Chapter 3

Congestion-Aware Multipath Routing Protocol to Support QoS

3.1 Introduction

This Chapter aims to tackle the problem of congestion in multi-hop wireless ad-hoc network design from network protocol perspective. A dense MANET with big number of nodes, large volume of data within and high node mobility can result in congestion; thereby degrade the QoS of the transmitted data significantly. Congestion in any network causes high packet delay, severe fall in network throughput due to packet loss, degraded video quality, and session disruptions. Therefore, it is highly required to understand the behavior of heavily utilized and congested wireless MANETs along with the data transmitted over it and to provide the solution of the challenges.

We have designed a multi-path routing protocol to mitigate the effect of congestion from wireless ad-hoc networks under heavy traffic load. The proposed protocol considers the probabilistic congestion model at any intermediate node based on M/M/1/B queuing network. The performance of the proposed protocol is evaluated for various QoS parameters like end-to-end delay, jitter, and routing overheads. The results obtained in this work outperform from the existing multi-path path protocols. [81-85]
provide solutions for dealing congestion problem in ad-hoc networks. Many authors suggested bypass (alternate) routes from the congested node to the destination node whereas others focused on MAC utilization, data rates, queue length, and delay to the destination. These protocols are designed to solve the congestion problem at any one node on the path but do not take account of congestion at all intermediate nodes between a source and destination node. Hence decisions made to mitigate congestion are not correct. The proposed protocol considers congestion at all intermediate nodes between a source and destination node and then take decision of best path among all available paths.

Existing congestion aware protocols like CRP [36], E-CARP [37], CARM [38], and [80-83] face the problem of high overheads and low throughput because of poor modeling of congestion.

Fig. 3.1: An Example of MANET Based on M/M/1/B Queuing Network Model
In Section 3.2, we have proposed a probabilistic model for evaluating congestion level in the wireless ad-hoc networks considering randomness of packet arrival and forwarding rate in the network and proposed a multi path routing protocol which take account of congestion of all nodes and calculate the average congestion of the path. The advantage of the proposed routing protocol is that it chooses the best path which takes least waiting time to transmit a packet between a source and destination node. Averaging the congestion gives good results in the situation when one node in a path is more congested than all nodes in other path but average congestion of first path is lesser than the average congestion of the other path.

3.2 System Model

To model congestion in a multi hop MANET we consider M/M/1/B queuing network model. Thus, in a model for MANET, the mobile nodes are considered as service centers. If a mobile node is busy in transmitting a packet then newly arrived packets join the queue at that node buffer and wait for its turn to get transmitted on First Come First Serve (FCFS) basis.

![Diagram of packet arrival and forwarding rate at the network](image-url)

Fig. 3.2: Arrival and Forwarding Rate of Packets at $j^{th}$ Node in the Network
After completion of service at one node, the packet may move to another node for service or at destination the packet is absorbed.

We assume following assumptions in our modeling:

1. The mobile nodes are independent and identical.
2. The traffic generated by the nodes follows Compound Poison Process.
3. Congestion is considered to occur if the buffer at a node is full.
4. The wireless channels are noise free.
5. A packet is dropped at a node under one or more of the following conditions:
   - When a packet wants to enter the queue of a node and the buffer is full (congestion).
   - The time that the packet has been buffered exceeds the limit.
   - Link breaks due to mobility of nodes.

Fig. 3.1 describes the ad-hoc network topology where $N$ mobile nodes with finite buffer capacity $B$ are randomly scattered. Dashed circles represent the transmission range of a node. The nodes in the network are considered to have the same range. They are said to be neighbours if they fall in the transmission range. A packet generated from a source node $(S)$ reaches to a destination node $(D)$ through intermediate forwarding nodes 1, 2, 7, and 8. At node 2, the packets arrive from neighboring nodes 1, 3, 4 and 6. They are forwarded to nodes 5 and 7. A functional block diagram of $j^{th}$ node falling in the route is shown in Fig. 3.2. The packets are assumed to be forwarded from $k$ nodes to a neighbouring node, say, node $j$ directly connected with them. All the packets are buffered at node $j$ for onward transmission to separate destinations. Further, the node $j$ generates its own packets and keeps in the same buffer. The packets at $i^{th}$ node may face three situations – (a) the packets join the queue at node $j$ with probability, say, $P_{ij}$, or, (b) the packets are lost due to mobility having probability of loss as $(1 - \Sigma_{i=1}^{k} P_{ij})$, or, (c) joins the queue at other than node $j$. Since the output from $i^{th}$ node flows to
The $j^{th}$ node through one of the $k$ transmission paths, therefore, packet forwarding probability from $i^{th}$ node to $j^{th}$ node becomes:

$$P_{ij} = \begin{cases} \frac{1}{k} & \text{for } i \neq j \\ 0 & \text{for } i = j \end{cases} \quad (3.1)$$

Let $\lambda_g$ be the packet generation rate of the $j^{th}$ node and $\lambda_i$ be the total packet arrival rate from other neighboring node(s) to node $j$. Then number of packets forwarded ($\lambda_i$) to node ‘$j$’ from ‘$k$’ neighboring nodes in $\Delta t$ time can be given as $Q_i \cdot \sum_{i=1}^{k} P_{ij}$, where $Q_i$ is the number of packets in the buffer of $i^{th}$ neighboring node at any instant of time ‘$t$’ from which the packets are being forwarded to node ‘$j$’.

From Fig 3.2, the total mean packet arrival rate ($\lambda_j$) at node $j$ is given as

$$\lambda_j = \lambda_g + \lambda_i \quad \forall j \quad (3.2)$$

By definition of the single server queuing system, the probability of ‘$m$’ packets being in the queue at node ‘$j$’ is given as

$$P_m = \frac{\left(1 - \frac{\lambda_j}{\mu_j}\right)^m}{\left[1 - \left(\frac{\lambda_j}{\mu_j}\right)^{B+1}\right]} \quad (3.3)$$

where $\mu_j$ is the total packet forwarding rate at node $j$. From Eq. (3.3), the average number of packets in the queue at $j^{th}$ node is given by

$$Q_j = \sum_{m=1}^{B} m P_m = \frac{\lambda_j \left[1+B\left(\frac{\lambda_j}{\mu_j}\right)^{B+1}-(B+1)\left(\frac{\lambda_j}{\mu_j}\right)^{B}\right]}{(\mu_j-\lambda_j)\left[1-\left(\frac{\lambda_j}{\mu_j}\right)^{B+1}\right]} \quad (3.4)$$

Therefore, the average time spent by a packet in the queue, i.e., waiting time at $j^{th}$ intermediate node

$$W_j = \frac{Q_j}{\lambda_j} \quad (3.5)$$
Let a packet passes through ‘s’ intermediate nodes to reach destination node, then by symmetry, the average queuing delays at all stations are the same. Therefore, the expected queuing delay that a packet waiting for transmission in the entire network from source to destination node is given by

\[ \bar{W} = \sum_{j=1}^{s} \bar{W}_j \]  

(Eq. 3.6)

Eqs. (3.2)-(3.6) are used to determine various network parameters, like, end-to-end-delay, congestion level at a node, and jitter lying between source and destination node.

**Calculation of End to End Delay**

End to end delay in wireless system is the sum of queuing, transmission (MAC delay), and propagation time. Let it is denoted by \( E_D \) and calculated using Eq. (3.7) as follows:

\[ E_D = \bar{W} + T_D + P_D \]  

Where \( T_D \) and \( P_D \) are the transmission and propagation delay of a packet, respectively.

**Congestion Level Calculation**

CA-AOMDV calculates congestion level for all paths available between source and destination nodes by taking average of congestion levels of all intermediate nodes available on the path. If the congestion level at any intermediate node in the path is 1 (buffer is full) then calculation of congestion level for other succeeding intermediate node(s) is aborted and the node at which it is aborted sends an \textit{RERR} packet to the source node. This process is repeated for all available paths between source and destination node.

(a) Congestion Level of a Node

Let \( CL^p_j \) be the congestion level at \( j^{th} \) node of \( p^{th} \) then \( CL^p_j \) can be given as:

\[ CL^p_j = \frac{\text{Number of packets in queue at } j^{th} \text{ node}}{\text{Buffer Size}} \]  

(Eq. 3.8)
\[ C_{ij}^P = \frac{Q_j}{B} \]  

Using Eq. (3.3), we obtain

\[ C_{ij}^P = \left( \frac{\lambda_j^{B+1} - (B+1) \lambda_j^B}{\mu_j^{B+1} - (B+1) \mu_j^B} \right) \]  

The value of \( C_{ij}^P \) lies between zero and one. For \( \overline{Q}_j = 0 \), i.e. for empty buffer it is zero and one when buffer is full i.e. \( \overline{Q}_j = B \). We have considered three classes of congestion level a path: (i) **low** (ii) **medium** and (iii) **high**. A node is low congested, if its congestion level \( C_{ij}^P \) is \( \leq 0.50 \), medium if \( 0.50 < C_{ij}^P \leq 0.75 \) and high for \( C_{ij}^P > 0.75 \).

(b) **Average Congestion level of a Path**

The average congestion level \( ACL_p^C \) for \( p^{th} \) path at any instant of time ‘\( t \)’ can be calculated as follows:

\[ ACL_p^C = \frac{\sum_{\text{congestion at all intermediate nodes}}}{\text{number of intermediate nodes}} \]  

\[ ACL_p^C = \frac{\sum_{j=1}^{s} C_{ij}^T}{s} \] (3.12)

Proposed method is based on finding out the least congested path among available multipath(s). The source node executes the proposed congestion adaptive routing algorithm to find out all available paths between itself and the destination node. The algorithm calculates the congestion level at each intermediate node and then obtains average congestion level of the path. These calculations are repeated for every discovered path. The average congestion levels are compared and the path with
minimum congestion level is chosen as primary path and rest of the paths are saved in the routing table for later use in increasing order of their congestion level. This method is beneficial for small size ad-hoc networks in the situation when intermediate nodes are highly congested.

### 3.3 Congestion Aware QoS Routing Protocol

Two Paths (S → 1 → 2 → 7 → 8 → R) and (S → 10 → 11 → 12 → R) with 4 and 3 hops, respectively are shown in Fig. 3.3.

![Diagram](image)

Fig. 3.3: An Example of Data Transmission in Congested MANET

Each node is equipped with a FCFS queue, which stores the data packets in order of their arrival and forwards them to the next node in the path. It may be observed from the Fig. 3.3 that Path1 with four hops is less congested compared to the path2 with three hops. The CA-AOMDV would select the less congested but longer path Path1 and is expected to give better QoS.
(a) Route Discovery Process

In Fig. 3.3, a source node ($S$) starts route discovery process by generating $RREQ$ packet, and initiating its flooding throughout the network. On receiving $RREQ$ packet, an intermediate node compares its destination sequence with the destination sequence number available in $RREQ$ packet.

Procedure 3.1: Route Discovery Algorithm of CA-AOMDV

1. Source node send $RREQ$ packets to its $n_nodes$ // Route Discovery Process
2. for p=1 to n do
3. while $n\_node \neq d\_node$ do
4. neighbors set a reverse path to the source
5. if a forward path to $d\_node$ exists then //neighbors check for forward path
6. If ($CL_j^p = 1$) then
7. send $RERR$ packet to the $s\_node$ // Abort the route discovery process
8. else calculate $\sum ACL_j^p = \sum ACL_j^p + CL_j^p$
9. send $RREP$ packet to the $s\_node$ through reverse path
10. end if
11. else forward copy of $RREQ$ packets to all its $n\_node$
12. Calculate $\sum CL_j^p = \sum CL_j^p + CL_j^p$
13. end if
14. end while
15. Calculate Path congestion level ($ACL_j^p$) using Eq. (3.11)
16. Compare $ACL_j^p$ of all paths
17. Store primary route and secondary in routes in routing table
18. Start data transmission on current route
19. if current route breaks then
20. Send $RERR$ message
21. Select a new route with next higher congestion level from the routing table
22. Go to step 18
23. end if
If destination sequence number available in \textit{RREQ} packet is greater than destination sequence number of the intermediate node then intermediate node calculates its congestion level and add it to the value of congestion level in \textit{RREQ} packet and then forwards the \textit{RREQ} packet to its neighboring nodes, if there is no direct forward path from it to the receiving node (R). An intermediate node can receive multiple copies of \textit{RREQ} packet and are examined to form alternate reverse paths.

These reverse paths are formed only for those copies of \textit{RREQ} packets which follow loop freedom and disjoint path conditions. When an intermediate node finds a reverse path to source node, it checks for the one or more forward paths to the destination node. If forward path exists, it responds with a \textit{Route Reply Packet (RREP)} to the source node via reverse path. If an intermediate node does not have forward path, it further re-broadcasts the \textit{RREQ} packet until destination node arrive. Destination node responds with an \textit{RREP} to all copies of \textit{RREQ} packets through a loop-free disjoint reverse path to the source node. Every intermediate node on reverse path(s) uploads its congestion level on \textit{RREP}.

\textbf{(b) Primary Route Selection}

On receiving RREP(s) from different reverse paths, source node calculates average congestion level (\textit{ACL}) of the paths by using Eq. (3.11). Depending on the ACL value, a path can be in any of the following three states:

(i) highly congested, \[ \text{if } ACL^t_p > 0.75 \]
(ii) medium congested \[ \text{if } 0.50 \leq ACL^t_p \leq 0.75 \] and
(iii) low congested \[ \text{if } 0.0 \leq ACL^t_p < 0.50 \]

If discovered paths lie in different states of congestion then the source node selects a path with low congestion level as \textit{primary route} for data transmission. If these paths lie in the same state of \textit{ACL} then the difference between the average congestion levels of
the paths may not be significantly high. In such situation it is critical to decide a path to select as primary route and others as secondary route.

(c) Secondary Route Selection

Routes other than primary routes are assumed as secondary routes in CA-AOMDV. On failure of primary path, either due to mobility or buffer full ($CL^p_j = 1$) condition at an intermediate node ‘$j$’ of path ‘$p$’, the path with next higher congestion level is selected as current (secondary) path for continuing data transmission and this condition follows for all available routes between the source and the destination node.

From Fig. 3.3, we can understand the importance of calculating path congestion level. Congestion level of nodes in 1, 2, 7, and 8 in path (S-1-2-7-8-R) is 0.83, 0.33, 0.0 and 0.66 respectively where as congestion level of nodes 10, 11, 12 in path (S-10-11-12-R) is 0.66, 0.66 and 0.66 respectively Though congestion level node A is higher (0.83) in path (S-1-2-7-8-R) than any node in path (S-10-11-12-R), even then proposed protocol will select path (S-1-2-7-8-R) as primary path because average congestion level of the path (S-1-2-7-8-R) $\approx 0.45$ is lesser than the average congestion level of the path (S-10-11-12-R) i.e. 0.66. Since the number of packets (11) in queues at different node of path (S-1-2-7-8-R) are lesser than number of packets (12) in queues at intermediate nodes of path (S-10-11-12-R), total waiting time will also be lesser and hence improved E2E delay and jitter.

(d) Avoiding Loop Formation

Loop formation is very common in multi path routing protocols and produces stale routes. To avoid loop formation while processing multi paths, following two issues arises:

(i) Since each of these paths may have different congestion levels, which path has to publish to neighboring nodes among these nodes?
(ii) Among these published paths which one should be accepted by a neighboring node?

These problems are demonstrated using Fig. 3.4 and Fig. 3.5. In Fig. 3.4, node S is the source node and node R is the destination node. Node S has two paths from node S to node R: Path1 and Path2. Let node S, advertises path1 to node F and path2 to node G, each of them have path to node R through node S but with different congestion levels.

A source node in MANET cannot decide whether a neighboring node is an upstream node or downstream node to it. As in fig. 3.4, S cannot decide whether the nodes G, E, F are upstream nodes or downstream nodes for the path to the destination node R. These nodes can have a path to node R through node S with lesser congestion level than path1, then there is a possibility of loop formation. Fig. 3.5 shows another loop formation situation. Node A and node D have same congestion level to a destination node R. Let node D obtains another path to node R via node A with more congestion level. In this case, node D cannot decide whether node A is an upstream node or downstream node. A path with higher congestion level may cause loop in the path. To provide solution to this loop formation problem, we use highest sequence number as solution. Entries of new routes are made into routing table (Table 3.1).

- **For Highest Destination Sequence Number**: Routes are maintained for highest sequence number only. We can avoid a loop with a restriction that a node with multiple paths will have same destination sequence number.

- **For Same Destination Sequence Number**: A source node never
advertises a route having a lesser congestion level and the neighbor node never accepts a route having higher congestion level than advertised.

(e) Disjoint Paths

The major challenge in multiple paths routing protocols is to maintain disjoint loop-free paths. CA-AOMDV observes the available disjoint paths as: (i) link disjoint path and (ii) node disjoint path. Consider two paths P1 and P2 from a source node S to the destination node D. To show P1 and P2 are link disjoint (Fig. 3.6), contradiction logic is followed: Let these two paths have at least one link in common, say, link I-J. Using inimitable next hop criteria, there will exactly one path from node I to the destination node via node J. It shows that node I will advertise only one path to node D via J having information about corresponding last hops. More than one path may exist from node J to node D having different last hop. Hence, it implies that more than one path cannot exist from upstream nodes of node I sharing the link I-J through node I.

For node disjoint path, uniqueness of next hop and last hop is not sufficient and need an additional restriction. In Fig. 3.7, both the paths S-B-I-X-D and S-C-I-Y-D are not node disjoint. Node disjoint path has been obtained by applying following restriction: If the nodes which are common in a group of link disjoint paths allow other upstream nodes exactly one path through them, a node disjoint paths is obtained. This could be obtained because for one destination sequence number, each node advertises only one path to the neighboring nodes.

(f) Route Maintenance

On failure of current route, CA-AOMDV looks into the routing table for next low congested available route and sends the data via this new route. The lost packets due to link break are resend through this new alternate route. In case of failure of all routes, the node generates and forwards RERR packets towards destination node to restart route establishment process. In MANET, a route may not be active for longer time and for a
very short duration may lose the benefit of multipath routing. CA-AOMDV uses a moderate setting to timeout value of a route and uses HELLO packets to proactively remove the old routes.

CA-AOMDV uses source sequence number and destination sequence number for updating the information about latest route between source and destination node. Source and destination sequence numbers are time stamps which allow a node to compare how fresh their information on other node is. The structure used in the algorithm is shown in Fig. 3.8. Parameter \( \text{advertised\_con\_level} \) is used to advertise the maximum value of path congestion level to avoid loop formation. Route list contains the (next hop, last hop, hop count, path congestion level, time to live) information about each alternate path. An intermediate node ‘i’ compares its destination sequence number \( seq_{num}^d_i \) with the destination sequence number of \( RREP \) packet \( seq_{num}^d_f \). If \( (seq_{num}^d_i < seq_{num}^d_f) \) then node ‘i’ update the route list with latest sequence number and initialize corresponding advertized congestion level as follows:

\[
\text{advertised\_con\_level} = \max (\text{cong\_level}_1, \ldots, \ldots, \text{cong\_level}_n)
\]

**(g) Data Packet Forwarding**

A source node with real time data in CA-AOMDV, initiates with route establishment process, selects a route with minimum congestion level and forwards data packets to the destination node.
Table 3.1: Routing Table Structure of CA-AOMDV

<table>
<thead>
<tr>
<th>Destination</th>
<th>Sequence Number</th>
<th>Advertised Hop Count</th>
<th>Advertised Cong. Level</th>
<th>Route List</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Next Hop1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Last Hop1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Hop Count1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Cng_level1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TTL1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Next Hop2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Last Hop2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Hop Count2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Cng_level2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TTL2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>……</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>……</td>
</tr>
</tbody>
</table>

This process continues until all routes existing in routing are exhausted. When all routes are exhausted, source node restarts route discovery process.

```plaintext
if (j=d) then insert (j,i,1,0) into route_list^d;       // neighbor is the destination
else insert (j, last_hop^d_jk, advertised_hop_count^d_j + advertised_cng_level^d_j + ng_level_j ) into route_list^d;
endif

if ((seq_num^d_j = seq_num^d_i) and (advertised_cng_level^d_j > advertised_cng_level^d_i)) then
  if (j=d) then
    if ((next_hop^d_{ik1} = ) and (last_hop^d_{ik2} = )) then // uniqueness of next hop and last hop is checked for path k1 and k2 respectively.
      insert (j,i,1,0) into route_list^d;
    endif
  endif
endif
else
  if ( (next_hop^d_{ik1} = ) and (last_hop^d_{ik2} = ast_hop^d_{jk}) ) then
    insert (j, last_hop^d_{jk}, advertised_hop_count^d_j + advertised_cng_level^d_j + ng_level_j ) into route_list^d;  // uniqueness of next hop and last hop is established
  endif
endif
endif
endif
```

Fig. 3.8: Route updating process in CA-AOMDV, invoked by a node ‘i’ on receiving a route advertisement for a destination ‘d’ from a neighbor ‘j’.
On failure of current path or packet drop, CA-AOMDV switches to next available path with minimum congestion level available in routing table and continue data packet forwarding.

### 3.4 Simulation Results and Discussions

NS-2.35 simulator is used to test the performance of the network. Observations are taken for E2E Delay, Jitter and Routing Overhead for different congestion levels and node mobility. The nodes are free to move in all directions and may create link breaks at unknown intervals. Simulation parameters are given in Table 3.2.

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Network Components</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Mobility Model</td>
<td>Random Waypoint</td>
</tr>
<tr>
<td>2.</td>
<td>Number of Nodes</td>
<td>100</td>
</tr>
<tr>
<td>3.</td>
<td>Number of Connections</td>
<td>50</td>
</tr>
<tr>
<td>4.</td>
<td>Link Bandwidth</td>
<td>1 Mbps</td>
</tr>
<tr>
<td>5.</td>
<td>MAC Layer</td>
<td>MAC/802_11</td>
</tr>
<tr>
<td>6.</td>
<td>Cache Time</td>
<td>5 Sec.</td>
</tr>
<tr>
<td>7.</td>
<td>Simulation Area</td>
<td>1000m×1000m</td>
</tr>
<tr>
<td>8.</td>
<td>Node Mobility</td>
<td>1-15 m/s step 4</td>
</tr>
<tr>
<td>9.</td>
<td>Packet Rate</td>
<td>1-50 packets/s step 10</td>
</tr>
<tr>
<td>10.</td>
<td>Packet Size</td>
<td>512 Bytes</td>
</tr>
<tr>
<td>11.</td>
<td>Queue Buffer Size</td>
<td>50 packets</td>
</tr>
</tbody>
</table>

**Packet Rate vs. Congestion Levels**

Comparison of average path congestion discovered by AOMDV and CA-AONDV is given in Fig. 3.9. For packet rate lying between 1 packet/s and 10 packets/s, path congestion level, $ACL_p^L$ in AOMDV is observed ($0 \leq ACL_p^L < 0.50$), for packet rate $> 10$ packets/s and $\leq 15$ packets/s it lie between 0.50 and 0.75; for packet rate between 15 packets/s and 20 packets/s, $ACL_p^L$ ranges between 0.75 and $< 1.00$. $ACL_p^L$ reaches to 1.00 for packet rate $> 20$ packet/s in AOMDV. On the other hand, in CA-AOMDV,
these congestion levels are observed for higher packet rates that is < 37 packets/s, 38 packets/s- 43 packet/s, 43 packets/s -47 packets/s and > 47 packets/s, respectively. From this comparison, we can observe that even for high packet rates, congestion level is low in CA-AOMDV when compared with AOMDV which make it suitable for video packet transmission in MANET.

![Fig. 3.9: Path Congestion Levels in AOMDV and CA-AOMDV](image)

![Fig. 3.10: E2E Delay at different packet rates in AOMDV and CA-AOMDV](image)

**End-to-End Delay**

Fig. 3.10 shows the effect of packet rates on E2E delay at average speed of 5m/s of mobile nodes. Performance of AOMDV and CA-AOMDV is observed same on low packet (1-4 packet/s) rate but for high packet rates CA-AOMDV is proven better than AOMDV. For packet rates lying between 1 packet/s and 10 packets/s (0 \( \leq ACL_p \leq 0.50 \)), E2E delay of CA-AOMDV is less than 2s and reduced in the range \( \approx 1\%-24\% \) that of AOMDV. The reason that E2E delay in both the protocols is close at low packet rates is that low path congestion occurs due to low packet arrival rate (\( \lambda_j \)) at any intermediate node ‘\( j \)’ as compared with the packet forwarding rate (\( \mu_j \)). For packet rates ranging between 10 packets/s and 20 packets/s and path congestion level (0.50 \( \leq ACL_p < 0.75 \)), a rise of 55% (\( \leq 8.2 \) s) and 43% (\( \leq 3.6 \) s) more delay is observed in AOMDV and CA-AOMDV, respectively. E2E delay exceeds 10s in this packet rate range therefore AOMDV is not found suitable for video transmission beyond this packet rate range. A sharp rise, more than 100%, is observed in AOMDV protocol for packet rate greater
than 30 packets/s having ($ACL_p^t \geq 0.75$), in which there is no mechanism to mitigate congestion forcing retransmissions of affected packets. On the contrary the congestion aware nature of CA-AOMDV is observed to give much improved performance with E2E delays restricted to only 24% -32% of that at middle congestion level ($0.50 \leq ACL_p^t < 0.75$). When the paths are further congested by raising the packet rates to 50 packets/s congestion E2E delay has a rise of more than 52% in CA-AOMDV and go beyond 10s for packet rate > 48 packets/s. AOMDV has not been found suitable for video streaming beyond 20 packets/s whereas CA-AOMDV fails after 40 packet/s. Therefore, CA-AOMDV is more compatible for transmission of high voluminous video data in MANETs as compared AOMDV.

We conduct the rigorous result analysis to observe the effect of mobility on performance of the two protocols. Packet rate vs. E2E delay is drawn in Fig. 3.11 and Fig. 3.12 for AOMDV and CA-AOMDV, respectively at node mobility 1m/s, 5m/s, 10m/s and 15 m/s.

E2E delay follows exponential nature with packet rate. The slope of these curves is found to increase (rate) with mobility for both the protocols but the rise in slope is less in CA-AOMDV as compared to the other protocol. The high slope of AOMDV curves is caused due to more retransmissions of mobility affected drop packets on already congested transmission path where as in the congestion aware protocol CA-AOMDV, the slope is low for all mobility in consideration in spite of retransmission of mobility affected packets that to made in congestions which occur at higher packet rate. These retransmissions are comfortably adjusted due to inherently low congestion levels at the intermediate nodes in the path. The slope of each curve is increasing with respect to increased mobility therefore E2E delay is increased. The E2E delay reaches its limits of 10 s (Fig. 3.13) at low packet rate under high node mobility while transmitting sensitive data in AOMDV. CA-AOMDV protocol on the other hand provides high mobility with
packet rate (40 packets/s) within the delay limit 10 s. Its highest achievable packet rate (20 packets/s) is at mobility (10 m/s) within the aforesaid delay.

![Fig. 3.11: E2E Delay in AOMDV at Various Node Speed](image1)

![Fig. 3.12: E2E Delay in CA-AOMDV at Various Node Speeds](image2)

**Jitter**

The jitter is observed in both AOMDV and CA-AOMDV as shown in Fig. 3.13 at mobility of 5 m/sec. But the effect of jitter is more pronounced in AOMDV as compared to CA-AOMDV as the former routes the packets without considering path congestion; with the rise in packet rate, the AOMDV suffers more. The packet drop in AOMDV raises steeply at packet rates greater than 20 packets/s where the jitter has increased by more than 200% (29 ms to ≈ 90 ms). Whereas the CA-AOMDV gives much better jitter, its value is increased by 51% of the jitter value at the same packet rates in AOMDV due to lower path congestion (Fig. 3.13). Moreover, beyond 20 packets/s AOMDV produces the worst packet loss and jitter making high speed transmission impractical. On the other hand, the proposed protocol is able to provide much better jitter with low packet drop (lesser than 35%) for the packet rates ranging up to 40 packets/s, the protocol, therefore, becomes more suitable for high speed transmission. However, at packet rates greater than 40 packets/s, the performance is found to be unsatisfactory as jitter crosses the limit of 30 ms.

Mobility induces packet drops there by raising the congestion levels and other QoS
parameters in the two protocols are shown in Fig. 3.14 and Fig. 3.15 respectively at node mobility 1m/s, 5m/s, 10m/s and 15 m/s. The faster the movement of the nodes more is the number of link breaks and loss of packets. The trend of jitter graphs in AOMDV protocol (Fig. 3.14) shows deterioration at higher and higher speed. The observations made above on QoS parameter demonstrate the effectiveness of the proposed CA-AOMDV protocol.

Fig. 3.13: Comparison of jitters in AOMDV and CA-AOMDV

Fig. 3.14: Jitter in AOMDV at Different Node Speed vs Packet Rate

**Routing Overheads**

Due to additional field in route discovery packet to calculate path congestion level initial overheads in CA-AOMDV are more than AOMDV but with increasing congestion level it is recovered as compared to AOMDV. Initially routing overheads are ≈3% and ≈4% in AOMDV and CA-AOMDV, respectively due to selection of multiple paths and packet payload. For $ACL_p^L$ between 0.0 and 0.50, there is no packet drop (Fig. 3.17); therefore routing overheads are low. But as packet rate more than 20 packet/s, path congestion level reaches to 1 (Fig. 3.9) which results in high packet drop and retransmissions of these congestion induced packets add additional $RERR$, $RREQ$ and $RREP$ packets in the network. Therefore routing overheads are increased by a factor of 2 it reaches to ≈55% at 50 packets/s as most of the packets are dropped at various intermediate nodes due to buffer full condition. On the other hand, in CA-AOMDV routing overheads are increased at a low rate because of low packet drop and retransmission rate. Routing overheads are ≈3%, 6%, 8%, 14% and 16% at packet rate
10 packets/s, 20 packets/s, 30 packets/s, 40 packets/s and 50 packets/s, respectively which is very less as compared to AOMDV as shown in Fig. 3.16.

![Graph showing Jitter in CA-AOMDV at Different Node Speed and Packet Rate]

**Packet Loss**

To observe the visual effect of packet loss we transmit ‘foreman’ video using both the protocols and observe following results: The congestion level of the paths discovered by AOMDV and CA-AOMDV are 0.78 and 0.47, respectively. Frame number 27, 52 are shown in Fig. 3.18 when used AOMDV and CA-AOMDV. From these figures we can observe that distortion is low and video quality is good at the end user in case of CA-AOMDV as compared to AOMDV by virtue of CA-AOMDV.

![Graph showing Comparison of Routing Overheads in AOMDV and CA-AOMDV]

![Graph showing Packet Rate vs. Packet Loss]

Fig. 3.17: Packet Rate vs. Packet Loss
3.5 Conclusions

Multipath protocols based on minimum hop counts do not fulfill QoS requirements of real-time data transmission in MANET. In this chapter, we propose a congestion-aware multipath routing protocol (CA-AOMDV) for small-size MANET to transmit data under heavy load conditions. CA-AOMDV detects all paths between source and destination nodes which are less congested. In CA-AOMDV, E2E delay is improved by 20%-80%, jitter is reduced by 25%-47%, and even though initial routing overhead is increased by 10% but overall routing overhead is reduced by 40-70%. This work can be extended in future for simultaneous transmission of higher priority video packets (I
packets) on less congested path and low priority video (P and B) packets on higher congested path.