Thesis for the Degree of Master of Science

Link Failure and Congestion Aware Reliable Data Delivery in Mobile Ad Hoc Networks

Exam Roll: Curzon-502

Session: 2009-2010

Regi. No: HA-1103

Department of Computer Science and Engineering
University of Dhaka
Dhaka, Bangladesh

June, 2012

Thesis for the Degree of Master of Science

Link Failure and Congestion Aware Reliable Data Delivery in Mobile Ad Hoc Networks

By

Md. Manowarul Islam
Exam Roll: Curzon-502
Session: 2009-2010
Regi. No: HA-1103

Supervised By

Dr. Md. Abdur Razzaque
Associate Professor

Department of Computer Science and Engineering
University of Dhaka
Dhaka, Bangladesh

June, 2012

Declaration of Authorship

We, hereby, declare that the work presented in this thesis is the outcome of the investigation performed by us under the supervision of Dr. Md. Abdur Razzaque, Associate Professor, Department of Computer Science and Engineering, University of Dhaka, Dhaka, Bangladesh. We also declare that no part of this thesis and thereof has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

.....

(Dr. Md. Abdur Razzaque)

Supervisor

Signature

.....

(Md. Manowarul Islam)

Candidate

Abstract

Nodes in Mobile Ad Hoc Networks (MANETs) are self-organizing and self-configurable. The MANETs are very useful where it is difficult to establish fixed infrastructure and centrally controlled network. Potential application of MANETs are in battlefield, military communication, disaster recovery, search and rescue etc. But as there is no fixed infrastructure and topology changes frequently, routing of data packets in such network is more difficult and complex than the wired network. Because of dynamic topology, nodes can enter and leave the network any moment. Their limited transmission range and energy-constraint make the routing protocol much difficult. Link failure and congestion are two major problems in MANETs.

Link failure may occur due to the dynamic topology, node mobility and congestion may occur due to queue overflow, limited bandwidth, packets collision or medium or channel overloading. Link failures and packet drops due to congestion are two frequently occurring problems in Mobile Ad Hoc Networks (MANETs), degrading the network performance significantly. In this thesis, we propose a link failure and congestion aware reliable data delivery mechanism (LCRDD) that jointly exploits *local packet buffering* and *multi-level congestion detection and control* approaches for increasing the data delivery performance. In LCRDD, we define a *cross-layer interface* between the network layer and the transport layer at each node in the network. Each intermediate node can use this interface to buffer data packets in an effective and efficient manner during link failure or the congestion of the network.

On detection of link failure or congestive state, LCRDD routing nodes buffer the incoming packets at the transport layer queue and resume transmission when the route is repaired locally. Also, LCRDD's mutli-level congestion detection helps it to take the most appropriate actions proactively. The cross-layer packet buffering approach of LCRDD helps to reduce unnecessary retransmission of data packets. Its multi-level congestion control ensure the network's safety state so that the network can avoid the congestion in an efficient way. Thus, it offers increased reliability and throughput and decreased end-to-end packet delivery delay and routing overhead compared to state-of-the-art protocols. The performance evaluations are carried out in Network Simulator v-2.34. The results of our simulation state that, the LCRDD outperforms than the other protocols.

Acknowledgment

I wish to acknowledge my gratefulness to my thesis supervisor Dr. Md. Abdur Razzaque, Associate Professor, Department of Computer Science and Engineering, University of Dhaka for his valuable suggestions, continuous help, long discussion, countless time and guidance from the beginning to the end of the project work. However, I believe, it is not possible to acknowledge properly the effort of my supervisor in written words.

I would like to thank specially our honorable teacher Shahed Anwar, Assistant Professor, Department of Computer Science and Engineering, University of Dhaka for his advice, suggestions and support. I would like to thank the Department of Computer Science and Engineering, University of Dhaka for giving me the opportunity to carry out my research here and providing me with the necessary resources and materials.

Finally, I would like to thank my parent, teachers and senior brother for their constant support and encouragement, which was an inspiration for me.

Last but not the least, I thank the almighty Allah for giving me the strength to complete this thesis work.

Md. Manowarul Islam

June, 2012

Table of Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

In this chapter, we overview the basic concept and problems of Mobile Ad Hoc Networks(MANETs). We discuss the current challenges of Mobile Ad Hoc Networks. Then we describe the motivation, contributions of our thesis work along with the goals.

1.1 Introduction

A Mobile Ad Hoc Network (MANET) [1] is a non-infrastructure mobile network having wirelessly connected mobile nodes. It is a set of wireless devices called wireless nodes or mobile nodes. Each of these mobile node be laptops, mobile phone, PDAs which can move frequently [2] [3] [4]. Nodes in MANET communicate with each other via wireless medium. There is no centralized controller and infrastructure in Mobile Ad Hoc Network. Because of its dynamic network topology [5], links between the nodes may be broken frequently. Another serious problem in MANETs is the packet drops due to congestion in the network. As a result, achieving reliable and timely data delivery is a challenging problem.

The nodes in MANETs are independent of each other and can move frequently causing routes to break. For this reason, the lifetime of a established route is very short and hence the data delivery efficiency decreases [2]. Most of the on-demand routing protocols operate in two phases; (a) route discovery phase and (b) route maintenance phase [1]. A source node initiates a route discovery process on demand whenever it needs to transmit

data. If a route is found, the source node starts transmitting data packets. On the occurrence of a route failure, the source node initiates a fresh route discovery process, which might take a considerable amount of time and thus the data delivery performance of the network sharply decreases. Figure 1.1 shows a mobile ad hoc network where the nodes are wirelessly connected with each other.

Figure 1.1: A mobile ad hoc network

1.1.1 Characteristics of Mobile Ad Hoc Networks

A node in a MANET network acts as a source or as a destination or as a forwarder or intermediate node [6]. Because of dynamic network topology, nodes in a MANET network move frequently and randomly. This gives some special characteristics to mobile ad hoc network. Some of them are as follows:

i. **Symmetric environment:** Nodes in MANET are independent of each other and communicate with other nodes without any fixed infrastructure. So MANET is fully symmetric network.

ii. **Node mobility:** Every node in MANET is independent of each other and can act

both as a host or as a router. As there is no centrally controlled fixed infrastructure, mobile nodes in MANET enter the network freely any time also can leave the network frequently [3].

iii. **Dynamic topologies:** The nodes in MANET are mobile and they are self-organizing. Because of node mobility, the topology of the network changes a lot over the time [3] [4]. Topology changes in MANET breakdown of a node due to loss of energy [6].

iv. **Link failure:** Because of node mobility, links between the nodes break in moment of time [7] [8]. Thus MANET suffers from link failure that causes the packets dropping in the network.

v. **Security:** Since mobile nodes can enter and exist the network any moment from the network, security is an important issues for MANET network. eavesdropping, spoofing can be occurred in a ad hoc network more easily than a wired network.

1.1.2 Challenges of Mobile Ad Hoc Networks

Because of dynamic network topology MANETs networks suffers from some problems. Some of these are given below:

i. **Distributed network:** Mobile Ad Hoc Network is a distributed network. Each of the mobile node in a MANET network is free to move anywhere which causes to change topology rapidly.

ii. **Transmission range:** The links in MANET network have lower capacity than their hardwired counterparts [9]. The transmission range of any mobile node in ad hoc network is lower than a wired network since nodes share the wireless medium. Effects of multiple node accesses, noise, and interference of the mobile nodes affect

the transmission range significantly. As a result data delivery rate of MANET is lower than wired network.

iii. **Bandwidth and Energy-constrained:** Nodes of a MANET network run on battery other than exhaustible means for their energy [10] [11]. Energy of such battery continuously reduce due to transmission, reception and overhearing of data packets [9] [12].

iv. **Packets loss frequently:** Nodes in MANET can move frequently. Packet dropping rate increases due to link failures, node mobilities and congestion. Because of limited bandwidth and transmission range packet dropping rate is much higher in this type network.

1.1.3 Advantages of Mobile Ad Hoc Networks

Mobile Ad Hoc Network has no centrally controlled fixed infrastructure [13]. The advantages of a Mobile Ad Hoc Network are as follows:

- Nodes in Mobile Ad Hoc Network are wirelessly connected. The deployment process does not need any cabling like wired network.
- Since there is no fixed infrastructure and central controller for the nodes and it can be deployed on the fly, it is not expensive or costly as wires or data cables are not required.
- MANET offer more flexibility and easy adaptability to change in the network configuration.
- Mobile users can access to real-time information.
- Network can be set up or extended to any places which can not be wired like war area or disastrous area etc.

1.1.4 Disadvantages of Mobile Ad Hoc Networks

Some of the disadvantages of MANET network are as follow:

- Link failure and congestion cause the low throughput.
- Low Bandwidth and energy-constrained hampered the network efficiency [14].
- Security issue is a major problem for MANET network.

1.2 Link Failure and Congestion in Mobile Ad hoc Networks

A Mobile Ad Hoc Network consists of a collection of mobile nodes forming a dynamic autonomous network [15]. Nodes communicate with each other over the wireless medium without the intervention of centralized access points or base stations. Hence, they form a fully mobile infrastructure.

1.2.1 Effects of Link Failure and Congestion

Link failure and Congestion lead to low throughput and thus reduce the overall efficiency of Mobile Ad Hoc Network [16] [17] [18]. These two vital issues lead to the following problems of the network:

i. **Long delay:** Frequently link failure and congestion leads to large amount of packets to be dropped. The source nodes need to retransmit a lot of data packets. As a result end-to-end packet delivery delay increases. Again for a congested network, data packets may collide and thus increase the end-to-end packet delivery delay and decrease the network performance.

ii. **High overhead:** Excessive route discovery latency of MANET increases the routing overhead. Also during the congestion, nodes in MANET network cannot deliver

their data packets properly and need to retransmit a lot of data packets which increases the routing overhead.

iii. **Packets losses:** Link failure of nodes causes to packet drop in the network [7]. Nodes in the network cannot exchange data packets properly due to the congestion of the network which in tern increases the packets dropping rate or packet loss rate.

iv. **Low throughput:** Longer delay and high packet loss rate in MANET reduce the overall throughput of the network and thus affect the efficiency of the network.

1.2.2 Causes of Link Failure and Congestion in Mobile Ad Hoc Networks

The nodes in MANETs are independent from each other and can move frequently causing routes to break. For this reason, the lifetime of a established route is very short and hence the data delivery efficiency decreases. Because of dynamic topology, link between the two nodes may break. Link failure may be occurred due to highly reduces received signal or for congestion or low capacity of the shared medium. Link failure leads to low throughput of the network.

Most of the on-demand routing protocols operate in two phases; (a) route discovery phase and (b) route maintenance phase [1]. A source node initiates a route discovery process on demand whenever it needs to transmit data. If a route is found, the source node starts transmitting data packets. On the occurrence of a route failure, the source node initiates a fresh route discovery process, which might take a considerable amount of time and thus the data delivery performance of the network sharply decreases [19].

Another important problem in MANETs is the packets dropping due to congestion in the network. Congestion may occur due to link failure, queue overhead or channel or media overloading [20] . The congestion leads to packet losses, throughput degradation of networks, wastage of time and energy for congestion recovery. Consider the Fig. 1.2, there

are three routes namely $S \to A \to B \to C \to D$, $X \to B \to C \to D$, $E \to B \to C \to D$. All the sources S, X and E send data via the intermediate node B. The node B receives data packets and forwards them to destination node D. When only one route is active (indicated by solid line), B can deliver data packets properly as queue occupancy is normal.

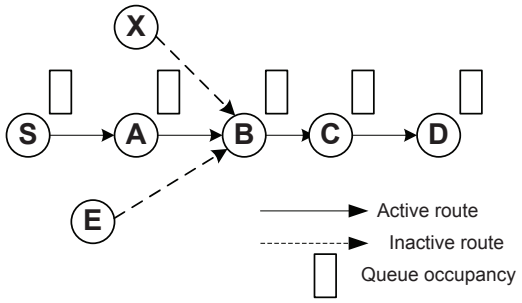


Figure 1.2: Queue occupancy is normal

But in Fig. 1.3, when all the three routes are active, queue occupancy at node B increases rapidly. It may not able to deliver all the received data packets if its incoming data packet rate becomes higher than the outgoing rate. As a result, packets may drop since queue overflow occurs at node B. For an active route, these dropped packets may traverse a longer path. So the network may suffer from high overhead, long delay and high packets loss [16] [17]. Thus throughput of MANETs decreases for network congestion.

Again, since there is no fixed infrastructure, all the nodes in MANET share the transmission channel; many nodes may contend for the channel simultaneously to transmit data packets, increasing packet collisions in the network. In such a situation, the congestion collapse may be occurred when no node will be able to transmit their data packets [17].

Consider the Fig. 1.4 when one route is active (indicated by solid arrow), node B forwards data packets of S since the channel loading is normal. But if the other nodes X and E try to send data packets simultaneously, then the channel loading rises to a

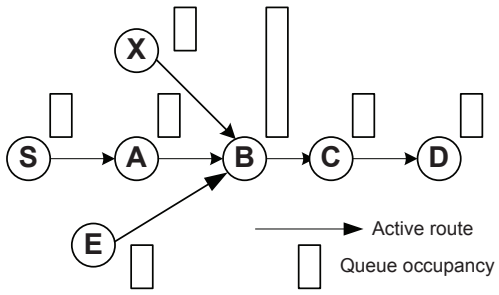


Figure 1.3: Congestion occurs in the network due to queue overflow

high level at node B. So B may not be able to deliver data packets properly. In Fig. 1.5,

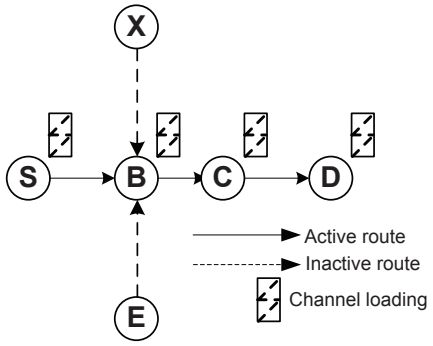


Figure 1.4: Channel loading in the network is normal

node B forwards data packets from three source nodes S, X and E and thus the channel loading in the network raises to a high level. Therefore, the node will not be able transmit data packets successfully. As a result, queue overflow occurs at the node B and packets start dropping.

In the literature, we find Spilt Multi-path Routing (SMR) [21], which uses multiple routes to split traffic and mitigate congestion; nodes in congestion adaptive Routing Protocol (CRP) [22] use bypass routes to mitigate congestion, etc. But, the problem is that multiple route maintenance overhead affects the network throughput.

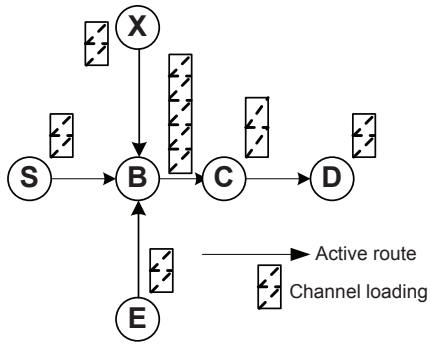


Figure 1.5: Congestion occurs due to channel overloading

A lot of research works focused on the problem of congestion control of MANETs in past [16] [23]. A huge analysis and modeling on routing protocols have been taken to overcome the congestion. We can categorize them as congestion-adaptive [24] routing or congestion aware routing protocols. In congestion adaptive routing protocol, the rate of data exchange between the nodes is adaptively changed based on the status of the network. The problem of this methodology is that, congestion may occur and later it handle using congestion control mechanism. To avoid congestion, we propose a congestion aware routing mechanism where nodes can handle congestion by analyzing the current status of the network.

Link failure due to node mobility is another important problem in Mobile Ad Hoc Network. It leads to low throughput, high end-to-end delay and routing overhead [2] [7].

In Fig. 1.6, three source nodes S, X and E are sending data to destination node D. Suppose, link between nodes B and C breaks. Node B drops all the packets for destination node D. All the intermediate nodes A, Y and F will simply drop all the data packets for D. As a result, packet losing rate of the network increases. As a result, the source node S needs to find a new route and retransmit all the data packets to desti-

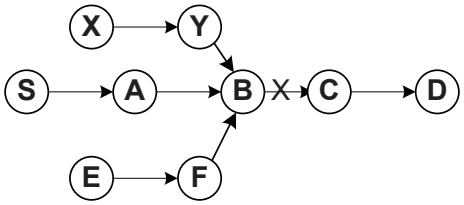


Figure 1.6: Link failure in mobile ad hoc network

nation. Since the network topology changes frequently, the source needs to retransmit huge amount of packets when link failure occurs. As a result, delay and overhead of the network increases and thus decreases the overall performance of MANETs.

There have been many research work to solve this problem. For example, Multiple paths to the destination may be used to handle link failure [25] [36] [39]. When link failure occurs, alternative path can be used. But as compared to single path multiple overhead may arise due to concurrent maintenance of multiple paths [26].

The nodes in MANETs are independent of each other and can move frequently causing routes to break. For this reason, the lifetime of a established route is very short and hence the data delivery efficiency decreases. In Local Reapair AODV [27] based on Link Prediction, LRAODV_LP [28], if a node detects that the signal strength goes below a predefined threshold, it initiates a fresh route discovery rather than sending error message backward. However, packets might be dropped at the intermediate nodes if the local route discovery takes longer period of time.

1.3 Dissertation Problem and Solution Methodology

The problems addressed above and their solution methodologies have been briefly introduced in this section.

1.3.1 Motivation

The two major problems link failure and congestion in Mobile Ad Hoc Network lead to low throughput, higher end-to-end delay and large routing overhead. To cope up with both the link failure and congestion, we propose a link failure and congestion aware reliable data delivery mechanism LCRDD that introduces *packet buffering* concept during link failure and local route discovery. The nodes on an active route, buffer their incoming data packets at their local transport layer queues (TQs) and, on finding a new path, resume their transmission. As a result, packet dropping rate of the network decreases. In addition to that, we employ a multi-level congestion detection and control mechanism at the source and intermediate nodes that can judiciously take the most appropriate decision for congestion control in the network proactively. Figure 1.7 shows the overall overview of LCRDD.

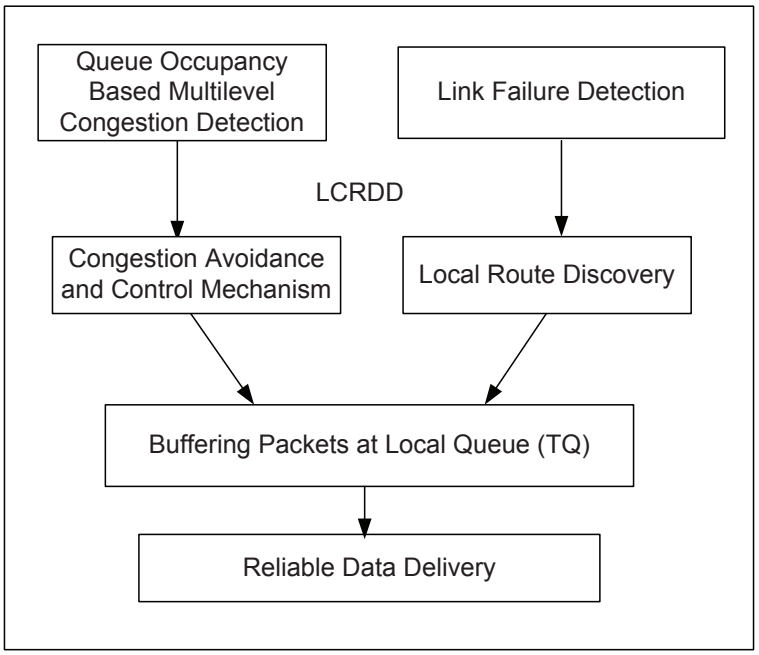


Figure 1.7: Overview of developed LCRDD mechanism

1.3.2 Design Goals

The main goals of our propose LCRDD mechanism are given bellow:

- **Buffering capability:** Nodes in LCRDD, use a mechanism for packet buffing with the help of a cross-layer interface. Nodes can buffer their packets during the link failure and congestion of the network.
- **Congestion avoidance:** LCRDD uses multiple level of congestion detection mechanism by which nodes in the network can efficiently detect the current congestion status of the network. Nodes can easily take appropriate actions based on the current status before the network enter into congestion. Thus LCRDD avoids the congestion.
- **Reduce unnecessary retransmission:** In Mobile Ad Hoc Network, nodes have to retransmit a lot of data packets due to link failure and congestion. To avoid unnecessary retransmission of data packets, LCRDD enhances the buffering capabilities of each node in the network.
- **Reliable data delivery and high throughput:** Using LCRDD mechanism, nodes in the network can handle both congestion and link failures more effectively. Thus packet delivery ratio increases over the time as well as the packet delivery delay is decreases. It ensures end-to-end reliable and efficient data delivery and improves network throughput significantly.

1.4 Thesis Contributions

In this thesis work, we develop a link failure and congestion aware reliable data delivery mechanism to address the link failure and congestion of Mobile Ad Hoc Networks. Our proposed LCRDD mechanism can handle route failure and congestion of Mobile Ad Hoc Networks in an efficient and effective manner. We introduce a cross-layer interface for

buffering the data packets during route failure. To avoid unnecessary retransmission, nodes in our network can buffer their incoming data packets during link failure and congestion of the network. LCRDD also capable of avoiding the congestion of the network using multilevel congestion detection mechanism. The main contributions of this work are summarized below:

- LCRDD provides an efficient buffering mechanism to buffer the packets when link failure occurs.
- LCRDD achieves an efficient congestion control mechanism, where nodes can detect multiple congestion levels of the network and can take proper control actions to reduce the packet dropping. Thus it avoids the congestion.
- Nodes in the network can handle both congestion and link failures more effectively and thus it ensures end-to-end reliable and efficient data delivery.

1.5 Thesis Layout

Rest of the chapters are organized as follows: Chapter 2 contains the background study about the thesis. In Chapter 3, we discuss about network model and detail description of our propose LCRDD mechanism. Chapter 4 contains the simulation and performance analysis of our propose LCRDD mechanism. Finally, Chapter 5 concludes the thesis, along with limitations and future works. Important references are included in Bibliography section.

Chapter 2

Background Study

In this chapter, we describe the detail description of routing protocols of Mobile Ad Hoc Networks. We try to state the protocols or research works, those are capable of handling link failure or congestion of the Mobile Ad Hoc Networks. We summarize our discussion by comprising the working principles of existing protocols those are related to our work.

2.1 Introduction

Routing means the act of moving or exchanging data packets or information from one node to another in an network. Two activities are mainly involved to the routing concept: firstly, finding and selecting an optimal route from available routing routes and secondly, delivering the information or data groups (called packets) properly through the optimal route. Since Mobile Ad Hoc Networks are self-organizing, network topology changes dynamically due to the mobility of the nodes.

2.2 Properties of Ad Hoc Routing Protocols

There are some properties that are desirable in MANETs:

- **Distributed operation:** The protocol should be distributed in a sense that the nodes in a MANET network should not be dependent on a centralized controlling node. Nodes in an ad-hoc network can enter or leave the network any time in an easily manner.

- **Loop free:** To enhance overall performance, the routing protocol should assurance that the routes are loop free in the network [29]. This helps to avoids the misuse of bandwidth or channel capacity or CPU consumption.

- **Demand based operation:** To minimize the routing overhead in the network and to use the network resources efficiently, the protocol should be reactive [33]. This ensures that, the routing protocol should react only when needed.

- **Unidirectional link support:** The radio environment can cause the formation of unidirectional links. Utilization of these links and not only the bi-directional links improves the routing protocol performance.

- **Security:** The environment of a MANET network is vulnerable to impersonation attacks. To ensure the wanted behavior, we need some sort of security measures such as authentication and encryption.

- **Power conservation:** The nodes in the ad hoc network can be laptops and PDA's that are run on battery power and therefore uses some standby mode to save the power [11]. The routing protocol should have the support for these sleep modes which is very important.

- **Multiple routes:** To reduce impacts of topological changes and congestion multiple routes can be used in MANETs [26]. If primary route becomes invalid, another stored route could still be valid and thus saving the routing protocol from high latency of initiating another route discovery process [19].

- **Quality of service support:** Some sort of quality of service is necessary to incorporate into the routing protocol [30][31]. This helps to find what these networks will be used for. It could be for instance real time traffic support.

2.3 Classification of Ad Hoc Routing Protocols

Routing protocol in MANETs can be classified by many ways, but most the classification is based routing strategy and structure of the network [32][33]. Routing protocols [15] in ad hoc mobile network can generally be divided into three groups:

- **Table Driven Routing Protocols**
- **On-demand Routing Protocols**
- **Hybrid Routing Protocols**

Figure 2.1 shows the classification of routing protocols in MANETs.

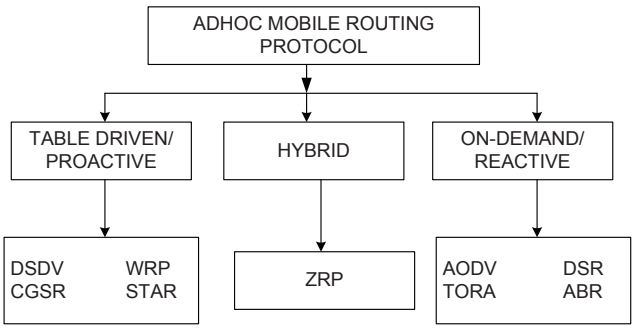


Figure 2.1: Classification of routing protocols in MANET

2.3.1 Table Driven Routing Protocols

These protocols are also called as proactive protocols [33]. Each node in the network maintains all of the routing information to other nodes through routing table. These routing table information are updated in a timely manner and periodically based on network current topology. When any node wants to send information to other nodes in the network, it does not need to discover the route. It just take the routing information from routing table and starts data delivery process using that information. This type

of protocols are not suitable for large network. In this type of protocol, the nodes have their own routing table and it becomes very large for large networks which increases the routing overhead as well [34][35]. Examples of table driven routing protocols such as Destination Sequenced Distance Vector DSDV and Wireless Routing Protocol WRP, are commonly known proactive routing protocol.

2.3.2 On-demand Routing Protocols

These protocols are also called reactive protocols or on-demand routing protocols. In this type of protocols, the nodes does not need to maintain routing information or routing activity. In this type of protocol, a node searches for new route in an on-demand manner and establishes the connection with other nodes to transmit and receive the packets. The route discovery occurs by broadcasting the route request packets throughout the network. Example: Ad-hoc On-demand Distance-Vector AODV, Dynamic Source Routing - DSR.

2.3.3 Hybrid Routing Protocols

These protocols combine the features of the proactive and reactive routing protocols. Nodes belonging to a particular geographical region or within a certain distance from a concerned node are said to be in the routing zone and use table driven routing protocol. Communication between nodes in different zones will rely on the on-demand or source-initiated protocols. Example: Zone Routing Protocol ZRP.

2.4 Ad-hoc On-demand Distance Vector (AODV) Routing Protocol

Ad-hoc On-demand Distance Vector AODV [27] is on-demand routing protocol which is a very simple, efficient, and effective routing protocol for Mobile Ad Hoc Networks. It is a reactive routing protocol works on a network, that does not have fixed topology

and centrally controlled infrastructure. To find any route or path, the AODV routing protocol uses a reactive approach. Each mobile host in the network acts as a router and as a host making the network self-starting.

2.4.1 Characteristics of AODV

AODV is an on-demand routing protocol which provides some important characteristics as follows:

- It provides unicast, broadcast and Multi-cast communication.
- It has relatively small delay for on-demand route establishment.
- Link breakages of the network in active routes can repair efficiently.
- All routes are loop-free are loop free since the nodes use sequence numbers.
- Keeping sequence numbers provides accuracy of information.
- Nodes in the network only keeps track of next hop rather the entire route.

2.4.2 Route Table Management in AODV

In AODV, each mobile node keeps an route entry for each destination in its routing table. Each node maintains following information in routing table:

- **Destination Address -** IP address of destination node.
- **Next hop -** Next node which is used to forward data packets towards the destination node.
- **Hop count -** number of nodes to the destination node.
- **Destination sequence number -** Sequence number for the destination node.

- **Active neighbors for this route -** Nodes which are neighbor in the route entry.
- **Life time -** Time duration in which the route remain valid or active.

2.4.3 Description of AODV

Ad-hoc On-demand Distance Vector is a simple and efficient on-demand routing protocol. As AODV is an on-demand routing protocol for MANETs, it has mainly two phases:

- **Route Discovery**
- **Route Maintenance**

2.4.3.1 Route Discovery

In this phase, a node in MANET network creates a route for data delivery to other node on-demand basis. When a node wants to communicate with another, it creates a control message called Route Request message(RREQ) and broadcasts this route request (RREQ) packets to its neighborhood nodes. This RREQ message may contain the following fields:

- Source address
- Source sequence
- Destination address
- Destination sequence
- Hop count

When any intermediate node get this RREQ message it performs two actions depends on its status. It can either forwards the RREQ packet or prepare a Route Reply (RREP) packet if it is the destination node. But if it is not the destination node, then it looks into

its local routing table to see if there is any path to the destination is already exists. If there is no route is available, it also forwards the RREQ to its neighbors; or a comparison is made between the destination sequence number in the RREQ packet and the destination sequence in its existing route cache. When the destination node gets the RREQ message it generate a Route Reply message RREP and sends it backward to the source node.

While transmitting a RREQ packet, every intermediate node enters the previous node's address. A timer is also maintained by the node in an attempt to delete a RREQ packet in case the reply has not been received before it expires. Thus,as RREQ travels from node to node in the network, it automatically sets up the reverse path from all these nodes back to the source.

When an intermediate node receives a RREP packet, it stores the information of the previous node for forwarding packets for the destination. The node plays a role of a "forward pointer". By this way, each node contains only the next hop information rather than all the intermediate nodes on the route to the destination are stored.

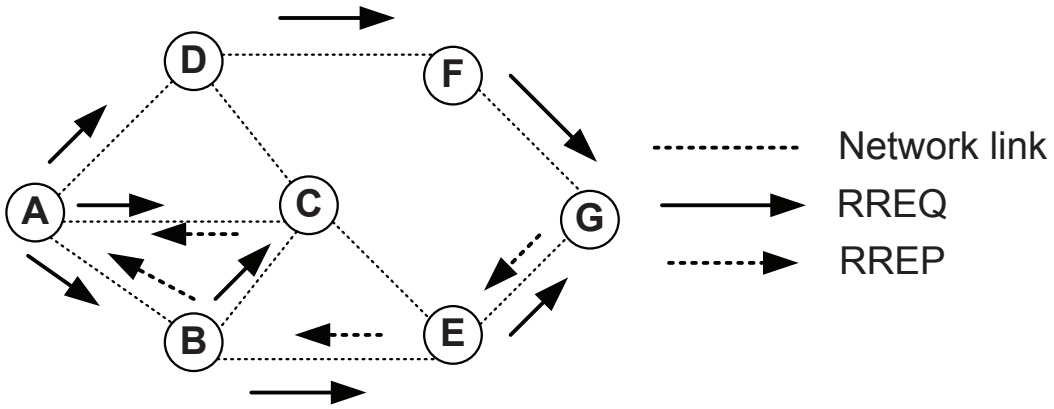


Figure 2.2: Route discovery process of AODV protocol

Figure 2.2 depicts route discovery process of AODV protocol. Suppose node A wants to send data packets to node G but it does not have a path to node G. So it initiates discovery process by flooding RREQ packets to all its neighboring nodes, namely B, C

and D. The solid arrows indicate the flow of RREQ messages.

When this RREQ packet reaches to nodes B, C and D, these nodes immediately search their route caches for an existing route to node G. Since there is no route to the node G, all the three nodes B, C and D forward RREQ message. When the destination node G gets an RREQ message, it generates a Route Reply message (RREP) and sends back to the source node. In this case, node G sends back RREP message after receiving the first RREQ message from node E. One possible route is $A \rightarrow B \rightarrow E \rightarrow G$ is then created. The intermediate nodes on the path from source to destination update their routing tables with the latest destination sequence in the RREP packet.

2.4.3.2 Route Maintenance

Route maintenance refers the keeping the route information up to date. When a node can detect that link to a neighbor is no longer valid (via link layer acknowledgments or HELLO messages)[45], it will inform the other nodes using that route by a control message called Route Error message (RERR), that the route is no longer exist for data transmission. The node then drops all the packets for that corresponding destination. All the intermediate nodes also drop their packets for the corresponding destination on receiving RERR message. When the source node gets this RERR message, it stops its transmission process and starts a fresh route discovery for data transmission.

This route maintenance process is illustrated in Fig. 2.3. Suppose the link between nodes B and F breaks on the path $A \rightarrow B \rightarrow E \rightarrow G$, RERR packets will be sent by node B to notify the source node A about the link failure.

2.4.3.3 Advantages of AODV

The main advantages of AODV routing protocol are as follows:

- **Simple -** AODV is simple on-demand routing protocol where each node maintain a simple routing table.

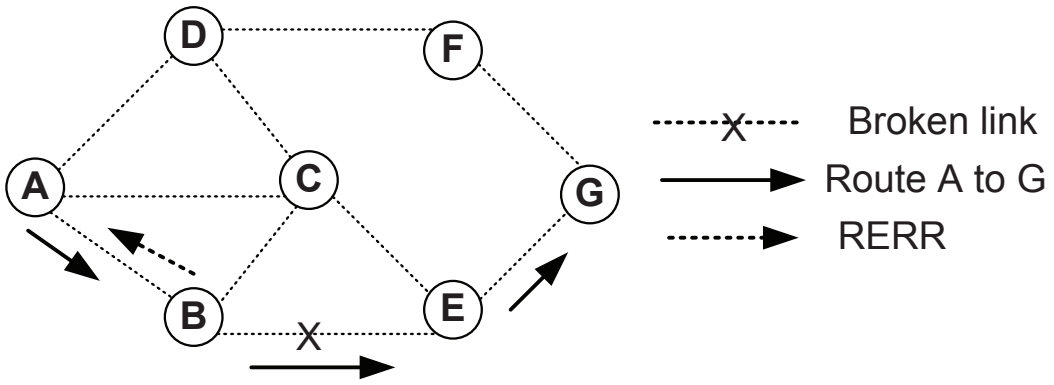


Figure 2.3: Link maintenance of AODV protocol

- **Loop-free -** The AODV protocol maintains loop free route.
- **Highly scalable -** AODV is highly scalable with compared to DSDV because of the minimum space complexity.
- **Less routing overhead -** AODV uses HELLO messages to routes maintenance which are range-limited, thus reduce unnecessary overhead in the network.

2.4.3.4 Disadvantages of AODV

- **High route discovery latency:** AODV does not discover a route until a flow is initiated. This route discovery latency result can be high in large-scale mesh networks.
- **Overhead on the bandwidth:** Periodic beaconing leads to unnecessary bandwidth consumption.
- **It is vulnerable to misuse:** The messages can be misused for insider attacks including route disruption, route invasion, node isolation, and resource consumption.

- **Routing overhead:** Multiple Route Reply packets in response to a single Route Request packet can lead to heavy control overhead.

2.5 Dynamic Source Routing Protocol (DSR)

Dynamic Source Routing protocol (DSR)[15] is another source initiating routing protocol in MANETs. Like AODV, when any node wants to send data to other nodes, it try to discover the path on-demand basis.

2.5.1 Description of DSR

Like AODV, as an on-demand routing protocol, DSR has mainly two phases:

- **Route Discovery**
- **Route Maintenance**

2.5.1.1 Route Discovery

In this phase, a node in MANET network creates a route for data delivery to other nodes on-demand. The source node creates a control message called Route Request message (RREQ) and broadcasts this route request (RREQ) packets to its neighborhood nodes. Figure 2.4 shows an example of the route discovery phase. When node A wants to communicate with node G, it broadcasts the request packets (RREQ) to its neighboring nodes B, C and D. However, node C receives duplicate RREQ packets from nodes B and D. It then drops both of them. The other nodes follow the same procedure. When the RREQ packet reaches at node G, it inserts its own address and reverses the route in the record and unicasts a Route Reply message (RREP) it back on the reversed path to source node A which is the originator of the RREQ. A route cache is maintained at every node so that, whenever a node receives a route request. Thus node A create a path $A \rightarrow B \rightarrow E \rightarrow G$.

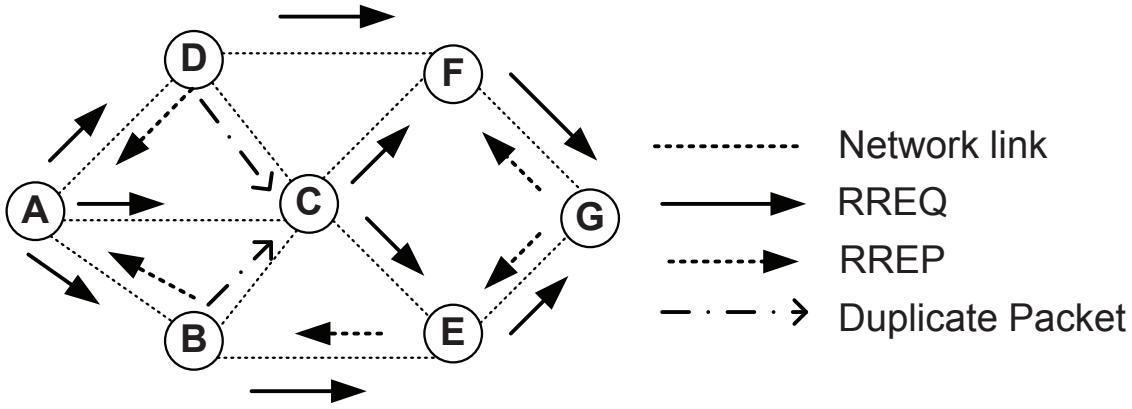


Figure 2.4: Route discovery process of DSR protocol

2.5.1.2 Route Maintenance

The route maintenance phase is needed to know whenever there is a broken link between two nodes in the network. As shown in Fig. 2.5, when the link between E and G is broken, a Route Error packet (RERR) is sent by E back to the originating node A. The source node A on receiving RERR message, re-initiates a fresh route discovery procedure to find a new route to the destination and also removes any route entries from it's cache to that destination node G.

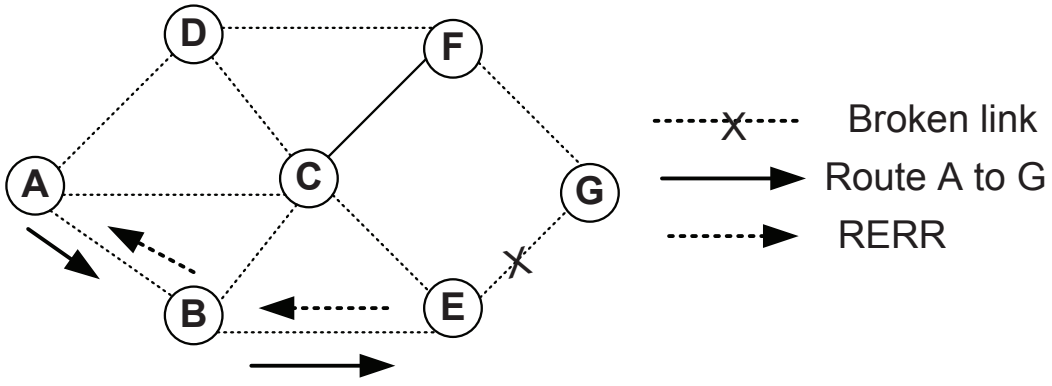


Figure 2.5: Link maintenance of DSR protocol

2.6 AODV-Based Backup Routing Scheme (AODV-BBS) in Mobile Ad Hoc Networks

AODV-Based Backup Routing Scheme(AODV-BBS)[36] is the modification of AODV routing protocol to maintain backup route to the destination node. In AODV-BBS each node uses 2-hop routing information to find backup routes. Figure 2.6 shows an example of AODV-BBS protocol.

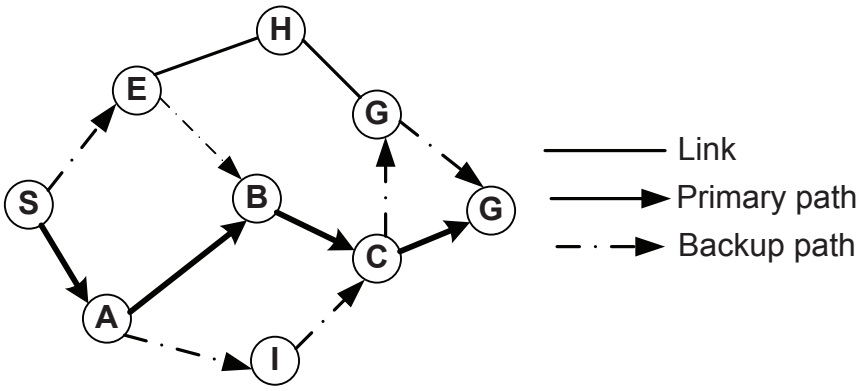


Figure 2.6: AODV-BBS routing scheme

2.6.1 Description of AODV-BBS

AODV-BBS has the following three phases:

- **Local Connectivity Management**
- **Path Discovery**
- **Path Maintenance**

2.6.1.1 Local Connectivity Management

In AODV-BBS, each node generate HELLO message to obtain 2-hop neighbor knowledge. Each node broadcasts HELLO messages containing list of all neighborhood nodes. Nodes that can learn about neighbors that are in two hop distance from its one hop neighbor's HELLO messages. Thus a node can keep the 2-hop neighbors knowledge.

2.6.1.2 Path Discovery

In AODV-BBS, both the primary and backup path are created during the route request phase. A node in the main path, can identify the border node (the node next to the downstream node) and reverse border node (the node next to upstream node). By using the 2-hop knowledge, each node in the main path can discover backup path with reverse border node. AODV modified the main RREQ message, RREP message and routing table by adding some additional fields, as shown in table 2.1.

Table 2.1: Additional fields for AODV-BBS

RREQ	**RREP**	**Routing Table**
Backup_count	Border	*Backup_count*
Reverse_border	–	Border

The route discovery process is same as the conventional routing protocol. However each intermediate node calculates the backup path using reverse_border field of the RREQ message and two hop routing information. Figure 2.7 shows that node C creates a back up path using node E.

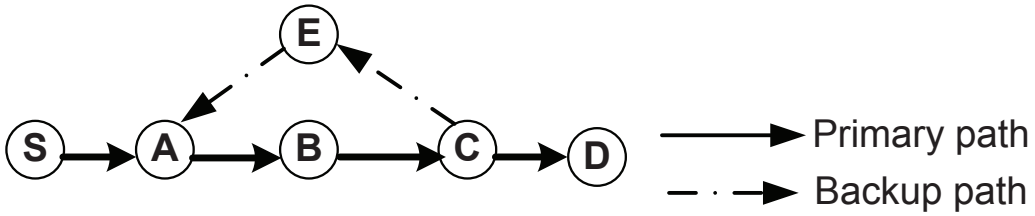


Figure 2.7: Backup path discovery in AODV-BBS

2.6.1.3 Path Maintenance

In AODV-BBS data packets are delivered through main path unless the primary route is disconnected. If an intermediate node detects a link failure, it uses the backup routes contained in its routing table that has 2-hop neighbor information. Node A in Fig. 2.8 uses the backup path using node E and thus handle the link failure.

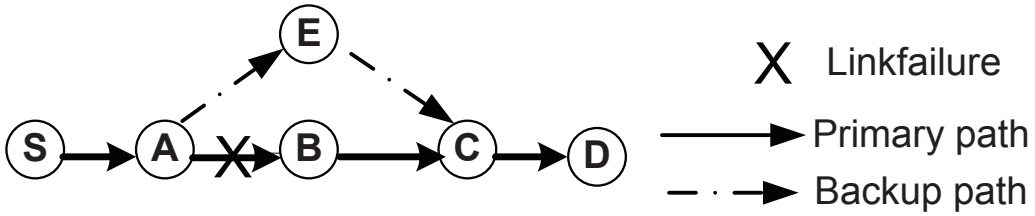


Figure 2.8: Link failure recovery in AODV-BBS

2.6.1.4 Disadvantages of AODV-BBS

- High overhead for maintaining the multiple routes in the network.
- Suffers from congestion.
- Packet Delivery Ratio reduce during congestion.
- Overall performance decreases during the congestion of the network.

2.7 Split Multipath Routing (SMR) with Maximally Disjoint Paths in Mobile Ad Hoc Networks

Several multipath routing for Mobile Ad Hoc Network has been proposed in the literature [16] [29] [37] [38]. Split Multipath Routing(SMR) [21] is an on-demand routing protocol that uses multiple paths that are maximally disjoints with one another. In SMR data traffic loads of the network is split over multiple paths so that the node can avoid congestion of the network.

2.7.1 Description of SMR

Like AODV routing protocol SMR has the following two phases:

- **Route Discovery**
- **Route Maintenance**

2.7.1.1 Route Discovery

In SMR a source node creates multiple duplicate paths having maximally disjoints node. Since the RREQ message floods in a network, several duplication of RREQ will traverse and reach the destination node. The destination node will select multiple paths by sending RREP message to the source node. This maximally disjoint paths are used to prevent the nodes in the network to congested and to make the network efficient.

To avoid every duplicate RREQ message, each intermediate nodes forward those packets, receive from different linkss. Also it consider the hop count that is not larger than that of the first RREQ message. Figure 2.9 shows an example, where source node creates multiple disjoints paths to the destination node.

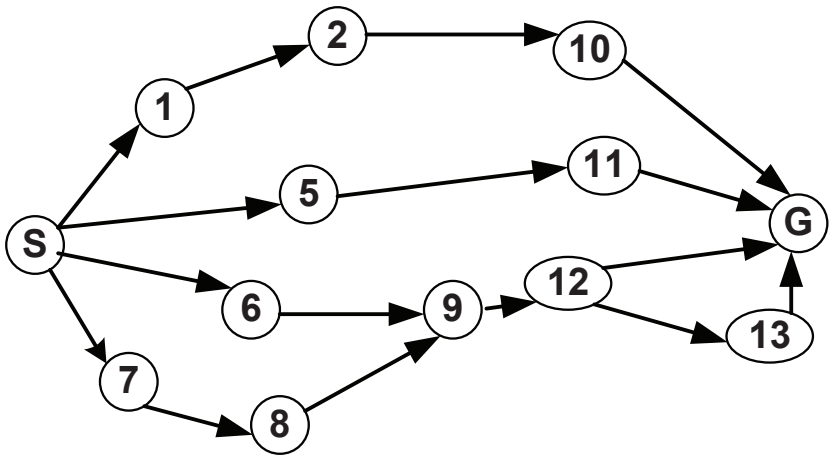


Figure 2.9: Multiple path in SMR

2.7.1.2 Route Maintenance

Whenever any link failure occurs in the network, the node create an RERR message and send back to the source. Upon receiving the RERR message, the source node removes all the information from its routing table of the broken link. Then it uses the alternative route to deliver the data packets.

2.7.1.3 Disadvantages of SMR

- High overhead for high latency of discovering multiple routes.
- Suffers from link failure, many packets has to retransmit.
- Packet Delivery Ratio reduce during link failures.
- Throughput of the network decreases rapidly due to link failure.

2.8 Implicit backup Routing-AODV (IBR-AODV) in Mobile Ad Hoc Networks

Implicit backup Routing-AODV(IBR-AODV) [43] takes the advantages of backup route through fast recovery of link failure in Mobile Ad Hoc Networks.

2.8.1 Basic Operation of IBR-AODV

In IBR-AODV, nodes generates many backup nodes and uses fast recovery process for handling the link failure. These backup nodes are created based on overhearing the transmitted data packets. If a node overhears the same data packets from a sequent three nodes of a route, it acts like a backup node. Backup node makes a backup routing table and store that packets. But if the node does not overhear any packets, it deletes the backup routing information.

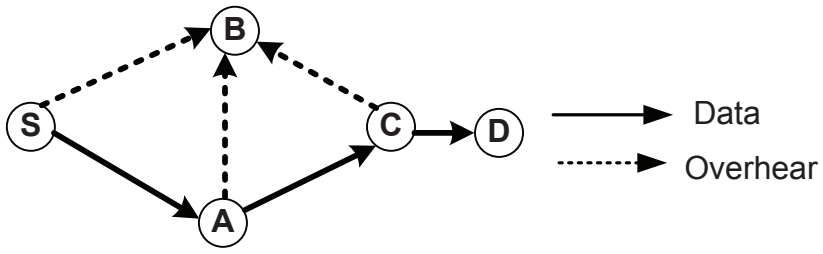


Figure 2.10: Backup node creation in IBR-AODV

Figure 2.10 shows the backup node creation process. The route between the source node S to the destination node D is $S \rightarrow A \rightarrow C \rightarrow D$. Node B is the backup node since it overhear the data packets from node S, A and C.

If their any link failure occurs in the network and the backup node hears about the link failure, it waits for a back-off period. Then the backup node sends a Route Change (RC) packet to the source node to handle link failure and waits for the acknowledgment

from the node that has a broken link. If the backup node gets any acknowledgment, it starts transmitting the packets that it has stored previously. Therefore data packets are delivered to the destination node by fast recovery process.

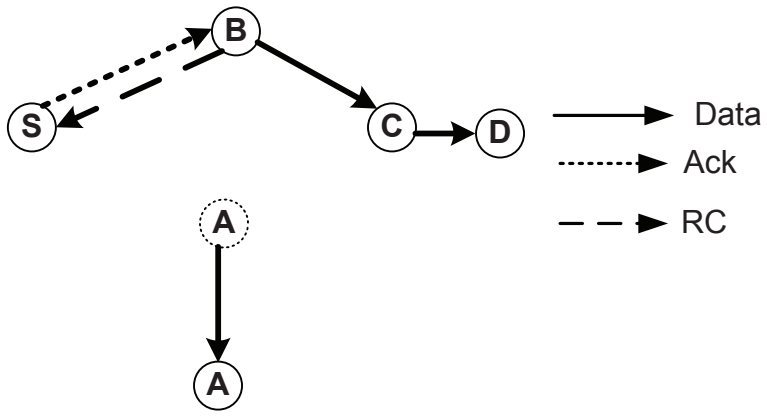


Figure 2.11: Basic Operations IBR-AODV

Figure 2.11 show an example where node A moves from active route. The backup node B starts a local recovery and create a new paths $S \rightarrow B \rightarrow C \rightarrow D$. After creating a new route data are delivered using this new routes. Thus IBR-AODV helps to fast recovery during the link failure in Mobile Ad Hoc Networks.

2.8.1.1 Disadvantages of IBR-AODV

- Multiple nodes can act as back-up node for the same packets and same destination nodes. Thus many nodes will try to transmit the same data packets unnecessarily.
- Network becomes idle due to multiple backup nodes for same packets.
- Packet Delivery Ratio reduce during congestion.
- Overall performance of the decreases as network become congested.

2.9 MANET Performance Enhancing with Packet Buffering using Two Hop Routing (THR)

In THR [44], nodes of a Mobile Ad Hoc Network can store packet temporarily using packet buffering in separate memory module. Each mobile node in THR, uses a separate memory as shown in Fig. 2.12. If the mobile nodes are in the visibility of two-hop distance, there will be minimum packet loss.

The intercommunication of nodes can be compared to a network with multiple processors that are interacting with memory module of multiple devices, where each memory model resides in the MANET devices. Each of the mobile node receives or delivers data packet only to or from one of its neighborhood node.

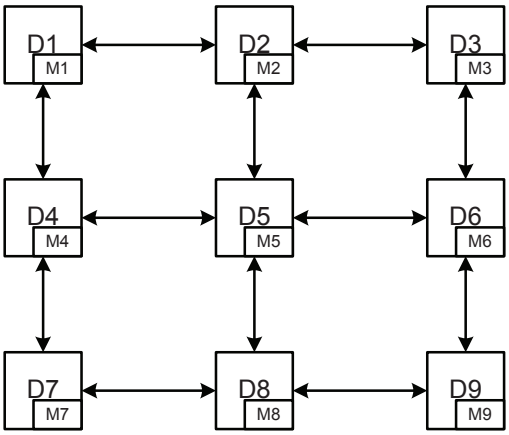


Figure 2.12: Backup node creation in IBR-AODV

The THR is a type of table driven routing protocol. Rather keeping the all information about each and every neighbor node as in proactive routing protocol, THR only keeps the evidence about the immediate neighbor and its neighbor node, i.e. information about the two hop neighbor. Therefore route discovery process takes less power and also the size of the routing table maintained at every node will also reduced.

When any intermediate node wants to discover any path, it first check its own routing table to find some nodes at two hop distance. If the information is available in its routing table only then data packets are transmitted to the nodes.

Node communicate with each other by creating routes between them. But if there is any link failure and node can detect the link failure, it then buffers temporarily its data packets to its separate memory module.

2.9.1 Disadvantages of THR

- Maintaining separate memory module important issue in MANETs.
- THR does not handle congestion.
- Packet Delivery Ratio reduce during the congestion of the network.
- Implementation of such separate memory module in mobile nodes is difficult.

2.10 The Link Failure and Congestion Management Routing Protocols in MANETs

Mobile nodes in a MANET network make a self-organizing and self-configurable network which has no central co-coordinator. Congestion is treated as vital causes of packet loss in wired network. However, wireless characteristics such as interference of radio signal, low bandwidth can lead wireless link unreliable. Link failure mostly occurs when mobile node tries to move out of its neighborhoods transmission range. In addition, battery depletion can make link breakage [11]. Thus, in addition to congestion, link failure and wireless channel error have significant contribution in generating loss in MANET.

A significant research effort has been observed in recent years on handling route failures and congestion in Mobile Ad Hoc Networks. We can categorize them into four different types based on their working principles as shown in table 2.2.

Table 2.2: Protocol classification based on working principle

Category	**Working Principle**	**Example**
First	Use back up route	AODV-BR, AOMDV
Second	Use Multiple routes to balance traffic loads	MP-DSR, SMR
Third	Use Local recovery process	IBR-AODV
Fourth	Store packets temporarily in memory	THR

The first category of works use backup routes on the failure of primary route. In AODV with Backup Routing (AODV-BR) [39] and Multi-path AODV, AOMDV [40], source nodes create alternative routes to the destination and on failure of any one of them nodes deliver data packets using an alternative route; however, they suffer from two problems: stale route and duplicate packet transmission. In AODV-Based Backup Routing Scheme, AODV-BBS [36], each node maintains two hop neighborhood information for finding alternative routes. But, the maintenance of multiple alternative paths is difficult, costly and time-consuming, which turn reduces the network efficiency.

The second category of routing protocols use multiple routes to balance traffic loads on the event of congestion [17] or route failures and thus improve the network performance. For example, a distributed Multi-path DSR protocol (MP-DSR) [41] improves QoS with respect to end-to-end reliability; Split Multi-path Routing (SMR) [21] uses multiple routes to split traffic and mitigate congestion; nodes in congestion adaptive Routing Protocol (CRP) [22] use bypass routes to mitigate congestion, etc. But, the problem is that multiple route maintenance overhead affects the network throughput.

Thirdly, some research works focus on link failure of MANETs using local recovery process [19] [42]. Most of them try to find out a local path whenever a link failure occurs in the network. For example, in Local Repair AODV based on Link Prediction, LRAODV_LP [28], if a node detects that the signal strength goes below a predefined

threshold, it initiates a fresh route discovery rather than sending error message backward. In Implicit Back-up Routing-AODV (IBR-AODV)[43], a neighbor of an active route temporarily stores (overheard) data packets and acts like a back-up node. Whenever any link failure occurs in the network, this back-up node creates new route to the destination and sends data packets. Here, the problem is that many neighbor nodes have to store data packets unnecessarily. Also more than one neighbor nodes may try to send the same data packet. This results in duplicate packet transmission of data packets and reduces network efficiency.

The fourth category of works handle link failure using packet buffering or route caching. Ela Kumar et.al proposed a two hop routing (THR) [44] protocol, in which if a node needs to transmit data it first checks its own routing table which contains route of two hop distance nodes. If any route is found only then data packets are delivered; otherwise, a fresh route discovery process is initiated. If route failure occurs, the intermediate nodes starts packet buffering in a separate physical memory module. However, the requirement of a separate memory module is not only costlier but also not implementable for all devices in the network.

The aforementioned protocols can handle either congestion or link failure. To cope up with both the problems simultaneously, we propose LCRDD, in which each node is capable of buffering data packets. When a node detects that the networks is congested or link toward the destination node is broken, it buffers the incoming packets into its own transport layer buffer. Later, on finding a new path, the node resumes transmission process from local buffer. Also, our LCRDD implements a congestion-aware reliable data delivery mechanism by using multi-level congestion detection and control mechanism. The table 2.3 shows the comparison of various routing protocols based on capable of handling link failure and congestion control.

Table 2.3: Protocol comparaison

Protocol	**Congestion**	**Link Failure**	**Buffering**
AODV-BR	No	Yes	No
AODV-BBS	No	Yes	No
AOMDV	No	Yes	No
SMR	Yes	No	No
CRP	Yes	No	No
IBR-AODV	No	Yes	No
THR	No	Yes	Yes
LCRDD	Yes	Yes	Yes

2.11 Discussion

In this chapter, we try to describe the various routing protocols with their basic characteristics and working principles. Some of the existing routing protocols are capable of handling the route failure, while some others cope up with congestion.

2.12 Summary

In summary, we can say that, the existing routing protocols either can handle congestion or link failure in Mobile Ad Hoc Networks. Link failure and the congestion are two vital problems. To make the network more efficient, we propose the LCRDD mechanism that can handle both link failure and congestion of the MANETs. What follows, we will describe the overall architecture of LCRDD mechanism and detail working procedure in Chapter 3.

Chapter 3

LCRDD Mechanism

In this chapter, we describe the proposed Link Failure and Congestion Aware Reliable Data Delivery Mechanism (LCRDD) for Mobile Ad Hoc Networks. LCRDD introduces a new buffering concept in each node's transport layer queue (TQ) to handle link failure and congestion. We explain in detail how LCRDD can handle both link failure and congestion of Mobile Ad Hoc Networks.

3.1 Introduction

Nodes in Mobile Ad Hoc Networks(MANETs) are self-organizing and self-configurable. Topology of MANET network changes due to nodes mobility which causes packet loss. To minimize the link failure as well as the packets loss rate, we introduce a buffering concept. We assume that, nodes can store or buffer the packets in their local transport layer queues (TQs) during link failure with the help of cross-layer interface. We propose a reliable data delivery mechanism namely LCRDD with detail description and the working procedure. LCRDD is capable of handling link failure. The multi-level congestion control mechanism leads to a reliable and effective data delivery mechanism in Mobile Ad Hoc Networks.

3.2 Network Model and Assumptions

We consider a Mobile Ad Hoc Network (MANET) with large number of nodes communicating over multi-hop paths with each other. The nodes may be mobile phones, laptops or PDAs, which may move frequently. The nodes use on-demand routing protocol AODV [27] or DSR [41] for establishing multi-hop routing paths. In what follows, we describe three important processes assumed in our network: Link failure detection process, local route repaire process and transport layer queue (TQ) management process.

3.2.1 Link Failure Detection Process

Since nodes in MANETs are self-organizing and independent of each other, they can move randomly and frequently. As a result topology of MANETs changes very frequently causing failure of links between neighbor nodes. Also, the link failure may be occurred due to highly reduced received signal strength. Link failure leads to route failure between nodes and reduces the network throughput as well. Therefore, the link failure detection is a subjective research issue in wireless networks. Link failure detection can be performed using either periodic HELLO messages [45] or link layer feedback [8], [46]. These HELLO messages are local advertisements for the continued presence of the link.

The nodes in our proposed LCRDD mechanism exchange HELLO messages periodically to ensure link connectivity. The following two parameters are associated with a HELLO message: $HELLO\_INTERVAL$ is the maximum time interval between two consecutive HELLO messages transmissions and $HELLO\_LOSS_{allowed}$ is the maximum number of loss of HELLO messages that a node can tolerate before it declares the link breakage. If a node does not receive any HELLO message from its neighbor node within $HELLO\_LOSS_{allowed} \times HELLO\_INTERVAL$, then the node assumes that the link is not available for data transmission.

3.2.2 Local Route Repairing Process

Link failure degrades the performance of a network significantly. Local repair of route can reduce the effects of link failure to some extent. There are a number of local route repair techniques in the literature. Query localization technique [47] keeps the history of an old path and floods the RREQ message to some restricted area during local query process. Sung-Ju Lee and Mario Gerla proposed AODV-BR (Backup Route) [48] which can improve the performance of the AODV routing protocol by providing multiple alternate routes. Expanding ring search [49] is another local repairing technique to finding local path during link failure. We will use one of these methods for finding partial path during link breakage.

3.2.3 Transport Layer Queue (TQ) Management Process

Since a mobile node in MANETs can act as both as a router and as a host, in our LCRDD mechanism, all intermediate routing nodes use a separate transport layer queue (TQ) to store incoming data packets, when needed. We assume, as long as there is no link failure or the network does not become heavily congested, nodes act like a conventional router and simply forward data packets. But when the network is heavily congested or link failure occurs, nodes use their TQs for buffering incoming packets.

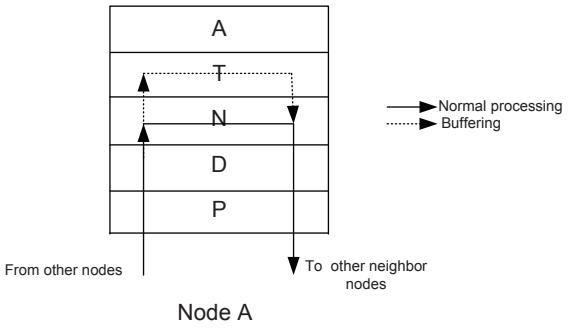


Figure 3.1: A MANET node works both as a router and a host

Figure 3.1 shows the node A is acting as router in normal operation (indicated by solid line). After receiving packets, it just forwards the packets to next hop using lower three (physical, data link and network) layers of TCP/IP model. But in case of link failure, node A uses its transport layer queue to buffer data packets (indicated by dotted line) and starts a local query process for a partial path. If a new partial path is found, node A starts its transmission process from transport layer queue as well as node A continues it normal data delivery process from network layer queue. The dotted lines indicate performing a transmission process from local TQ and solid lines indicate normal data delivery process.

Now a question arises - *Why does LCRDD use separate transport layer queue (TQ) for buffering the packets?* A node in LCRDD uses a separate transport layer queue for the following reasons:

- **Avoid congestion**
- **Reduce unnecessary retransmission**
- **Ensure end-to-end reliability**

3.2.3.1 Avoid congestion

Nodes in MANETs deliver data packets from multiple source nodes to multiple destination nodes. For a lightly loaded network, all types of packets are queued in the network layer queue of a mobile node for a very small period of time. A node forwards packets from network layer queue by examining the destination node. But in case of link failure or for a congested network, nodes have to store data packets for relatively longer period of time since they have to wait for a partial path or a state when the network become normal. So congestion may occur in the network as queue overflow arises if all the packets are stored at network layer queue. To avoid the congestion, each node in the network

use a separate transport layer queue so that node can perform its normal operation from network layer.

3.2.3.2 Reduce unnecessary retransmission

Nodes cannot store data packets in their network layer queue during link failure or when congestion arises. If a node buffers incoming packets in network layer queue then packet forwarding for other connections might be hampered and queue overflow will arise and the node becomes congested. As a result, packet dropping rate at the node will increase and overall throughput of the network will decrease. The source node has to retransmit a lot of data packets due to congestion. For this reason, nodes in our network use separate queue in transport layer to buffer the data packets.

3.2.3.3 Ensure end-to-end reliability

In normal routing, source nodes are mainly responsible for ensuring the end-to-end reliability. But in LCRDD, during link failure each intermediate node buffers its incoming data packets in separate transport layer queue and the node does not stamp any new sequence number to these stored data packets. These buffered packets are delivered to the destination node with special care so that the end-to-end reliability can be ensured from the intermediate node rather than the source node. So the source node does not need to concern about the end-to-end reliability. As a result, reliability of the network increases and which in terns increases the network performance significantly. What follows, we describe these issue with an appropriate example.

In Fig. 3.2, S_1,S_2 and S_3 are transferring data packets to three different destination nodes D_1, D_2 and D_3 respectively using two intermediate nodes A and B. Suppose links B to D_1 and B to D_3 are broken. If node B uses network layer queue for buffering process, B has to buffer all the incoming data packets of source nodes S_1 and S_3 and at the same time it has to forward the data packets of destination node D_2. As a result, buffering

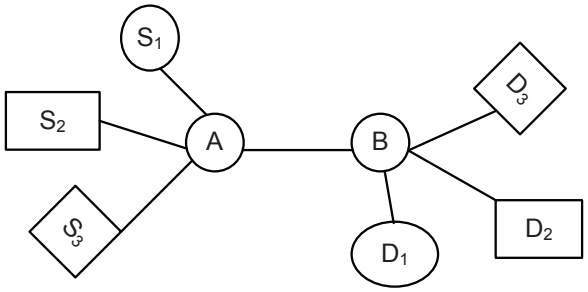


Figure 3.2: Node delivering data for multiple destination nodes

overflow will occur at network layer queue and thus network will be congested, leading to loss of data packets. To reduce this packets dropping rate, in LCRDD, node B uses separate transport layer queue (TQ) for buffering the data packets for destination nodes D_1 and D_3. So it helps to avoid the congestion occurring as well as ensure end-to-end reliability.

3.3 The LCRDD Architecture

Our proposed Link failure and Congestion aware Reliable Data Delivery (LCRDD) mechanism exploits cross-layer buffering capability of nodes, spanning through network and transport layers. In LCRDD, when an intermediate node detects a link failure, it creates a Route Disconnection Notification message (RDN) and sends toward the source node. It then buffers the packets into the transport layer queue and starts a local query to find any partial path to the destination. All the intermediate nodes, on receiving RDN message stop further delivery of the data packets and store the incoming data packets in their local transport layer queues. When the source node receives the RDN control message, it just stops transmission of data packets and waits for a new partial path to the destination.

If any partial path is found, node creates another notification message called Route Successful Notification message (RSN) and sends it back to the source node. It then resumes its transmission process from transport layer queue. All the intermediate nodes resume their transmission from local buffer after receiving RSN message. When the source node receives this RSN message it resumes the transmission process. So unlike the traditional routing protocols [1] [15], the source node does not need to deliver all the data packets during link failure. The source node only retransmits the packets that are dropped during the link failure. Thus packet loss rate is minimized and overall throughput of the network increases. But if no partial path is found, the intermediate node create another control message called Route Unsuccessful message (RUN) and sends it back to the source node. The source node on receiving the RUN message, starts a new route discovery process.

LCRDD uses congestion aware data delivery mechanism so that nodes in the network can easily identify the congestion level of the network and can take appropriate actions. For a heavily loaded network, nodes will send an ALERT message to previous nodes not to increase the data forwarding rate. In a heavily congested network, nodes buffer their incoming packets to reduce packet loss rate. Thus LCRDD control the congestion in Mobile Ad Hoc Network in an efficient and effective manner.

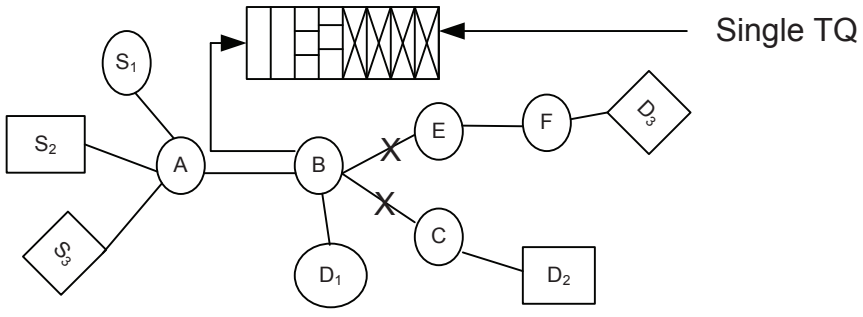


Figure 3.3: The use of transport layer queues at intermediate nodes

To handle link failure and congestion and to provide reliable data delivery mechanism, nodes buffer their incoming packets in their local queues. Each intermediate node maintains a queue in transport layer for multiple destination nodes with the help of cross-layer interface. For example, in Fig. 3.3, node B delivers data packets to three destination nodes D_1, D_2 and D_3. As the link between B to C and B to E are broken, node B starts to buffer the incoming data packets in its transport layer queue and it does not need to stamp any new sequence number for each packet during buffering process. Thus the proposed transport layer queue is only for storing the packets temporarily. What follows, we discuss the LCRDD operations at different mobile nodes in detail and describe the congestion-aware reliable data delivery mechanism.

3.3.1 Nodal Operations

In LCRDD, nodes can buffer packets in their local transport layer queue for link failure and congestion control. So every node in the performs some distinct functions than the conventional mobile nodes. We describe these distinct functions of a mobile nodes with detail description.

3.3.1.1 LCRDD in Source Nodes

In most of the routing protocols, the source node starts route discovery process by creating a message called route request message (RREQ) and floods the message [1] [35]. The source node then waits for a route to the destination. Whenever the source node gets a route reply message (RREP), it starts data transmission process. If there is a link failure due to node mobility or any other reasons such as limited bandwidth, lacks of power etc., and if the source node gets the route failure notification message, it stops further transmission of data packets. Then it starts a new fresh route discovery process to deliver the data packets. Due to link failure, source node need to retransmit a lot of data packets specially when the packets traverse a longer path. Thus data delivery rate

Algorithm 1 LCRDD in any source node $s \in S$

```
1. Begin
2. s broadcasts RREQ message
3. s waits until a message(msg) is received
4. if (msg=RREP) then
5.       s starts delivering data packets
6. else if (msg=RDN) then
7.       s stops transmission and waits for a partial path
8. else if  (msg=RSN) then
9.       s resumes transmission process
10. else if  (msg=RUN) then
11.      s initiates a fresh route discovery process
12. else
13.      s waits for a new message
14. end if
15. End
```

decreases and overall performance of the network also decreases.

In our model, the source node stops it transmission process during link failure if it gets a RDN message from any intermediate node and waits for a local path rather conducting a fresh route discovery process for that particular destination. When a partial path is established and the source node is notified by the RSN message it resumes the transmission process. The algorithm 1 shows the overall operations of a source node, $s \in S$, where S is the set of all source nodes.

3.3.1.2 LCRDD in Intermediate Nodes

In conventional routing protocols, all the intermediate nodes act as routers or forwarder nodes [1] [13]. Each node receives the incoming packets and forwards them to the next

node. When a link is broken, the intermediate node sends a route error (RERR) message to source node and drops all the packets for that destination. As a result, a lot of data packets are dropped due to link failures at each intermediate node.

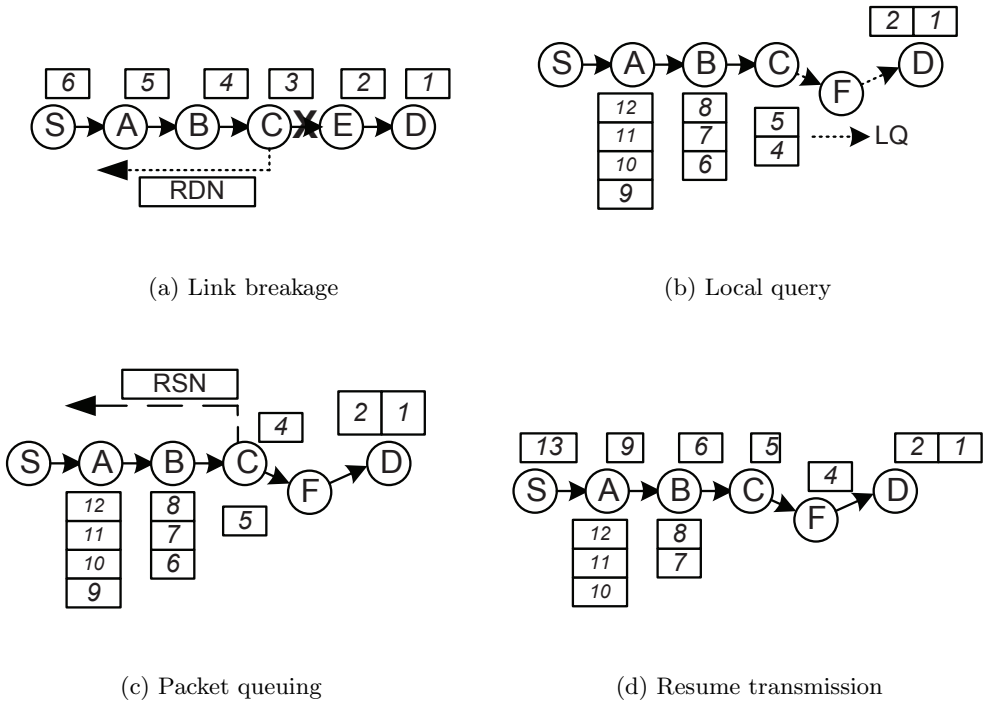
It is mentioned earlier that, in LCRDD, when an intermediate node detects a link failure, it starts buffering all the incoming packets in its local transport layer queue (TQ) rather dropping them. It then sends a Route Disconnection Notification (RDN) message towards the source node and all the intermediate nodes store their incoming data packets on recieving RDN message and wait for local path. In the meantime, the detecting node sends a local query to find a partial path using any of the existing algorithms, mentioned in section 3.2 and waits for a predefined short period of time T_t. If a partial path is found within T_t, the node starts transmission process from its TQ and sends Route Successful Notification (RSN) message toward the source node. All the intermediate nodes resume their packet delivery process from local queues after receiving the RSN message. However, if no partial path is found, the intermediate node sends the Route Unsuccessful Notification message (RUN) toward the source node. After getting RUN message the source node starts a new fresh route discovery process. The algorithm 2 shows the overall operations of any intermediate node, $n \in N$, where N is the set of all intermediate nodes in the route.

Algorithm 2 LCRDD in any intermediate node $n \in N$

```
1. Begin
2. if (n detects a link failure) then
3.     n initiates local query for partial route discovery
4.     n starts buffering data packets in TQ
5.     n waits for partial route for T_t duration
6.     if (partial path is found) then
7.         n resumes transmission process.
8.     else
9.         n sends RUN message to source
10.    end if
11. else
12.    n delivers the data packets
13. end if
14. End
```

The following example describes the operations of the nodes in LCRDD, using the above two algorithms.

In Fig. 3.4(a), the source node S is sending data packets via route $S \rightarrow A \rightarrow B \rightarrow C \rightarrow E \rightarrow D$ to destination node D. Suppose node C detects that the link between C and E has broken. It then generates a RDN message and sends it back to the source node to inform about the link failure. Also, node C starts buffering the incoming packets for node D and at the same time it initiates a local query (LQ) process to find a partial path. Figure 3.4(b) shows local query process. After receiving RDN message, all the intermediate nodes stop data delivery for the destination node D and buffer the incoming data packets in their local TQs. Figure 3.4(c) shows buffering process at each intermediate nodes (A, B, C). When the source S receives the RDN message, it stops its transmission and waits for a partial route.



(a) Link breakage

(b) Local query

(c) Packet queuing

(d) Resume transmission

Figure 3.4: Link failure handling in LCRDD

After finding a partial path $C \rightarrow F \rightarrow D$, node C sends back a RSN message toward S and resumes its transmission process from its local TQ. The intermediate nodes A and B and the source S resume their transmission process on receiving the RSN message. Note here that, the source node S does not need to retransmit all the data packets. Figure 3.4(d) shows the retransmission process whenever a route reconstruction process is successful. However, if node C's local query process fails to find a partial route within T_t, it sends the RUN message toward the source S. Upon reception of RUN message, S starts discovering a new route.

3.3.2 Congestion Control in LCRDD

In MANETs, the data traffic from many source nodes may converge at a point of the network, where some nodes might be congested due to their low packet forwarding rate compared to corresponding arrival rate. As a result, their local buffer might overflow and packet may be dropped [50]. Such a congestive state also leads to the following problems in the network: (i) longer end-to-end packet delivery delay and (ii) reduced network throughput.

To cope up with above drawbacks, in this paper, we propose a congestion-aware data delivery mechanism that works as follows. At each intermediate node, we measure the congestion level and piggybacks that information toward the source node so that appropriate control actions can be taken in time. We use two bits control flag in both data packets and acknowledgment packets, referred to as Congestion Notification (CN) flag. Every node in an active route sets this flag when they forward packets. The value of the CN flag detects the congestion level of the network according to table 3.1. From the value of the CN flag, the neighborhood nodes can easily be informed about the congestion status of the network and they can take proper actions to handle congestion. What follows, we describe how a node detects the congestion level and assign the value of CN and how the congestion is controlled based on the value of CN flag.

Table 3.1: Congestion notification

Value of CN	**Congestion level**
00	Lightly loaded
01	Loaded
10	Heavily Loaded
11	Congested

3.3.2.1 Detection of Congestion Level

Based on queue occupany, here we use an early congestion detection technique by which a node can detect the current congestion status. We use minimum and maximum thresholds, Q_{min} and Q_{max}, respectively, for queue occupancy at any node as follows:

$$Q_{min} = l \times Q_{size}, \tag{3.1}$$

$$Q_{max} = h \times Q_{size}, \tag{3.2}$$

where, l and h are two control parameters; in our simulation, we set $l = 0.5$ and $h = 0.9$, respectively. If the queue length of a node is less than the Q_{min} then we can say the network is lightly loaded, e.g., queue occupancy is less than 50%; if the queue length is greater than Q_{min} but less than Q_{max}, then it is operating in the safe region; and, if the queue length is greater than Q_{max}, the node is considered as congested. Even though the above thresholds help to identify congestive or non-congestive states, they don't protect nodes moving from non-congestive state to congestive one. In support of implementing congestion-aware data delivery mechanism, we introduce a warning threshold parameter Q_{warn}, defined as follows:

$$Q_{warn} = w \times Q_{size}, \tag{3.3}$$

where, w is a weight factor and in our simulation we choose $w = 0.8$.

We then calculate average queue occupancy of a node every after a certain interval using Exponentially Weighted Moving Average (EWMA) formula as follows:

$$Q_{avg} = (1 - \alpha) \times Q_{avg} + Q_{curr} \times \alpha \tag{3.4}$$

where α is a weight factor and Q_{curr} is the current queue size. Now, based on the value of Q_{avg}, we determine the value of CN flags as follows:

- if $Q_{avg} < Q_{min}$ then CN = 00

- if $Q_{avg} >= Q_{min}$ and $Q_{avg} < Q_{warn}$ then CN = 01
- if $Q_{avg} >= Q_{warn}$ and $Q_{avg} <= Q_{max}$ then CN = 10
- if $Q_{avg} > Q_{max}$ then CN = 11

3.3.2.2 Congestion Control Mechanism

In our proposed LCRDD, a node takes proper actions to control the congestion according to the congestion level of the network. We can divide the actions into following parts:

- **Normal Operation:** If the value of CN flag at a node is '00', it assumes the network is lightly loaded. In such case, the node performs its normal operations. It allows the other nodes of the network to transmit packets through it. So the node accepts new RREQ messages from new source and rebroadcasts to create new routes through it.
- **Admission Control:** When the value of CN is '01', an intermediate node discards any new RREQ messages, so that no new route can create through the node in order to avoid any future congestive states. So no new source can create new route to deliver data packets. Thus LCRDD control the admission of new RREQ message. However, in this case, the sources of the existing routes may increase their traffic rates passing through this node.
- **Rate Control:** If the value of the CN is '10', the network becomes heavily loaded. So, further increasing of data arrival rates from source nodes will lead the network to fall into congestive state. In this case, the node generates a new control message called ALERT message and sends back toward every source nodes so that they do not increase the data forwarding rates. Thus, our proposed LCRDD controls the network in ahead of time and implements a congestion-aware reliable data delivery mechanism.

- **Storing Packets:** But if a node detects the value of CN is '11', this means the network has already fallen into congestive state and it then starts buffering data packets in TQ and stops forwarding packets afterwards and waits for the normal state of the network.

The table 3.2 shows the operations taken by a node for different congestive states.

Table 3.2: Actions taken by a node to control the congestion

Congestion Level	**Actions**
Lightly loaded	Normal operation - no change
Loaded	Stop forwarding any RREQ message
Heavily loaded	Send ALERT message to all sources
Congested	Start buffering packets

3.3.3 Packet Buffering

In this section, we describe in detail the packet buffering mechanism. As mentioned earlier, nodes in our network use separate queue in transport layer (TQ) when link-failure or congestion occurs. Storing packets at separate queue reduce the probability of queue overhead at network layer.

We define a cross-layer interface between network layer and transport layer as shown in Fig. 3.5. The interface has two components: receive interface "R" and delivery interface "D". The interface "R" receives the packets from network layer queue and puts them into the transport layer queue when a link failure or congestion occurs in the network. Similarly, all the intermediate nodes buffer the packets in their transport layer queues for that corresponding destination using their cross-layer interfaces. Whenever, a partial path is found, the node informs transport layer through the interface. Then the interface

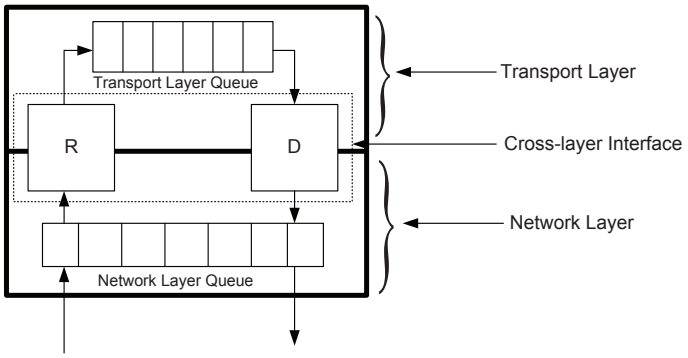


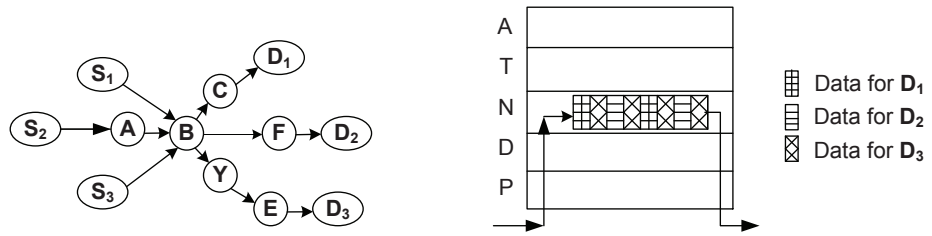
Figure 3.5: Cross-layer interface between network layer and transport layer

"D" delivers the data packets to network layer and the node resumes data transmission process. Similarly, all the intermediate nodes resume their transmission process. As a result, the source node does not need to retransmit all the data packets during a link failure, increasing the overall throughput of the network.

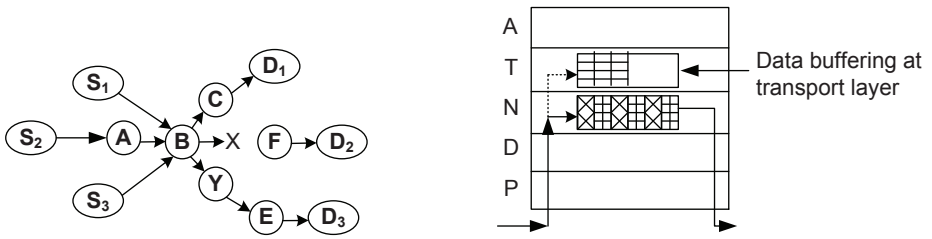
In what follows, we describe the aforementioned buffering mechanism with the help of an example. Consider the Fig. 3.6(a), where node B forwards the data packets of three sources S_1, S_2 and S_3 to three destination nodes D_1, D_2 and D_3, respectively. Figure 3.6(a) shows the normal situation in which all the data packets are queued at network layer queue of node B.

Now, suppose the link between node B and F is broken. As soon as node B detects the link failure, it buffers all the nodes for the destination node D_2 in its TQ and continues its normal operation for other destinations. Meanwhile, B starts a local query process for destination node D_2 and sends a RDN message toward the source node S_2. All the intermediate nodes buffer their packets for the destination node D_2 in their (TQs) on receiving RDN message. Node S_2 stops its transmission on receiving the RDN message and waits for a reply from node B for a partial path. Figure 3.6(b) describes this situation.

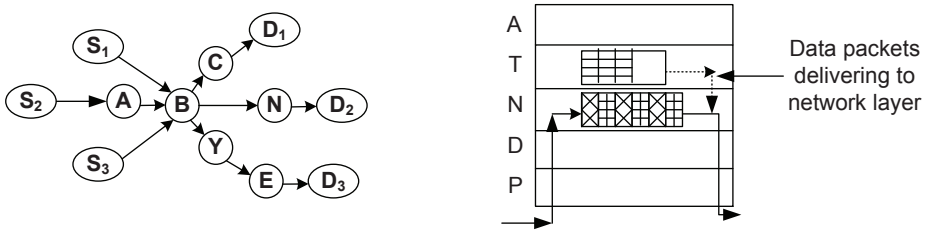
Whenever node B finds any partial path $B \rightarrow N \rightarrow D_2$, it sends RSN control message toward source S_2 and resumes its transmission process from transport layer queue. All



(a) Normal operation



(b) Packets buffering at TQ



(c) Packet Delivery

Figure 3.6: Operation of the proposed LCRDD mechanism

the intermediate nodes resume their transmission process same way after getting the RSN message. Node S resumes it transmission process after receiving the RSN message. Such a mechanism decreases packet dropping rate and node S_2 does not need to transfer all the packets. Figure 3.6(c) shows node B performs normal operation from network layer as well as resumes transmission process from its TQ for destination node D_2. Thus LCRDD helps to reduce packets dropping rate using the concept of buffering the data packets during link failure with the help of cross-layer interface between network and transport layer.

3.4 Discussion

Our proposed LCRDD mechanism can buffer packets during route failure as well as when congestion arises in the network. Since topology of the Mobile Ad Hoc Networks change dynamically and frequently, rapid change in routes leads to a lot of data packet to be dropped. Again for the congestion, nodes in a ad hoc network drop their data packets. As a result, throughput of the network decreases which reduce the network efficiency significantly. Our LCRDD mechanism minimize this packet loss and thus improve the network performance by it's packet buffering concept using cross-layer interface.

3.5 Summary

In summary we can say that, the our proposed LCRDD mechanism can handle congestion or link failure in Mobile Ad Hoc Networks with the concept of buffering capabilities. What follows, in Chapter 4 we will describe the performance comparison of LCRDD with several stated protocols those are capable of handling the link failures or the congestion of Mobile Ad Hoc Networks.

Chapter 4

Performance Evaluation

In this chapter, we evaluate the performance of LCRDD. We compare the performance of LCRDD with other studied protocols. The result shows that LCRDD outperforms than the other protocols.

4.1 Introduction

In this chapter, we evaluate the performance of our proposed LCRDD mechanism in network simulator v-2.34 [51] and compare the simulation results with that of AODV-BBS [36], SMR [21], IBR-AODV [43] and THR [44]. The results of our simulation state that the LCRDD outperforms than the other protocols.

4.2 Simulation Environment

In our simulation, we consider a square area of size $1000 \times 1000m^2$, where 100 mobile nodes are deployed randomly [52]. The simulation time is set to 200 seconds. Each node has the transmission range of 250 m. The source nodes of our network generate constant bit rate (CBR) data streams at the rate of 1 to 8 packets per second. This helps to measure performance for various traffic load at each mobile node. The size of each data packet is 512 bytes, link bandwidth is kept at 11 Mbps and the underlying transport

Table 4.1: Simulation parameters

Parameter	**Value**
Network area	1000m x 1000m
Number of nodes	100
Deployment type	Random
Number of sources	20
Node movement model	Random waypoint
Transmission range	100m
Transport layer protocol	UDP
MAC layer protocol	IEEE 802.11 DCF
Bandwidth	11 Mbps
Data packet size	512 bytes
Data packet generation rate	1 to 8 packets/sec
Propagation model	Free Space
Weight factor α	0.002
T_t	2 sec

and MAC layer protocols are UDP and IEEE 802.11 DCF, respectively. The table 4.1 summarizes the simulation parameters. For each data point in the graphs, we take the average of 10 simulation runs that helps us to study the steady state behavior of the protocols.

4.3 Performance Metrics

We use four performance metrics to compare the results of AODV-BBS, SMR, IBR-AODV, THR and LCRDD. Those metrics are as follows:

- *Packet delivery ratio* is measured as the ratio of the total number of received data packets by all the destination nodes to the total number of generated data packets by all the source nodes in the network.
- *Average end-to-end packet delay* is measured as the average time in ms required by all the data packets that are received by the destination nodes.
- *Per node throughput* is measured as the average amount of data bits received per unit time by all the destination nodes in the network.
- *Normalized routing overhead* is measured as the number of control packets generated during the simulation period for each successfully delivered data packet.

4.4 Impact of Varying Traffic Loads

In this section, we study the impact of different traffic loads on the performances of the protocols in terms of the above metrics. The data traffic loads at source nodes are varied from $1 \sim 8$ packets per seconds, i.e., from $0.5 \sim 4.0$ KBps. For this experiment, we fix the mobility speed of each node at 2 m/s within the network.

4.4.1 Packet Delivery Ratio

The packet delivery ratio for all the protocols decreases drastically with increased source traffic loads, as shown in Fig. 4.1. This is caused by increased packet drops at the intermediate nodes due to congestion, i.e., the forwarder nodes cannot deliver as many packets as they receive. The simulation results indicate that our proposed LCRDD machanism has the higher capability to handle congestion in the network and it outperforms

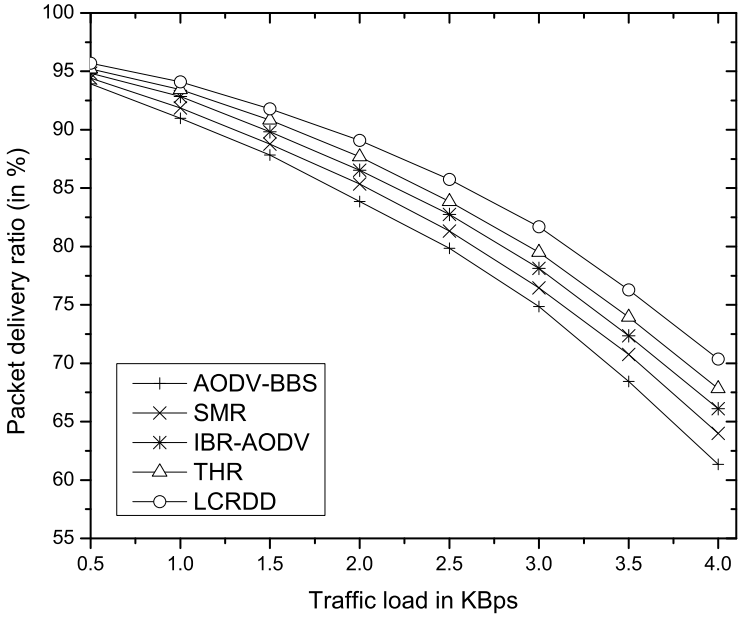


Figure 4.1: Packet delivery ratio

other protocols. Since LCRDD measures the pre-congestive and congestive states more accurately and propagates information along the routes and takes appropriate control actions in time, its congestion control mechanism becomes more efficient than others. Furthermore, the buffering mechanism of LCRDD reduces the packet drops a lot and thus increases the packet delivery ratio.

4.4.2 Average end-to-end Delay

Figure 4.2 shows the average end-to-end packet delay performances of the protocols for various traffic loads. It states that, as expected theoretically, the packet delivery delay increases with the traffic loads. AODV-BBS and SMR experience much higher delay than others due to high latency of maintaining multiple alternate routes. IBR-AODV has longer delay than THR and LCRDD since it can not handle the congestion, uses backup nodes only for providing local recovery from route failures. However, our LCRDD handles

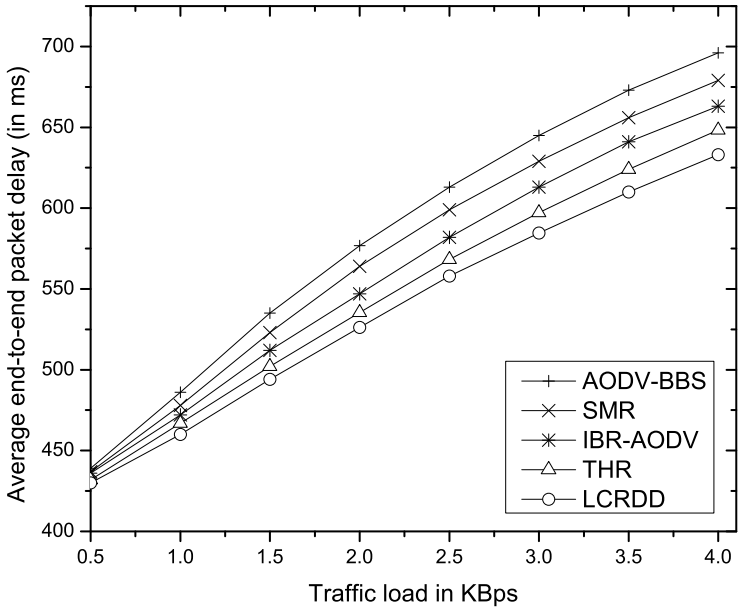


Figure 4.2: Average end-to-end delay

congestion proactively by not allowing an intermediate node to carry additional traffic (i) from new connections when it detects *Loaded* state and (ii) from existing connections when it detects *Heavily loaded* state. This strategy helps LCRDD nodes to operate in safe mode (i.e., loaded state) most of the time and thus decreases the queuing delays of the packets at the intermediate nodes, which in turn decreases the end-to-end packet delivery delay a lot.

4.4.3 Throughput

The performance comparisons for per node throughput of the studied protocols have been shown in Fig. 4.3. The throughputs of the protocols increase as the traffic load increases but it starts decreasing at around 3.5KBps traffic load, where the network reaches at saturation condition. The further increase of traffic load makes the network congested and thus the performance decreases. AODV-BBS experiences the lowest throughput since

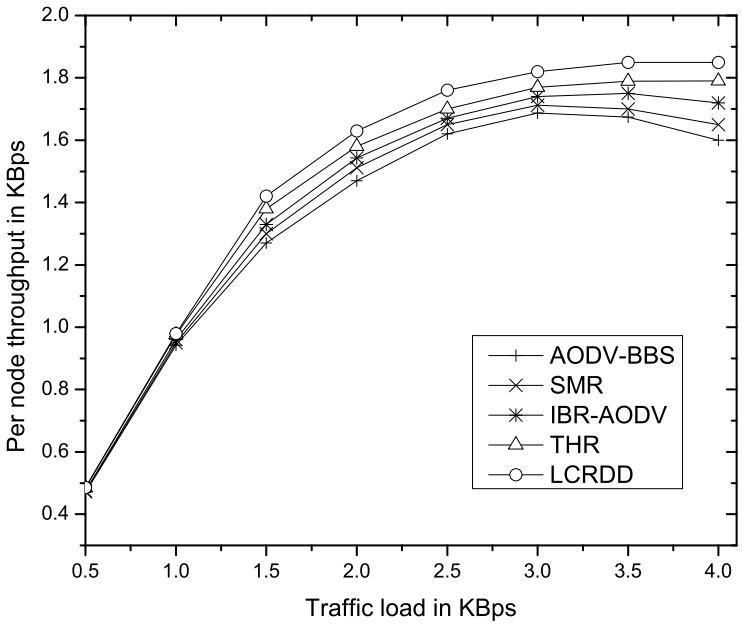


Figure 4.3: Throughput

packet loss due to congestion at each node arise with the increasing of traffic loads. SMR performs better than AODV-BBS since it use multiple routes in safe zone. IBR-AODV experiences higher throughput for fast recovery process. We observe that our LCRDD provides high performance than the other protocols since it ensures higher number of packet delivery at the destination within minimum end-to-end delay.

4.4.4 Normalized Routing Overhead

Figure 4.4 shows the normalized routing overhead of the studied protocols. The AODV-BBS has high latency of route discovery process for keeping multiple routes for same destination and generates large amount of control packets. During link failure, SMR needs a fresh route discovery process, producing a large number of control packets. Since IBR-AODV stores data packets at multiple neighborhood nodes and exchanges control packets on the failure of links, its overhead increases a lot.

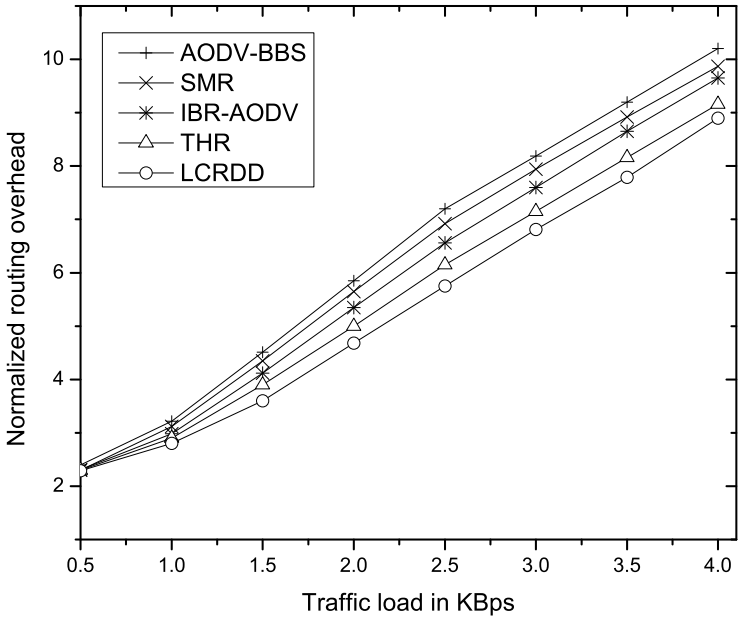


Figure 4.4: Normalized routing overhead

The THR can store data packets during link failure but can not cope up with network congestion as traffic load increases and local route discovery process starts on the failure of links. On the other hand, our LCRDD neither maintains any alternative routes nor it balances loads among multiple routes; rather, it uses multi-level congestion control mechanism and local packet buffering at transport layer for the period of local route discovery to mitigate link failures and congestion in the network. As a result, it generates minimum number of control packets and thus provides with the least normalized routing overhead among the studied protocols.

4.5 Impact of Varying Route Failure Rates

In this section, we evaluate the impact of varying route failure rates on the performances of the studied protocols, keeping the packets generation rate constant at 6 packets per seconds (i.e., 3KBps). We vary the route failure rates from $1 \sim 10$ routes/sec.

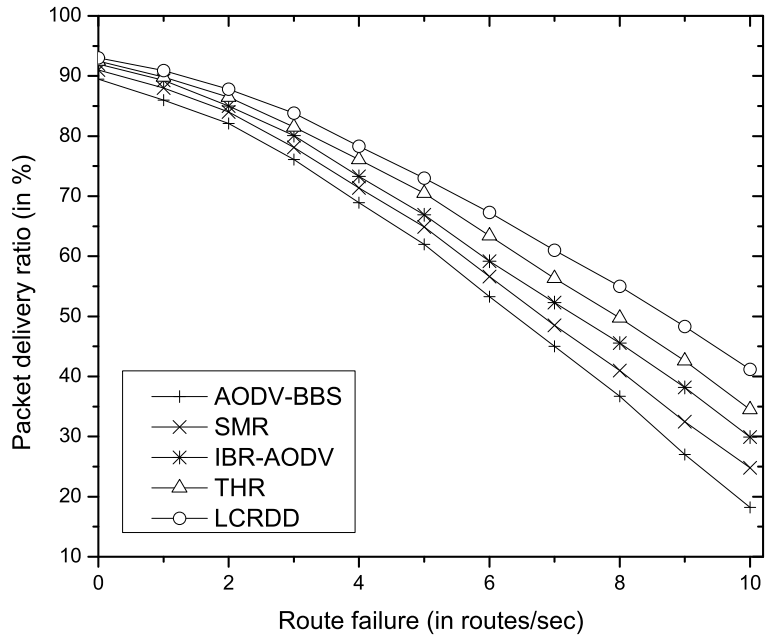


Figure 4.5: Packet delivery ratio

4.5.1 Packet Delivery Ratio

The graphs in Fig. 4.5 state that the packet delivery ratio for the all protocols decrease sharply with the increasing route failure rates. This happens because the failure of routes increases the number of packet drops. Since our proposed LCRDD jointly exploits the multilevel congestion control and packet buffering mechanisms on the event of link failures, it is capable to address the route failures more effectively and saves packets from dropping and thus it can increase the packet delivery ratio compared to other protocols.

4.5.2 Average end-to-end Delay

As expected theoretically, the end-to-end packet delivery delay increases with the route failure rates for all the protocols, as shown in Fig. 4.6. SMR experiences the longest

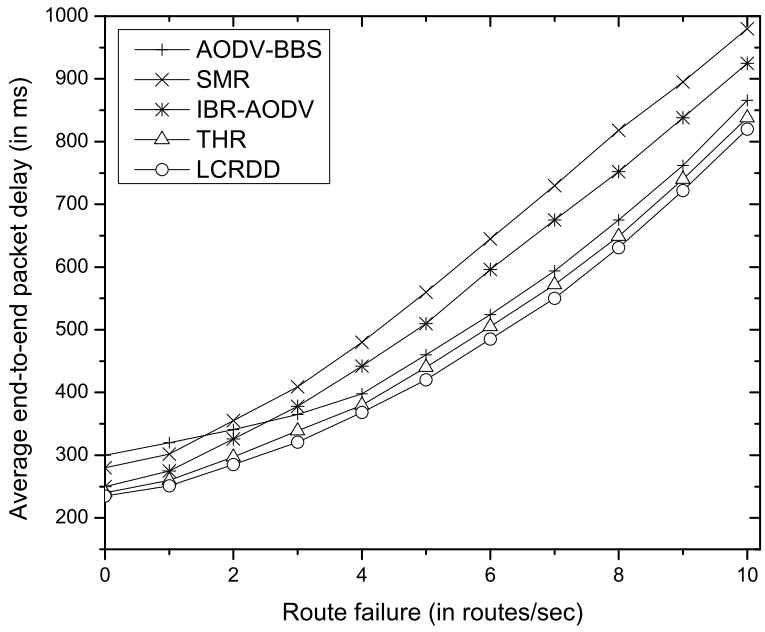


Figure 4.6: Average end-to-end delay

end-to-end packet delivery delay. IBR-AODV has lower end-to-end delay than SMR but higher than others protocols. AODV-BBS uses an altenative route whenever a link failure is detected and thus it decreases the delay than SMR and IBR-AODV. The local route discovery-assisted packet buffering mechanism in LCRDD helps it to handle route failures more efficiently than others. The number of retransmissions required on the failure of routes decreases a lot and thus it reduces the travelling time of data packets.

4.5.3 Throughput

Figure 4.7 shows the throughput performances of the studied protocols. The per node throughput decreases for all the protocols as the route failure rate increases. Because of high latency multiple paths, AODV-BBS provides lower throughput. As route failure rate increase, SMR has to lot of data packets. It experience higher throughput than AODV-BBS since it balance the load using multiple route simultaneously and when the

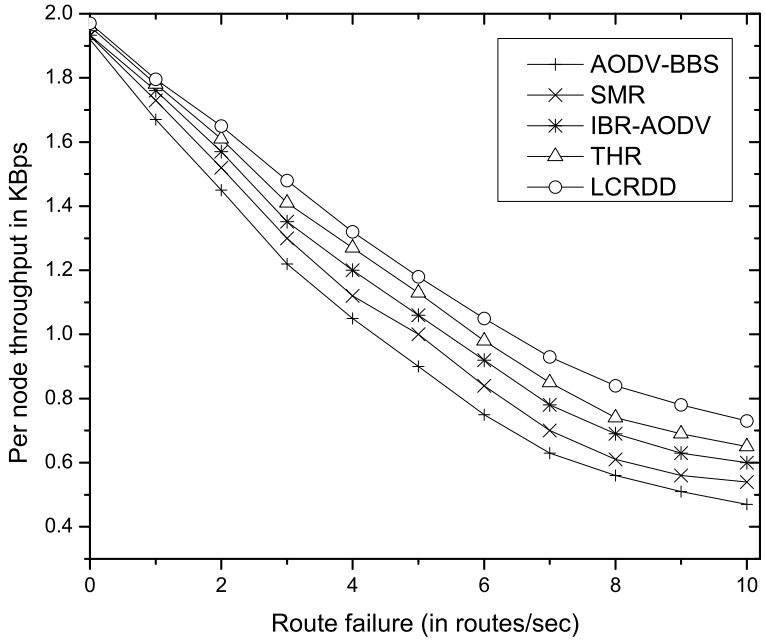


Figure 4.7: Throughput

network is in safe state. IBR-AODV provides higher results since it initiates fast recovery. LCRDD has the higher throughput over the four other protocol mainly for its (i) efficient route failure handling mechanism and (ii) congestion aware data delivery mechanism.

4.5.4 Normalized Routing Overhead

Figure 4.8 shows that the normalized routing overhead increases as route failure rate rises the later causes the substantial increase in the number of control packets generated in the network. For example, AODV-BBS has to generate more control packets for multiple route maintaining. Similarly for SMR link failure leads to retransmit a large number of data packets. SMR creates fresh route in such a case. As a result it generates a lot of control packets.

In IBR-AODV, setting up connection with all the back-up node requires more control packets. As route failure rate increases over the time it needs to communicate with more

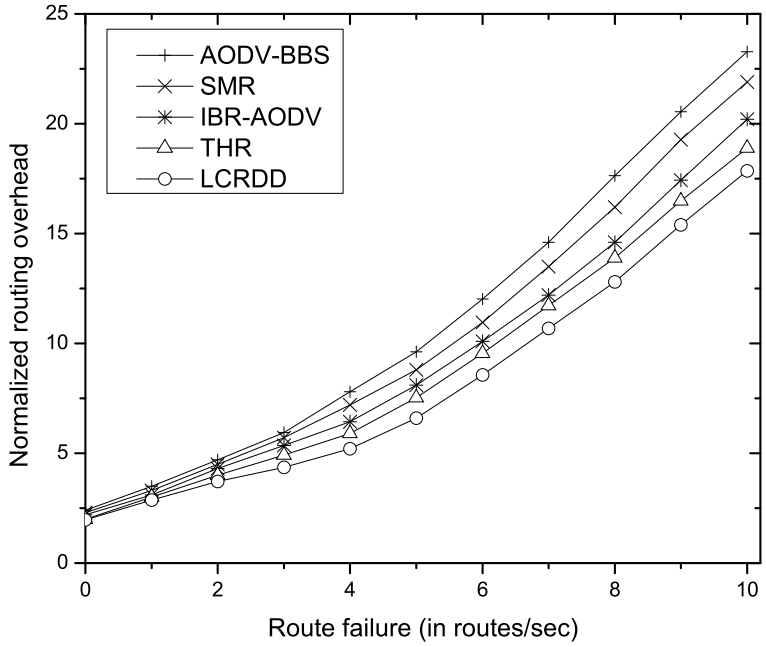


Figure 4.8: Normalized routing overhead

back-up node. For THR, nodes initiate fresh route discovery if it detects a link failure. The high latency of fresh route discovery process causes the higher routing overhead than LCRDD. Because of packet buffering capabilities and local recovery process, a LCRDD source node does not need to retransmit all the data packets. Comparing to the other protocol LCRDD generate lower number of control protocol. Thus it provides low overhead than other protocols.

4.6 Discussion

The results of the performance metrics indicate that, LCRDD provides better performance than the other studied protocols. The packet delivery ratio and throughput of LCRDD is better than the others. Because of reliable data delivery mechanism with buffering capability, LCRDD experiences the low end-to-end packet delivery delay and

low normalized routing overhead.

4.7 Summary

At the end of this chapter, we can summarize that, our proposed LCRDD mechanism outperforms than state-of-the-arts routing protocols. It is capable of increasing packet delivery ratio, throughput and reducing end-to-end data delivery delay and routing overhead of the network. We will mention some future works of LCRDD along with some limitations of LCRDD in Chapter 5.

Chapter 5

Conclusions

In this chapter, we describe the overall summary of our proposed LCRDD mechanism along with future work.

5.1 Summary of Research

In this research work, we develop a link failure and congestion aware reliable data delivery mechanism for Mobile Ad Hoc Networks that provides reliable data delivery using packet buffering concept. Link failure due to node mobility and congestion are two major problems in Mobile Ad Hoc Networks. Rapidly changing topology of Mobile Ad Hoc Networks causes the link failures. A lot of data packets need to retransmit for the source nodes during link failure. As a result, throughput of the network decreases significantly. Again channel overloading or queue overflowing leads the Mobile Ad Hoc Networks to a congested state. Thus, packets dropping rate increases rapidly which leads to longer data delivery delay and lower throughput of the network. To minimize the above two problems, we proposed our reliable data delivery mechanism LCRDD.

A significant research effort has been observed in recent years on handling route failures and congestion in mobile ad hoc networks. Some of them can handle link failure whereas some others can minimize the congestion. To handle both of the problems, we proposed LCRDD mechanism that provides reliable data delivery between nodes in a Mobile Ad Hoc Network.

To implement our proposed LCRDD mechanism we introduce the buffering concept with cross layer interface. Nodes in LCRDD, are capable of buffering the incoming data packets at transport layer queues (TQs) during link failure. On finding a partial path nodes can deliver the data packets from their local buffer. Thus, LCRDD helps to reduce packets loss rate at a minimized state and the efficiency of the network increases.

The multi-level congestion detection mechanism of LCRDD helps the nodes to take proper actions so that nodes can avoid the congestion occurring. Again, if the network become congested for other reasons, such as low power or bandwidth or link failure, nodes use their transport layer queue to buffer packets rather than dropping them and wait until the network become normal. Thus LCRDD leads to high throughput, low delay and overhead of the network.

Simulation results show that, LCRDD provides higher throughput and packet delivery ratio than the studied protocols. The end-to-end delay and the routing overhead are lower than the protocols that are related to handle the link failure and congestion.

We have addressed two very important problems: route failure and congestion in MANETs. Our proposed LCRDD mechanism introduces the concepts of local buffering at transport layer and multilevel congestion detection and proactive control actions which improve the network performance significantly. The results of our performance evaluations, carried out for various traffic loads and route failure rates, show that the proposed LCRDD mechanism outperforms a number of state-of-the-art approaches. Our mechanism is fully distributed and does not depend on network-wide information and thus, it reduces the operation overhead as well.

5.2 Limitations

Our proposed LCRDD improves the network performance during link failure and congestion of a Mobile Ad Hoc Network. We assume that, the network we use is densely

deployed. That is, the network has such number of nodes so that whenever any link failure occurs, node can find the path partially. If an intermediate node can not find any partial path during local recovery, then the source node has to starts a fresh route discovery. In such a case, our proposed mechanism will degrade the performance in terms of end-to-end delay. During the local recovery, all of the intermediate nodes buffer data packets. But if no partial path is found, the source node retransmits all those packets by creating a new route. So it will increase the end-to-end packet delivery delay significantly.

5.3 Future Works

Our proposed LCRDD mechanism introduces a new buffering concept to cope up with the two major problems in Mobile Ad Hoc Networks. LCRDD is fully capable of handling the link failure and can detect multi-level congestion of the network and controls with appropriate actions in Mobile Ad hoc Networks. Though LCRDD provides reliable and high packet delivery using the buffering concept, there may be enough scope to research on local queue management specially when buffering overflow occurs in Mobile Ad Hoc Networks.

Bibliography

[1] E. M. Royer and C. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications*, vol. 6, no. 2, pp. 46–55, 1999. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=760423

[2] V. Lenders, J. Wagner, S. Heimlicher, M. May, and B. Plattner, "An empirical study of the impact of mobility on link failures in an 802.11 ad hoc network," *Wireless Commun.*, vol. 15, no. 6, pp. 16–21, Dec. 2008. [Online]. Available: http://dx.doi.org/10.1109/MWC.2008.4749743

[3] V. Lenders, J. Wagner, and M. May, "Analyzing the impact of mobility in ad hoc networks," in *Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*, ser. REALMAN '06. New York, NY, USA: ACM, 2006, pp. 39–46. [Online]. Available: http://doi.acm.org/10.1145/1132983.1132991

[4] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *WIRELESS COMMUNICATIONS AND MOBILE COMPUTING (WCMC): SPECIAL ISSUE ON MOBILE AD HOC NETWORKING: RESEARCH, TRENDS AND APPLICATIONS*, vol. 2, pp. 483–502, 2002.

[5] V. Timcenko, M. Stojanovic, and S. B. Rakas, "Manet routing protocols vs. mobility models: performance analysis and comparison," in *Proceedings of the 9th WSEAS international conference on Applied informatics and communications*,

ser. AIC'09. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2009, pp. 271–276. [Online]. Available: http://dl.acm.org/citation.cfm?id=1628095.1628146

[6] L. W., "Minimizing energy and maximizing network lifetime multicasting in wireless ad hoc networks," *IEEE International Conference on Communication*, pp. 3375–3380, 2005.

[7] G. Lim, "Link stability and route lifetime in ad-hoc wireless networks," in *Proceedings of the 2002 International Conference on Parallel Processing Workshops*, ser. ICPPW '02. Washington, DC, USA: IEEE Computer Society, 2002, pp. 116–. [Online]. Available: http://dl.acm.org/citation.cfm?id=582629.848289

[8] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: the dynamic source routing protocol for multi-hop wireless ad hoc networks," in *In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5.* Addison-Wesley, 2001, pp. 139–172.

[9] K. D. and K. V., "EAAC: Energy-aware admission control scheme for ad hoc network," *World Academy of science, Engineering and Technology*, vol. 51, pp. 934 – 942, 2009.

[10] Liang, Weifa, Guo, and Xiaoxing, "Online multicasting for network capacity maximization in energy-constrained ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 9, pp. 1215–1227, Sep. 2006.

[11] W. Liang and Y. Yuansheng, "Maximizing battery life routing in wireless ad hoc networks," *Hawaii International Conference on System Sciences*, vol. 9, p. 90295, 2004.

[12] N. Vassileva and F. Barcelo-Arroyo, "A survey of routing protocols for energy constrained ad hoc wireless networks," in *Proceedings of the Future Generation*

Communication and Networking - Volume 01, ser. FGCN '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 522–527. [Online]. Available: http://dx.doi.org/10.1109/FGCN.2007.43

[13] S. Buruhanudeen, M. Othman, M. Othman, and B. M. Ali, "Existing manet routing protocols and metrics used towards the efficiency and reliability- an overview," 2007.

[14] L. Ouakil, S.-M. Senouci, and G. Pujolle, "Performance comparison of ad hoc routing protocols based on energy consumption," in *Ambience Workshop 2002*, 2002.

[15] R. R. and R. J., "A brief overview of ad hoc networks: challenges and directions," *Communications Magazine, IEEE*, vol. 40, pp. 20–22, 2002. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1006968

[16] S. Murthy and J. J. Garcia-Luna-Aceves, "Congestion-oriented shortest multipath routing," in *INFOCOM'96*, 1996, pp. 1028–1036.

[17] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Netw.*, vol. 1, no. 4, pp. 397–413, Aug. 1993. [Online]. Available: http://dx.doi.org/10.1109/90.251892

[18] V. Thilagavathe and D. K. Duraiswamy, "Cross layer based congestion control technique for reliable and energy aware routing in manet," *International Journal of Computer Applications*, vol. 36, no. 12, pp. 1–6, December 2011, published by Foundation of Computer Science, New York, USA.

[19] S.-J. Lee, E. M. Belding-Royer, and C. E. Perkins, "Scalability study of the ad hoc on-demand distance vector routing protocol," *Int. J. Netw. Manag.*, vol. 13, no. 2, pp. 97–114, Mar. 2003. [Online]. Available: http://dx.doi.org/10.1002/nem.463

[20] C. yih Wan and S. B. Eisenman, "Coda: Congestion detection and avoidance in sensor networks," in *Proceedings of ACM SenSys.* ACM Press, Nov. 2003, pp. 266–279.

[21] S. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," IEEE International Conference on Communications (ICC), pp. 3201–3205, 2001.

[22] A. Valarmathi and R. M. Chandrasekaran, "Congestion aware and adaptive dynamic source routing algorithm with load-balancing in manets," *International Journal*, vol. 8, no. 5, pp. 6–9, 2010.

[23] Y. Yu and G. B. Giannakis, "Cross-layer congestion and contention control for wireless ad hoc networks," 2008.

[24] H. Raghavendra and D. Tran, "Congestion adaptive routing in ad hoc networks (short version)," *ACM Intl Conf. Mobile Computing and Networking (MOBICOM)*, Oct. 2004.

[25] P. M., W. J.Y., L. W. Wang, Y. Zhong, and B. Bhargava, "Study of distance vector routing protocols for mobile ad hoc networks," *IEEE International Conference Pervasive Computing and Comm. (PerCom)*, pp. 187–194, 2003.

[26] M. R. Pearlman, Z. J. Haas, P. Sholander, and S. S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks," in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, ser. MobiHoc '00. Piscataway, NJ, USA: IEEE Press, 2000, pp. 3–10. [Online]. Available: http://dl.acm.org/citation.cfm?id=514151.514153

[27] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *In proceeding of the 2nd IEEE workshop on Mobile Computing Systems and Applications*, 1997, pp. 90–100.

[28] R. E, V. Kohir, and V. Mytri, "A local route repair algorithm based on link failure prediction in mobile ad hoc network," *World Journal of Science and Technology*, vol. 1, no. 8, 2011. [Online]. Available: http://worldjournalofscience.com/index.php/wjst/article/view/9396

[29] J. J. Garcia-Luna-Aceves, M. Mosko, and C. E. Perkins, "A new approach to on-demand loop-free routing in ad hoc networks," in *Proceedings of the twenty-second annual symposium on Principles of distributed computing*, ser. PODC '03. New York, NY, USA: ACM, 2003, pp. 53–62. [Online]. Available: http://doi.acm.org/10.1145/872035.872043

[30] C.-C. Liu, "An on-demand qos routing protocol for mobile ad hoc networks," in *Proceedings of the 8th IEEE International Conference on Networks*, ser. ICON '00. Washington, DC, USA: IEEE Computer Society, 2000, pp. 160–. [Online]. Available: http://dl.acm.org/citation.cfm?id=518913.847852

[31] T. B. Reddy, I. Karthigeyan, B. S. Manoj, and C. S. R. Murthy, "Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions," *Ad Hoc Netw.*, vol. 4, no. 1, pp. 83–124, Jan. 2006. [Online]. Available: http://dx.doi.org/10.1016/j.adhoc.2004.04.008

[32] Woo, Seung-Chul, M., Singh, and Suresh, "Scalable routing protocol for ad hoc networks," *Wirel. Netw.*, vol. 7, no. 5, pp. 513–529, Sep. 2001. [Online]. Available: http://dx.doi.org/10.1023/A:1016726711167

[33] J. Raju and J. Garcia-Luna-Aceves, "A comparison of on-demand and table driven routing for ad-hoc wireless networks," in *IEEE International Conference on Communications (ICC2000*, 2000, pp. 1702–1706.

[34] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based performance analysis of routing protocols for mobile ad-hoc

networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, ser. MobiCom '99. New York, NY, USA: ACM, 1999, pp. 195–206. [Online]. Available: http://doi.acm.org/10.1145/313451.313535

[35] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, ser. MobiCom '98. New York, NY, USA: ACM, 1998, pp. 85–97. [Online]. Available: http://doi.acm.org/10.1145/288235.288256

[36] T.-C. Huang, S.-Y. Huang, and L. Tang, "Aodv-based backup routing scheme in mobile ad hoc networks," in *Proceedings of the 2010 International Conference on Communications and Mobile Computing - Volume 03*, ser. CMC '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 254–258. [Online]. Available: http://dx.doi.org/10.1109/CMC.2010.313

[37] M. K. Marina and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks," in *in Proceedings of IEEE International Conference on Network Protocols (ICNP*, 2001, pp. 14–23.

[38] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *Proceedings of the INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution*, ser. INFOCOM '97. Washington, DC, USA: IEEE Computer Society, 1997, pp. 1405–. [Online]. Available: http://dl.acm.org/citation.cfm?id=839292.843010

[39] C. K. Toh, "Associativity-based routing protocol for mobile ad hoc networks," *Wireless personal communication*, vol. 4,2, pp. 103–109, March 1997.

[40] M. K. Marina and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks," *29th Annual IEEE International Conference on Local Computer Networks*, vol. 11-14 Nov., pp. 14–23, 2001. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1367254

[41] R. Leung, R. L. Jilei, E. Poon, A. lot Charles Chan, and B. Li, "MP-DSR: A QoS-aware multi-path dynamic source routing protocol for wireless ad-hoc networks," in *In IEEE LCN01*, 2001, pp. 132–141.

[42] N. Sarma, S. Nandi, and R. Tripathi, "Enhancing route recovery for qaodv routing in mobile ad hoc networks," in *Proceedings of the The International Symposium on Parallel Architectures, Algorithms, and Networks*, ser. ISPAN '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 39–44. [Online]. Available: http://dx.doi.org/10.1109/I-SPAN.2008.31

[43] J. Jeon, K. Lee, and C. Kim, "Fast route recovery scheme for mobile ad hoc networks," in *Information Networking (ICOIN), 2011 International Conference on*, Jan. 2011, pp. 419 –423.

[44] R. N. R. Ela Kumar, "THR: A two-hop look ahead with packet buffering protocol for manet," *International Journal of Information Technology and Knowledge Management*, vol. 4, pp. 109–112, Jan-June 2011.

[45] E. Natsheh, A. Jantan, S. Khatun, and S. Subramaniam, "Adaptive optimizing of hello messages in wireless ad-hoc networks," The International Arab Journal of Information Technology, 2007.

[46] A. Valera, W. K. G. Seah, and S. V. Rao, "Improving protocol robustness in ad hoc networks through cooperative packet caching and shortest multipath routing," IEEE Transactions on Mobile Computing, Sep 2005.

[47] C. Robert, D. S. Ranjan, and M. M. K., "Query localization techniques for on-demand routing protocols in ad hoc networks." *Wireless Networks*, vol. 8, no. 2-3, pp. 137–151, 2002.

[48] S. ju Lee and M. Gerla, "AODV-BR: Backup routing in ad hoc networks," in *Wireless Communications and Networking Conference, IEEE*, 2000, pp. 1311–1316.

[49] R. K. K. and S. R. M. C., "On the scalability of expanding ring search for dense wireless sensor networks," *J. Parallel Distrib. Comput.*, vol. 70, pp. 917–929, September 2010. [Online]. Available: http://dx.doi.org/10.1016/j.jpdc.2010.05.004

[50] S. ju Lee and M. Gerla, "Dynamic load-aware routing in ad hoc networks," IEEE International Conference on Communications (ICC), pp. 3206–3210, 2001.

[51] "The Network Simulator NS-2," http://www.isi.edu/nsnam/ns/.

[52] E. Hyytiä and J. Virtamo, "Random waypoint mobility model in cellular networks," *Wirel. Netw.*, vol. 13, no. 2, pp. 177–188, Apr. 2007. [Online]. Available: http://dx.doi.org/10.1007/s11276-006-4600-3

Appendix A

List of Acronyms

MANET	Mobile Ad Hoc Network
DSDV	Destination Sequenced Distance-Vector Routing
WRP	Wireless Routing Protocol
AODV	Ad-hoc On-demand Distance-Vector Routing
DSR	Dynamic Source Routing
ZRP	Zone Routing Protocol
AOMDV	On-demand multipath distance vector routing
AODV-BBS	AODV-Based Backup Routing Scheme
SMR	Split Multipath Routing
IBR-AODV	Implicit backup Routing-AODV
RC	Route Change
THR	Two Hop Routing
MP-DSR	Multi-path DSR
LRAODV_LP	Local Repair AODV based on Link Prediction
LCRDD	Link Failure and Congestion Aware Reliable Data Delivery
RREQ	Route Request Message
CN	Congestion Notification Flag
TQ	Transport Layer Queue
RREP	Route Reply Message
RERR	Route Error Message

RDN	Route Disconnection Notification Message
RSN	Route Successful Notification Message
RUN	Route Unsuccessful Notification Message

Appendix B

List of Notations

Q_{min}	Minimum threshold for queue occupancy
Q_{max}	Maximum threshold for queue occupancy
Q_{warn}	Warning threshold for queue occupancy
Q_{size}	Queue size
Q_{avg}	Average queue size
l	Constant parameter for determining minimum queue occupancy
h	Constant parameter for determining maximum queue occupancy
α	Weight factor used for determining average queue occupancy
w	Weight factor