

In [58], INSIGNIA (in band signalling system) is proposed, which is the first QoS framework for Mobile Adhoc Networks as shown in figure 2.2.

![Diagram of INSIGNIA QoS Framework](image-url)

Figure 2.2 INSIGNIA QoS Frame Work

The framework functionality mainly depends on the component INSIGNIA, which coordinates all other components for achieving quick restoration and
reservation to support adaptive services. In the frame work, the signalling system is a less complex component, which can respond quickly to the all QoS conditions and network topology changes. As shown in the figure2.2, the frame work has the components like routing module, Admission control, In-band signalling, Packet Scheduling, Packet forwarding and Medium access control (MAC).

- Routing module: It finds a route from source to destination node, which satisfies the user QoS requirements. In data transmission, the data packets are forwarded to the next hop intermediate node by the routing module. In this framework, routing module works independently without depending on other components, so that any routing protocol can be chosen. The routing module in the INSIGNIA has path recovery ability in case of link failures.

- In-band signalling: It is used to support the adaptive services between source and destination nodes. In this framework, the signalling module is independent of link layer protocol. The in-band signalling systems can reduce the control over head by maintaining the control information as a part of data packets. Here, an optional QoS field namely INSIGNIA in the data packet is used to maintain the control information. The in-band signalling system can support even higher data packet transmissions and hence it is suitable for wireless dynamic networks like MANETs.

- Admission control: it coordinates the acceptance/rejection of newly arrived QoS requests. In this framework, admission control follows the flow based resource allocation. The framework implements the soft QoS guarantee, where the resource reservations are periodically refreshed. The captured resources are released completely after completion of data transfer.

- Packet forwarding: It segregates the arrival packets and distributes them to the suitable components. If the data packet has INSIGNIA option, then it is sent to signalling module. If the packet is at intermediate node, then it is forwarded to next hop intermediate node with the aid of routing protocol. If the data packet is at the destination node, then it is sent to the one of running applications.

- Packet scheduling: When several packets are waiting in a queue at a node, the packet scheduler schedules the packets to get forwarded to the next hop node. INSIGNIA follows the weighted round-robin service discipline to forward the packets.
• Medium access control (MAC): In case of real time services, the framework uses the MAC layer services. The MAC protocol coordinates the access requests over the wireless medium to support QoS requirements.

2.3.2 SWAN Model

SWAN (stateless wireless ad hoc networks) is a QoS Service model proposed by Ahn et al[55] as shown in figure 2.3. It is a distributed network model, which takes MAC protocol services to support real time applications by implementing the feedback-based control mechanisms. SWAN uses different control mechanisms like local rate, source based admission and an explicit congestion notification (ECN) for handling new service requests like best-effort traffic, real-time sessions and dynamic real time sessions respectively. Unlike the framework INSIGNIA; the nodes are free from the responsibilities of maintaining aggregate state or per flow state information. In spite of network changes like node and link failures, the SWAN system can provide uninterrupted QoS support to the applications. Hence this system is robust, scalable and simple.

![SWAN Model Diagram](image)

Figure 2.3 SWAN Model

The different control modules are depicted in the figure 2.3. The packet classifier module receives packets from higher layer i.e IP layer. It classifies all the marked packets as real time packets and unmarked packets as best effort packets. All the best effort packets are forwarded to traffic shaper module and real time packets...
are directly forwarded to MAC layer. The traffic shaper follows the leaky bucket policy in forwarding packets to MAC layer. In the traffic shaper, the best effort packets are released at the rate confirmed by the traffic rate controller module. The call admission controller module takes the decision of either acceptance or rejection of new services. These decisions are made based on request and response probe packets.

**2.4 Classification of QoS Approaches**

In the MANETs, the classification of QoS solutions is based on two metrics. One is depending on the QoS provisioning mechanism; another is the layer where it is implemented.

**Classification of Quality of Service Approaches**

- **Based on communication between QoS Provisioning and routing mechanism**
  - Coupled: TBP, QoSAODV, TDR, INORA, AQR
  - Decoupled: INSIGNIA, SWAN, PRTMAC
- **Based on communication between MAC and Network Layer**
  - Independent: TBP, QoSAODV, TDR, INSIGNIA, SWAN
  - Dependent: TDR, BR, AQR, PRTMAC, CEDAR
- **Based on the routing information Update mechanism**
  - On-Demand: TBP, TDR, BR, AQR, PRTMAC, CEDAR, INORA, QoSAODV
  - Table-Driven: PLBQR
  - Hybrid: BR, CEDAR

**Figure 2.4 Classification of QoS approaches**

In figure 2.4, different approaches are used for classifying the QoS solutions i.e. (a) based on communication between QoS approach and routing protocol (b) based on communication between MAC and network layer (c) based on routing information updating methods. In the communication of routing protocol and QoS provision, the QoS solutions are of two types i.e coupled and decoupled. In coupled approach, QoS provision strongly depends on QoS routing, where if the routing protocol is changed then this approach could not assure the Quality of Service guarantee. In the communication of network layer with MAC layer, QoS solutions are
of two types like dependent and independent. In depended network layer takes the assistance of MAC layer in QoS provision. In routing information updating mechanism, QoS Solutions are of three types i.e on demand (reactive), table driven (proactive) and hybrid. In proactive, each node has a routing table, where it maintains the information of all the nodes in the network, which is used in route establishment. In reactive method, no routing tables are used by the nodes. But the source node initiates the route discovery process, when it is required. In Hybrid method, the routing information is updated based on the combination of proactive and reactive methods.

2.5 QoS Routing Protocols

Classical routing protocols concentrated on finding the shortest path from the source to destination but sometimes these established paths are not able to transfer data. Hence, the QoS routing protocols [9,18,92] are the extension of classical routing protocols, which select intermediate nodes with threshold level of quality parameters.

In below sections the QoS protocols, which are extension of basic routing protocols, i.e. AODV, DSR, OLSR and TORA are discussed.

2.5.1 AODV Based Extension QoS Routing Protocols

Here first the AODV routing protocol is discussed and later all the routing protocols those are QoS extension of AODV are discussed.

2.5.1.1 Adhoc On demand Distance Vector (AODV)

In AODV [94], the route is discovered by the source node when it is required (on demanded). AODV uses the destination sequence numbers (DestSeqNum) to establish the route to destination. The higher DestSeqNum represents the latest route information of the destination node. In routing process, source node prepares the Route Request (RREQ) packet and sends to the destination node through intermediate nodes. RREQ Packet contains the information like source sequence number (SrcSeqNum), source identifier (SrcID), destination sequence number (DestSeqNum), destination identifier (DestID), broadcast id (BcastID) and time to leave (TTL).

When a source node has the data to share with destination node, it prepares the Route Request (RREQ) packet and sends to all one hop neighbours. On receiving
RREQ packet, an intermediate node gives the reply (Route Reply-RREP) packets to source node if it has valid route information of destination node (i.e DestSeqNum value at the node is greater than the RREQ packet value). Otherwise it forwards the request packet to next hop neighbour. If multiple RREQ packets are received by an intermediate node, then it eliminates the duplicate RREQ packets based on (BcastID, SrcID) combination. After receiving all the RREQ packets, the destination node gives reply through a RREP packet to the source node. While forwarding RREQ/RREP packets, an intermediate node can gather route information towards the source and destination nodes respectively. This routing information is maintained in the routing table.

2.5.1.2 QoS Routing Protocols based on AODV

Zeinalipour et al.[88] proposed the AQODV(Adhoc QoS On demand Distance Vector), where the minimum delay and maximum bandwidth are considered as QoS parameters.

![Diagram showing minimum delay](image)

**Figure 2.5 Minimum Delay**

- **Minimum Delay:** In the figure 2.5, AQODV is explained considering delay as a QoS constraint. The delay represents the amount of time that a data transmission takes to get completed between source and destination. NODE_TRAVERSAL_TIME (NTT) of each node represents the amount of time that is required by it to process the RREQ packet. This time value depends upon the node processing power. Each node forwards the RREQ packet only if its NTT value is less than the RREQ packet delay otherwise it discards the packet. Before forwarding RREQ packet, each node subtracts its NTT value from the RREQ packet delay. In the figure 2.5, the source node(ingress-A) sends the RREQ1 packet with 100 milliseconds delay. At each core
router (intermediate node), this delay is subtracted by node NTT values, here the NTT value of nodes B and C are 30 and 50 respectively. On receiving RREQ packet, the destination node (egress D) gives the reply with initial delay value of 0. This delay is incremented by the NTT value at each node towards the source node and is recorded in the routing table for handling further RREQ packets. In the figure 2.5 the RREQ2 packet with delay 10 milliseconds is dropped by the node B, since node B has the routing information about delay to destination node as 80 ms. In the network, each node updates its NTT value changes through the small sized ICMP_QOS_LOST messages to all its affected neighbour nodes.

![Image](image_url)

**Figure 2.6: Maximum Bandwidth**

- **Maximum bandwidth:** In the figure 2.6, AQODV is explained by considering bandwidth as a QoS constraint. Here the ingress A (source node) sends the RREQ packet with minimum required bandwidth value that each link should have along the path. On receiving RREQ packet, an intermediate node compares its available bandwidth with required bandwidth. If the node lacks required bandwidth, then it drops the RREQ packet otherwise forwards the packet to the next hop node. The destination node gives the reply RREP to the RREQ packet it has received. In the RREP packet, the bandwidth filed is initialized to a large value (infinitive). While the RREP packet traverses through the intermediate nodes, it compares all the link bandwidths and maintains the minimum link bandwidth value (bottleneck bandwidth). Even intermediate nodes maintain this value for future request purpose. In figure 2.6 source node released RREQ2 packet with required bandwidth of 80Kbps, but it is dropped by the node B, since it has the information that bottleneck bandwidth to
destination node is 50. Each node updates its bandwidth changes through the ICMP_QOS_LOST messages to all its affected neighbour nodes.

Ad Hoc QoS On-demand Routing (AQOR) is proposed in [51], which is on demand QoS routing protocol. The source node creates route request packet with required bandwidth and time delay. On receiving the RREQ packet, an intermediate node verifies its bandwidth. If it has sufficient bandwidth, it creates an entry in routing table and forwards to the next hop node. The entry is eliminated, if the reply packet does not reach the node. Hence the replay packets, those arrived within delay are allowed to reach the source. With these requirements, a source node can establish the route with more bandwidth and small delay, but there is no route break prediction mechanism.

Espes et al. [37] proposed Delay and Bandwidth Constrained Routing protocol DBCTO based on TDMA approach. Here the protocol selects the intermediate nodes with the less number of neighbours (i.e node with high bandwidth). The RREQ packet is extended with bandwidth, delay requirements, number of neighbours and time slot information. But the protocol approach has deviated from the classical AODV, since intermediate and destination nodes can influence the route selection.

Shayesteh et al. [30] proposed QoS routing protocols as an extension of AODV. In the route selection, the additional parameters are considered like mobility, energy, bandwidth, and radio signal strength and transmission range. A weightage based sum function is used to combine all parameters and the resultant value is used in route selection. But the protocol can’t give strict QoS guaranty.

Agbaria et al. [38] presented a dynamic Bandwidth Management (dBM) protocol, which can assure bandwidth requirement even at higher node velocities. Here each node announces its bandwidth requirements up to 2-hop neighbours. Hence the resource reservation is limited to the area of 2-hop distance.

2.5.2 DSR Based Extension QoS Routing Protocols

Here the DSR routing protocol is discussed followed by a discussion on all the routing protocols namely QoS extension of DSR.
2.5.2.1 Dynamic Source Routing (DSR)

In DSR [96], the source node sends RREQ packets towards the destination node for route discovery. An intermediate node, on receiving RREQ packet, sends reply (RREP) packet back to the source node if it has valid route information. Otherwise, it forwards the RREQ packet to next hop neighbour. Based on sequence number in RREQ packet, an intermediate node can avoid the duplicate packets. The RREQ packets maintain the sequence of intermediate nodes information, through which it has traversed to reach the destination node. The destination node receives RREQ packets and gives the reply (RREP) to the RREQ packet, which has travelled through shortest path. On receiving the reply (RREP) packet, the source node establishes the shortest path to destination and starts the data transmission.

2.5.2.2 QoS Routing Protocols Based on DSR

Geng et al. [31] presented QoS multipath routing protocol (QAMR). This takes the help of MAC layer in bandwidth estimation based on channel status (idle / busy). It considered new metrics like node utilization factor and path congestion factor. The protocol establishes the route with less node utilization factor and small path congestion factor. But here, the intermediate nodes can’t give the reply to the route request packets.

Geng et al. [34] proposed Partial Bandwidth Reservation (PBR) scheme. The intermediate nodes will discard the RREQ packets that are lagging in bandwidth requirement. PBR improved the network capacity compared with full bandwidth reservation. But additional overhead is formed in maintenance of local bandwidth information at each node.

Y. Yang et al. [35] proposed Maximum Bandwidth Routing Protocol. Here the RREQ packet carries the occupied time slot information of the intermediate nodes. Through the RREQ packets, the destination node can establish the route to source node with maximum bottleneck bandwidth. But the established route reliability is not taken into consideration.

Roy Leung et al. [90] proposed multipath dynamic source routing (MPDSR). The routes are identified based on link reliability. Link reliability represents the
probability of successful data transmission between two nodes as per time constraints. The route is identified through reliable links.

2.5.3 OLSR Based Extension QoS Routing Protocols

Here the OLSR routing protocol is discussed and later all the routing protocols those are QoS extension of OLSR are discussed.

2.5.3.1 Optimal Link State Routing (OLSR)

![Figure 2.7 MPR Node Selection](image)

OLSR protocol is proposed in [89], which is the extension of Link State Routing protocol. The objective of OLSR protocol is to reduce control overhead with the selection of Multi Point Relay (MPR) Nodes. In figure 2.7, the MPR node selection is explained. Each node selects sub set of one hop neighbours as MPR nodes to reach all of its two hop neighbours. Each node communicates with its two hop neighbours through MPR nodes only. Hence it can reduce the duplicate packets in the network. The performance of OLSR depends on the optimal selection of multi point relay nodes.

Each node collects the neighbour nodes details; with those it has the bidirectional link status. For this, each node broadcasts the HELLO packets regularly to all of its neighbour nodes. Through the HELLO packets, each node can gather the information up to two hop neighbours and can select the MPR nodes to reach all the 2-hop neighbours. In the figure 2.7, node N selects the 4 one-hop neighbours as its MPR nodes to reach all of its 16 2-hop neighbours. Each node releases the Topology Control (TC) messages, which has its MPR selector list. From the TC messages, topology table is constructed at each node, which has the information in the form of
order pairs, i.e. [node, node’s MPR selector]. In establishing the route, the source node uses the topology table information to identify the MPR nodes from the destination node to itself.

2.5.3.2 OLSR Based Extension QoS Routing Protocols

Munaretto et al. [53], proposed a time delay constrained QoS routing protocol as an extension work of OLSR. Here, the HELLO packets are attached with their creation time. By exchanging these packets, a node can estimate the transmission delay of their neighbour nodes.

Shoba et al. [15] proposed the QoS aware OLSR protocol (WBTQ). It considered the bandwidth and trust parameters. Based on these parameters, the MPR nodes are selected. The Protocol facilitates to establish either bandwidth preferred or trust preferred routes.

Ge et al. [52] proposed bandwidth aware OLSR protocol. MPR nodes are selected with maximum bandwidth. Bandwidth is estimated based on the percentage of busy time to attain the required level of bandwidth. But the resource reservation and route maintenance are not addressed here.

2.5.4 TORA Based Extension QoS Routing Protocols

Here the TORA routing protocol is discussed and later all the routing protocols those are QoS extension of TORA are discussed.

2.5.4.1 Temporarily Ordered Routing Algorithm (TORA)

TORA is a reactive routing protocol [95], which limits the control packets to the small portion of the network and can establish the loop free routes to the destination node. Here each node maintains Height metric H(N), which represents the distance of the node N from the destination node. The source node begins the route finding process by sending Query packets to the neighbour nodes, Query packet contains the destination ID. On receiving Query packets, an intermediate node gives the replay if it has route information otherwise it forwards to the next hop node. After receiving Query packet, the destination node gives the reply with Update packet, which contains H(N) as 0. While forwarding Update packets, each node increments the H(N) value by 1. Through the Update packets, the source node establishes the
route as a set of directed links towards the destination. In the routing process, the network is maintained as a destination oriented directed acyclic graph (DAG) with $H(N)$ values.

2.5.4.2 QoS Routing Protocols Based on TORA

INORA [58] is the combination work of INSIGNIA and TORA. INSIGNIA has good resource management control and creation or recovery capability of flows. The source node sends RREQ packets with bandwidth requirement. All the intermediate nodes having good resources give the replay and reserve the resources.

Energy and delay aware TORA is proposed in [69]. On receiving RREQ packets, intermediate nodes check their residual energy. If the energy is above threshold value, then only it calculates the delay otherwise it discards the request packet. Hence the nodes that are capable of energy and delay are included along the path.

2.6 Trust in MANETs

In MANETs, a node has to participate and work together with strange nodes to run the network applications. Hence, the computation of a node trust value in quality and attitude wise is certainly an advantage to run the applications efficiently and effectively.

2.6.1 Trust Definition

In the literature, trust is defined in different ways by many authors [14, 59]. But the widely accepted definition is given in [22] and stated as “Trust represents the confidence or belief on the integrity, availability and QoS of target node”.

2.6.2 Trust Metrics

Trust is computed in different ways by applying various metrics. Here different metrics of trust computation are discussed as below.

Trust scale: trust scale represents the possible domain values of trust variable. Some methods [85] used discrete values in [-1, 1] to represent trust level and some methods [68] used continuous values in [0, 1] for trust measurement. In [83], the
threshold based trust is used, if the node behaviour is above the threshold level of satisfaction, then it is considered as trustworthy.

*Trust facets:* In [75], the trust is measured as the combination of \((T, C)\), where \(T\) represents trust vale and \(C\) represents level of confidence on measured trust value. In [67], trust is measured as triplet in 3-D space as shown in figure 2.8 i.e \((b, d, u) \in [0,1]^3\) and \(b + d + u = 1\), where \(b\) is belief, \(d\) is disbelief and \(u\) is uncertainty.

![Figure 2.8: Trust Triplet Space](image)

*Trust logics:* Some methods [70] applied probability concepts in trust calculation. In [77], a node trust value is defined as the ratio of the number of packets it forwarded correctly to the total number of packets it received. In [86], beta distribution is used for node trust evaluation. Here the two parameter of beta distribution \(\alpha, \beta\) are considered as good and bad experiences of the node respectively. In some papers [40, 60], trust is measured using fuzzy logic, where some linguistic terms (adjectives) are used to describe the range of trust values.

### 2.6.3 Trust Properties

Trust will follow three properties that are composability, asymmetry, and transitivity. In trust asymmetry, if node A trusts node B with some trust level, but it is not compulsory that B also trusts the node A with the same level of trust. According to transitivity property, if node A trusts node B and node B trusts node C then node A trusts node C with some level. In composability property, if a node received multiple trust recommendations from different channels then it can combine them to derive single final trust value.
2.6.4 Trust Computation

In MANETs environment, the trust computational methods are majorly classified into three types, i.e direct trust, indirect trust and hybrid method of trust. A node can compute the direct trust of its 1-hop neighbour through the direct interactions. For a multi hop distance node, the indirect trust is evaluated based on trust recommendations from other neighbour nodes. In Hybrid method, trust is evaluated through direct observation and from neighbours trust recommendations.

2.6.4.1 Direct Trust Computation

In [46], a node (trustor) evaluates its neighbour node (trustee) trust value based on the number of packets it has forwarded correctly. Node B trust value is computed by node A using the equation (2.17).

\[
T = R_p \times W(R_p) + R_q \times W(R_q) + R_e \times W(R_e) + D \times W(D)
\]

(2.17)

Here \(D\), \(R_q\), \(R_p\) and \(R_e\) are misbehaviour factors of data packets, route request, route reply, and route error packets respectively and are defined in eq(2.18) as normalized values [0, 1].

\[
R_q = \frac{R_{qs} - R_{qs}}{R_{qs} + R_{qs}} \quad R_p = \frac{R_{ps} - R_{ps}}{R_{ps} + R_{ps}} \quad D = \frac{D_s - D_f}{D_s + D_f}
\]

(2.18)

Here \(R_{qs}, R_{es}, R_{ps}\) and \(D_s\) are defined as number of route request, route error, route reply and data packets that are successfully forwarded by node B. \(R_{qf}, R_{ef}, R_{pf}\) and \(D_f\) are the number of route request, route error, route reply and data packets that are not forwarded (failed) by node B. Here \(W()\) represents the weightages that are given to different packets in trust calculation.

2.6.4.2 Recommendation Based Trust

In [78], the recommendations based trust is proposed. Here the trust is calculated based on confidence value, that is \(c_{ij}\) represents the node i confidence level on node j, where \(c_{ij} = [-1, 1]\). Confidence metric is asymmetric i.e \(c_{ij} \neq c_{ji}\), hence the confidence value between \(i, j\) nodes is calculated as

\[
\hat{c}_{ij} = \frac{c_{ij} + c_{ji}}{2}
\]

(2.19)
\( T_i(k) \) is the trust value of node \( i \) at \( k \) instant of time. \( T_i(k+1) \) is calculated as in equation (2.20)

\[
T_i(k+1) = \begin{cases} 
1, & m_i(k) > \tau \\
-1, & m_i(k) < \tau 
\end{cases}
\] (2.20)

Here \( m_i(k) \) is the trust value of node \( i \) based on the recommendations of other nodes. \( \tau \) is the trust threshold value.

\[
m_i(k) = \sum_{j \in N_i} \hat{c}_{ji} T_j(k)
\] (2.21)

### 2.6.4.3 Hybrid method

Here, the trust value of a node is evaluated based on own observations and recommendations from other nodes. In [97], the hybrid trust of node \( b \) is evaluated by node \( a \) as in equation (2.22)

\[
T_{a,b} = \alpha T_d + \beta T_i
\] (2.22)

Here the \( T_d \) is the direct trust and \( T_r \) is the recommended trust of the node \( b \). \( \alpha, \beta \) are the weights of the direct and recommended trust values respectively, where \( \alpha + \beta = 1 \). \( T_d \) is computed based on forwarding delay, number of dropped packets, misrouted packets and wrongly injected packets by node \( i \). The recommended trust \( T_r \) is calculated as the average weighted sum of trust proposals of other nodes.

### 2.6.5 Trust Based Protocols in MANETs

Here the existing trust handling protocols in MANETs are discussed. In MANETs, the trust based protocols are majorly classified into four interrelated categories as shown in figure 2.9, i.e trust computation, trust aggregation, trust propagation and trust routing. Trust computation protocols discussed trust computational methods through the direct interactions. Trust aggregation protocols refer the trust computational methods of multihop neighbours through recommendations of other nodes. In trust propagation protocols, the computed trust spreads in the network to avoid the recomputational risk. Trust routing protocols establish the route based on the node trust values.
2.6.5.1 Trust Computation Protocols

The protocol in [2], computes the trust based on nodes historical behaviour patterns. For modelling trust value they used dynamic grey markov chain model, where the trust changes are described through state transitions.

In [23], a node trust value is computed based on its packet forwarding nature. The misbehaviour factor of a node is computed as the ratio of number of packets it has forwarded correctly among the total number of received packets. A node trust value is computed in terms of misbehaviour factor of data and control packets (route error, route reply and route request). G. Lenzini, et al[67]. Defined a node trust value based on its three different attitudes like disbelief- $d$, belief- $b$, and uncertainty- $u$ with the condition $b + d + u = 1$. A node belief and disbelief values are incremented for each of its positive and negative activities respectively. A.Josang et al. [86] applied beta distribution for evaluation of node trust value, he considered two parameters $\alpha$ and $\beta$ as good and bad experiences with respect of forwarded and dropped packets. Some authors applied fuzzy logic to quantify node trust value by considering nodes different behaviour patterns as fuzzy sets [98]. Muhammad et al [1] proposed validation based trust value; here one trustworthy entity (which is referred as watchdog) is used for validating computed trust value. If the validated trust value is not deviated above the threshold value, then the trust value is taken as a final trust value. In [3], the trust value (final trust value) is computed by grouping its DTV(Direct Trust Value) and IDTV(Indirect Trust Value), DTV is computed based on data consistency and packet forwarding attitude. IDTV is evaluated by taking into consideration of multiple recommendations of other nodes at particular instance of
time. Similarity based trust value is defined in [4] between two node. The similarity is verified over the common attributes of the nodes. Here the computed trust value is getting deducted at their aging factor. In [10], trust is evaluated using fuzzy logic. Bayesian based conditional probability is applied for trust derivation.

A novel composite trust is proposed by Ing-Ray Chen et al. in [19]. Here the trust is defined as the Social QoS Trust (SQTrust), which is evaluated as the composition of Social Trust and QoS Trust. Social Trust of a node is computed in terms of its social relationships with its neighbours like friendship, similarity, privacy, and honesty. A node QoS trust is computed based on its reliability, cooperation and competence. But in this work, major focus is on theoretical concepts and small amount of implementation discussion is done.

The context based MobiFuzzyTrust is proposed by Fei Hao, et al. in [16]. The MobiFuzzyTrust is computed in three phases. In the first phase, between two nodes the similarity of two nodes is computed in different contexts (i.e familiarity, prestige, location and time). In second phase, the trust is measured based on similarity index. In the third phase fuzzy concepts are applied for conversion of trust value into linguistic terms like Low, Medium or High. But in this work, many approximations are made while applying these concepts to MANETs.

In [105], the fine grained analysis (FGA) method is used to improve the trust mechanism in MANETs. Usually packet loss parameter is used to estimate the node trust value, but in the network there may be many causes for packet loss like queue over flow, channel interference and mobility. Here the proposed method used FGA to investigate the real cause of packet loss and to improve the trust accuracy. In [106], multilevel trust based intrusion detection system is proposed with the help of elliptic curve cryptographic algorithms to identify different types of attackers. Here the proposed discussed the identification of three different types of attackers like flooding attack, selective packet dropping attack and black hole attack. In [103], game theoretical approach is followed to estimate node trust values and elimination of malicious nodes. The supervisory game theory model is used for monitoring selfish behaviour of nodes which can reduce the monitoring cost.

In [104] the social networking activities are discussed in the MANET. To protect the data in communication, two dimensional trust value is proposed. The trust
is established either at server side or at user level to identify harmful and unsecured activities of users and to improve the secure environment in social networks. In [107], the proposed routing protocol addresses both secure and energy parameters. The energy consumption is minimised by distributing the load in the network equally to all the intermediate nodes and the security is achieved by calculating trust values. Here the Ant Colony Optimization (ANT) method is enhanced as Ant Density Algorithm, where the routing algorithm selects the next hop intermediate nodes based on distance and their activities. A routing protocol has to re-establish the path in case of path breaks, but this risk can be avoided by identifying multiple paths to destination. In [108], the multi path routing protocol is proposed. The proposed routing protocol incorporates the trust value concept to the classical multipath routing algorithm. Here the trust values of nodes are estimated based on their previous interaction and recommendations from other nodes.

2.6.5.2 Trust Aggregation protocols

In [61], a node receives trust value of the target node in the form of gossips/rumours from multiple recommender nodes. The push sum operations are applied over the collected gossips to derive final trust value. In [62], the probability like approaches are applied for trust aggregation. Here the trust is aggregated either in sequential or parallel methods. In sequential, all trust recommendations from trust node to trustee are averaged. In parallel method, node receives trust recommendations from different paths. The recommended trust values are combined using different weights to the paths.

BoWang et al.[12], proposed the trust model, where a node follows hybrid trust computational methods to evaluate its neighbour node trust value, here the hybrid trust value is computed based on direct interactions and by considering other nodes trust recommendations. In trust computation, the QoS parameter (Link Delay) is considered. This is estimated using ETX (expected transmission count) method. In ETX, each node sends the probe packets to its neighbour nodes. The link delay with the neighbour node is estimated based on number of probe packets are dropped by the neighbour node.
Recommendations based trust is proposed by Antesar et al. in [8]. To evaluate multi hop away node trust value, an evaluator node considers the recommendations of its neighbour nodes. It may receive false recommendations from some of the neighbour nodes, to minimise the impact of false recommendations it applies the clustering technique. The evaluator node validate each recommender node in three ways i) confidence level on trust recommendations ii) trust centrality closeness with evaluator node iii) trust variation with threshold value. Here the security is ensured through trust but QoS issues are not discussed.

2.6.5.3 Trust Propagation Protocols

In [33,], social network principles are adopted in propagation of trust values. Here a node propagates the computed trust value to its direct neighbours and then to multi-hop neighbours. In hop by hop propagation, the trust value keeps getting deducted by d-factor (i.e. forwarding node trust). This flow of transition is continued till the trust value is reached the threshold level of trust value. For trust propagation, the MANET is considered as a graph. They used transitive graphs for trust propagation by applying small world approach. In [29], the trust is propagated in the form of trust tickets, a node initiates the trust request ticket and the trust provider (which has required trust information) replies through trust ticket. Both the tickets will meet at some reliable node (rendezvous) and then trust information reaches to the requester node. Here the rendezvous node is selected based on probability calculations.

2.6.5.4 Trust routing

Hui Xia et al[20]. Proposed the Trust-based Source Routing protocol (TSR), where each intermediate node trust value is evaluated based on packet forwarding ratio. At time $t$, node $v_i$ trust value on node $v_j$ is represented as $TV_{ij}(t)$. This value is evaluated as the combination of $CFR$ (Control Forward Ratio) and $DFR$ (Data Forward Ratio) of the node $v_j$. Here for the node $v_j$, $CFR$ and $DFR$ are defined like

$$\text{CFR} = \frac{\text{number of control packets forwarded}}{\text{total number of control packets received}}$$  \hspace{1cm} (2.23)

$$\text{DFR} = \frac{\text{number of data packets forwarded}}{\text{total number of data packets received}}$$  \hspace{1cm} (2.24)
The trust value \(TV_{ij}(t)\) is evaluated as in eq(2.25),

\[
TV_{ij}(t) = w_1 \times CFR_{ij} + w_2 \times DFR_{ij}
\]  
(2.25)

Where \(w_1 + w_2 = 1\)  
(2.26)

The route trust value between source \((v_s)\) and destination \((v_d)\) nodes at time \(t\) \((TV_{sd}(t)\) ) is evaluated as the continued product of all the intermediate nodes trust values along the route as shown in eq(2.27),

\[
TV_{sd}(t) = \prod \left( \left\{ TV_{ij}(t) \left| v_i, v_j \in P \text{ and } v_i \rightarrow v_j \right. \right\} \right)
\]  
(2.27)

In the equation (2.2.7), \(P\) represents the path/route and \((v_i, v_j)\) represents the all pairs of adjacent intermediate nodes along the route.

In figure 2.10, between source (S) and destination (D) nodes, there are three possible routes (i.e S-P-R-D, S-P-T-D and S-Q-T-D). But S-P-R-D is selected as the route for data transmission, since it has the higher route trust value (i.e. \(0.9 \times 1 = 0.9\)). Here trust value of destination node is 1.

In [36], trust based AODV protocol is proposed, which is the trust enhancement of the AODV. Here, the source node begins the trustworthy route discovery by sending trust request (TREQ) packets. These packets collect the trust information of the intermediate nodes, while travelling towards destination. The destination node selects the trust worthy route based on the trust information carried by TREQ packets and gives the reply to source node through TREP. In [45], trust routing protocol (TDSR) is proposed as an extension work of DSR. Here the trust value is increased for every successful transmission of data packet (i.e. acknowledge packet received) and the trust value decremented for every retransmission.
In [32], the trust is computed based on Intrusion Detection System (IDS) and local information. They used the AODV protocols to implement the trust. This light weight trust computation method increased the resource utilization. In [20], trust based source routing protocol is proposed, where the trust is evaluated as the combination of historical trust and current trust. Historical trust is computed based on packet forwarding ratio and current trust is evaluated based on battery power, CPU cycles, local memory and bandwidth. In [56], DSR based trust routing protocol is proposed. Request packet carries the route information in DSR; this nature is taken as an advantage in trust propagation. In [1], fuzzy based AODV is proposed, where trust is evaluated using fuzzy logic. For trust validation they defined threshold trust value. In the network the malicious nodes are separated by identifying its suspicious activities through fuzzy inference rules.

In [23], Sridhar et al. tried to improve the QoS provision by computing node trust values. In the MANETs, the QoS is improved by identifying and avoiding malicious nodes. Here a node trust value is computed in terms of its packet forwarding ratio of different packets (i.e Data packets, Route Reply-RREP and Route Request-RREQ). Nodes with less trust values than threshold level are treated as malicious nodes. But this research work is lack of good route recovery system.

Trust based Meta heuristic genetic algorithm is proposed by Zafar Sherin et al. in [11]. In the first stage, genetic algorithm is used to establish the QoS enabled route from source to destination. In the second stage, each intermediate node attaches trust weigh value to the packet before forwarding it to the next hop node. Based on these trust weigh values, the destination node can verify, whether the received RREQ packet has come across through the trustworthy route or not. But the genetic algorithm increases the complexity of proposed system.

In this research work, the methods are proposed to overcome the drawbacks that are discussed in the above existing methods. The proposed methods do not use any extra control packets, so they can limits the control overhead. The trust computation methods are proposed without using static weights; hence they can support the dynamic network conditions. Unlike the existing protocols, the proposed methods have good route recovery capability and used the light weight soft computing methods in trust evaluation.